Note to student: Attach to essay or copy and paste to front of online submission.

Note to the Marker

Please consider the following candidate's written work in respect of Specific Learning Difficulties and / or a neuro-disability with similar information processing traits (such as ADHD, autism, cerebral palsy etc)

AD12444
(Candidate Number)

This student has provided medical evidence to confirm a disability/impairment under the categories outlined above.

For The Marking of Work for Language Modules

In the case of language modules, please consider carefully what specifically is being assessed. If accuracy in the use of language is being assessed, (e.g. spelling, grammar and punctuation) then exceptions should NOT be made for disability. However, if the paper is being assessed either fully or partly on other criteria, then please follow the guidance described below.

Guidelines for Markers Assessing Coursework attached with this sheet

Unless core assessment criteria, and with the exception of essential technical vocabulary, all minor errors as described below should be disregarded in marking the work attached.

Although students with cover sheets should be marked with regards to the elements of content in their work, the logical argument of an essay or report may not be constructed in a very sequential manner. In addition, grammatical and sentence structure errors can be missed by the student.

Students with specific learning difficulties and those with other processing impairments commonly make errors with spelling, punctuation, and in the form of the omission of small function words, the addition or repetition of such words, the transposition of words and the substitution of other function words (the /a / an / for / from).

The use of spell- and grammar-checkers has a limited use for students with these challenges. Word substitution, phonetic equivalent and American spelling errors can occur.

Word count: if a word limit is applied please allow for variance due to the challenges described above affecting editing. A summary of approaches for markers is given below (from "Guidelines for Examiners" document available from the Examinations and Awards Office)

- Look for content, analysis evidence of reading/research
- Include positive/constructive comments in feedback and acknowledge cover sheet was taken into consideration
- Use clear English in feedback
- Avoid using red coloured pens for comments
- In correcting English, explain what is wrong and give examples

Academic year: 2022/23

The Optical Telescope

Sarthak Maji

Supervised by Prof. Malcolm Fairbairn

6CCP3131 Third Year Project

Contents

1. ABSTRACT

This project is about the optical telescope. The project aims to collect data on star clusters and to process that data in Python for applying physics to those data. We calculate the magnitudes of the stars to plot an HR diagram.

2. INTRODUCTION

The night sky is full of spectacles like stars, galaxies, etc. These spectacles are referred to as objects or deep sky objects in astronomy. To understand the physics behind these objects, we first need to gather data from these objects so we can comprehend how these objects in space function and evolve and the characteristics of these objects. The data from these objects are collected using a telescope. There are different types of telescopes like Optical telescopes and Radio telescopes. In addition, there are subtypes of optical telescopes like refractors, reflectors, and catadioptric telescopes. A refractor telescope is constructed of carefully crafted lenses which focus light to create an image that may be photographed [1]. A reflector telescope extends the overall light path using mirrors to reflect light at various angles inside the optical tube [1]. A catadioptric telescope combines a reflector and refractor telescope, which gives it the benefits of both telescopes, making the size of the telescope small and portable [1].

2.1. Brief history of telescope

The invention of the telescope profoundly affected how we see the night sky and helped expand our understanding of the stars. It is unknown when the telescope was invented, but the first known record of the telescope is in the Netherlands in 1608, created by Dutch spectacle maker Hans Lipperhey [2]. Early telescopes were employed in military and surveying applications. Nevertheless, Galileo Galilei was a member of the small team of astronomers that directed the telescope towards the night sky [3]. In 1609 Galileo created his telescope and presented it in Venice after learning about the Danish perspective glass [3].

2.2. Star Clusters

The term "star cluster" refers to a collection of stars with a common origin that has been gravitationally bound together for some time. These clusters are significant in astronomy because they enable studying and modelling star ages and evolution [4]. There are following types of star clusters.

2.2.1. Open Clusters

An open cluster is a collection of tens or hundreds of stars weakly linked by gravity. In an open cluster, the stars gradually drift apart over time. All the stars in an open cluster formed simultaneously; all the stars in the cluster were created from the same cloud of gas and dust, which means they had the same initial chemistry [5]. For this project, we collected data of M37, M36, and M35, all open clusters. These open clusters are part of the Messier catalogue.

2.2.2. Globular Clusters

Globular clusters are often significantly bigger and gravitationally more confined than open clusters. Because of this, they are far more densely inhabited, with populations of tens of thousands to millions of stars. Globular clusters are regularly spherical because of the strong gravitational interaction between the densely populated stars. In contrast to open clusters, they include older, redder stars [6]. We also collected data from M3, which is an open cluster from the Messier catalogue.

2.3. Nebulas

Nebula is a term used in astronomy for a cloud of gas and dust that inhabits the space between stars and newly forming stars. The term Nebula comes from Latin, meaning “mist, vapour, fog, smoke, exhalation" [7]. For a very long time, nebula was a term used to describe every observed cloudy object in space. Astronomers have been intrigued by these strange regions in the night sky since Ptolemy, who lived around 150 CE. Persