# Auroracam by Mark McIntyre

This document provides brief instructions explaining how to install and operate the Aurora Camera.

The system consists of a cheap Security camera connected to a Raspberry Pi4 computer, running my Auroracam software. The software captures an image from the camera approximately every 2 seconds day and night, and then creates timelapses of the day and the night. The individual images can be monitored via a simple web interface and the same web interface can be used to view historical images (subject to storage capacity). The system automatically deletes older data to free up space.

Although the system was designed as an Aurora monitor, it will also detect other transient atmospheric phenomena such as Noctilucent and Nacreous clouds. If the Moon or Sun transit near its field of view you may also detect Iridescence, Lunar and Solar haloes and possibly sundogs and arcs. You may also find these interesting to monitor!

### Contents of the Kit

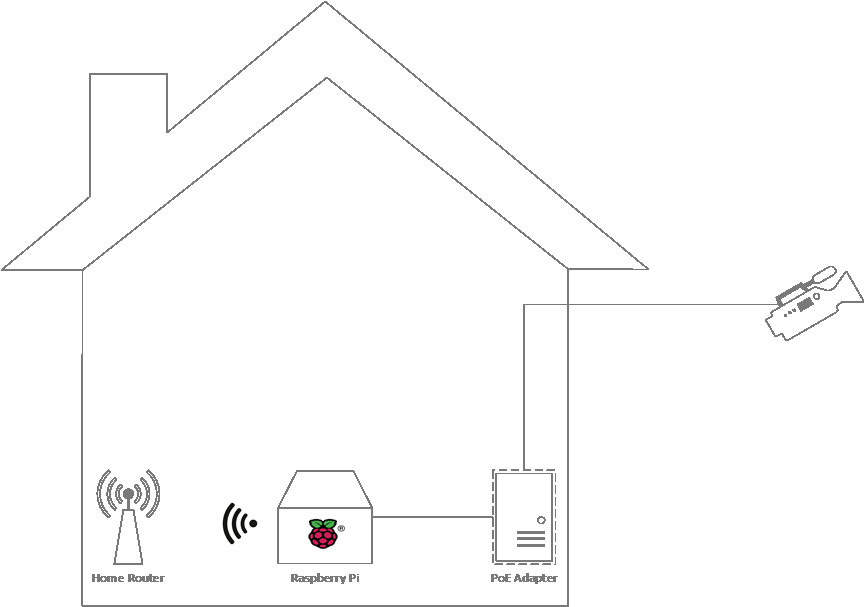
* Preassembled camera housing containing the camera module, and with a bracket for installation.
* PoE adapter to provide power and networking to the camera.
* Raspberry Pi computer preinstalled with the Auroracam software, a webserver to monitor activity, and remote access via VNC and Samba.
* A power supply for the Raspberry Pi.

You will need to provide two network cables, one long enough to go from the camera’s intended location to the PoE adapter, and another shorter cable to connect the PoE adapter to the Pi.

You will also need a keyboard, mouse and screen for the initial setup, so that you can connect the Pi to your home WiFi network.

### How it Connects Up

As the diagram below shows, the camera will connect to the PoE adapter with a long network cable. The PoE adapter’s LAN port will then be connected to the Raspberry Pi’s ethernet port with another network cable. Finally, the Pi will connect to your home network using Wi-Fi.



**Next: initial configuration**

### Initial Setup Indoors

Before installing the camera outdoors we’re going to test everything is working and connect the Pi to your network.

Check Hardware and Cables

Plug the PoE adapter into a wall socket, and connect the PoE port to the camera’s network port (white cable). The blue light on the cable should come on.

Connect the adapter’s LAN port to the Pi’s ethernet port, then power on the Raspberry Pi. The lights on the Pi’s ethernet port should come on and flicker, while the lights on the camera cable should also flicker.

After a minute, the lights on the Pi and cable should start flickering every few seconds, indicating that data capture is taking place.

Connect the Pi to your Wi-Fi

Plug the keyboard, mouse and screen into the Pi. Once you can see the desktop, click on the Network icon on the taskbar and connect to your home Wi-Fi by selecting your SSID and entering the password. Once it has connected, make a note of its IP Address which is visible in the Wi-Fi section of the Conky window that is onscreen.

Now test that you can remotely access the Pi using VNC. It should be on your home network as “auroracam2”, but if that doesn’t work you can use the IP address noted earlier. The username is “acuser” and the password is “acuser123”, though for security reasons you should change the password to something memorable.

If you want to use ssh to connect to the camera you can also add your public key to ~/.ssh/authorized\_keys. If you don’t know what this means, then its safe to ignore!

Camera Settings

You need to configure the camera with your geographical location. To do this, login to the Pi and open /home/acuser/source/auroracam/config.ini by double-clicking the link on the Desktop and selecting Execute.

Update the values of LAT, LON and ALT. East is positive and Latitude and Longitude must be decimal values so for example 51 degrees 45 minutes would be 51.75. Altitude should be in metres. It’s not vital to have precise values – its only used to decide when to create the timelapses.

Don’t change any other values in this file unless you know what you’re doing. There’s more about these in the Appendix.

**Next: Outdoor installation**

## Choosing a Camera Location

The camera should be installed outdoors in a location with a good clear view of the Northern horizon (Southern, if you’re in the southern hemisphere!). A network cable has to be run to the camera and the cable length must be no more than about 50m. Bear in mind that the cable run may be longer than the straight-line distance.

If possible, make sure there are no street lights or security lights in the field of view. A good location is up on the wall of a house, although a sturdy pole or post will also do.

Finally, consider how you will get to it for maintenance. I frequently have to spray the housing with spider repellent and you might need access to fix any problems, adjust its pointing direction or refocus it.

### Outdoor Installation

Once you’re happy everything is working, you can install the camera outdoors at your chosen location. Initially angle the camera up at about 20 degrees from the horizontal – shortly we’re going to adjust the angle to just graze your horizon. The Network plug comes with a waterproof cap and seal, but I recommend wrapping it in electrical tape just in case.

## Viewing the Live Images

Fire up a browser on your computer and go to <http://auroracam2/> which should bring up a browser window showing an image from the camera (see screenshot). The image will refresh every few seconds.

Adjust the camera pointing direction and angle of tilt till the horizon is just visible along the bottom edge.

## 

That’s it, you are done! The camera is now working, capturing data day and night. Each morning it will create a timelapse video from the data. Data will be kept as long as possible, but older data is periodically purged to make sure there’s enough space for the next run.

### Support

For support purposes, the camera is configured with Raspberry Pi Connect to allow me to remotely connect to it. If you don’t want me to be able to do this, you can disable the software.

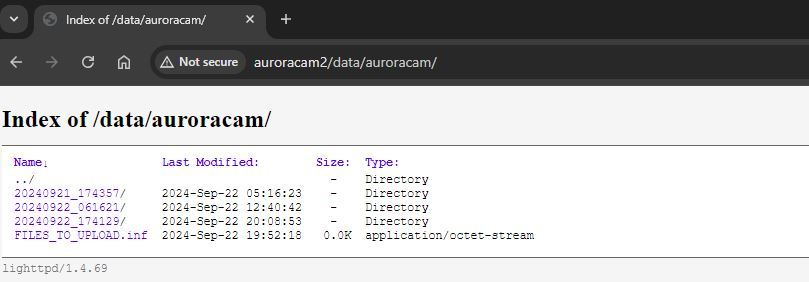
**Next: Viewing and Saving the Data**

## How Do I see the Data!?

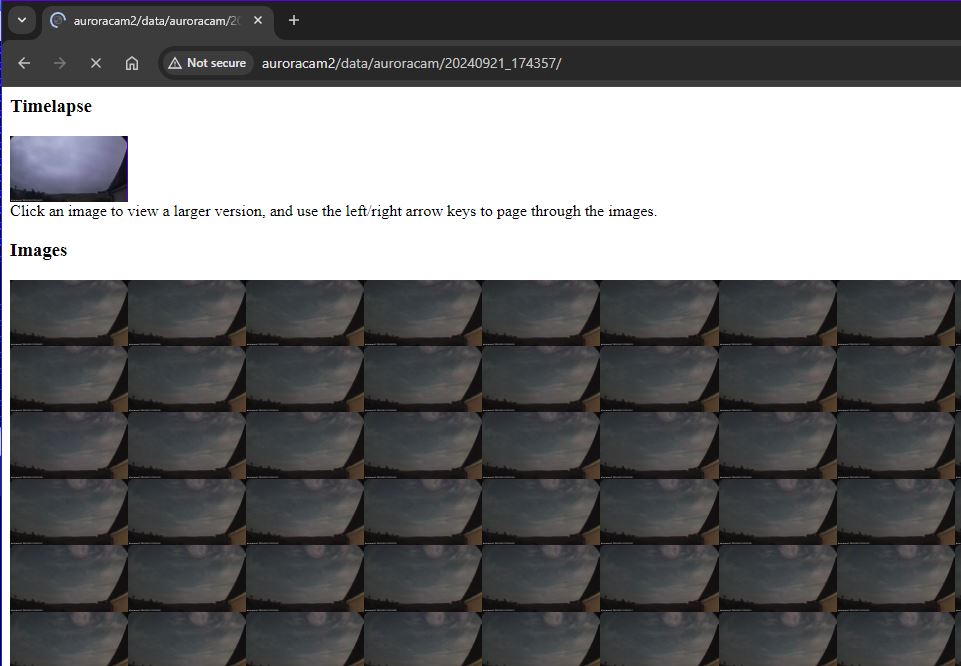
The web interface includes two links below the image, one labelled Previous Data and the other labelled Log Files.

Previous data

Clicking on *Previous Data* will take you to a list of folders containing the individual images and timelapses from previous days and nights.



Click on a folder to see the data from that period (screenshot below). You can click on the timelapse to play the video. To save the video to your PC, right click on it when it is playing. Similarly, you can click on an image to view and right click to save. When viewing the images you can use the left and right arrow keys to move to the next or previous images.



Logs

The other link takes you to the log files. This might come in handy if you have any problems.

**Next: Appendix: Additional Configuration**

## Appendix: Additional Configuration

There’s a link to the configuration file on the Desktop. Normally you will not need to make any changes as I have preconfigured it to your requirements.

However, the software has some additional capabilities which can be activated via sections of the configuration file. In particular, the software can upload an image to an FTP server or AWS S3 bucket, and can also upload a zip of any “interesting” data to an FTP server, for safe-keeping.

The file has four sections.

### [Auroracam]

This section is the main configuration of the software.

The first few settings control the basic configuration of the system. IPADDRESS is ip address of the camera module. LOGDIR, DATADIR are location of the logs and data respectively and CAMID the name that will appear in image timestamps. LAT, LON, ALT are the camera’s latitude, longitude and altitude above sea level.

DAYTIMELAPSE - set this to 0 if you NOT want to create a daytime timelapse as well as a nightly one.

DAYSTOKEEP – number of days’ data to keep on disk. At startup each morning the system checks that there’s sufficient free diskspace and deletes older files but will always try to keep at least DAYSTOKEEP worth of data.   
  
NIGHTGAIN, RGBADJ – these allow some adjustment of the camera settings. NIGHTGAIN is a number from 1 to 100 which increases the gain at night. RGBADJ allows you to change the colour balance. Its extremely unlikely you will need to change these.

### [uploads]

This section allows you to upload the live image to either an S3 bucket or an SFTP server, or both.

To upload to Amazon S3, via the AWS console create an AWS IAM user with the required access and add an Access key of type “other”. Download the key in CSV format and save it on the Pi. Now edit the config file and set S3UPLOADLOC to the bucket name and prefix you want to upload to eg *s3://mybucket/mycam* and IDKEY to point to the key file. (The IDSERVER setting is not used).

To upload to an SFTP server, enter the SFTP server name, userid and target location in the FTPSERVER, FTPUSER and FTPUPLOADLOC settings. The FTPKEY should point to a file containing the private half of an ssh keypair that has access to upload as the specified userid to the target location.

Changes will take effect after the Pi is next restarted.

### [youtube]

Its possible to automatically upload videos to Youtube but its an advanced process and is beyond the scope of most users. If you want to upload a video you can save it to your PC and then upload manually.

### [Archive]

Finally, the software can archive data to a server for safe keeping. This is configured in the same way as SFTP server uploads, with a server name, userid, user key and location. Once configured you can make use of this function as follows:

* Using the Web interface, identify the dataset you want to keep. For example, you might want to keep “20240814\_174500”, corresponding to the night of 14th August 2024.
* Login to the Pi with VNC and double click on the “Files to Keep” icon. Add an entry for the date you want to keep.
* The system will now automatically create a zip file of the data and upload it to the server.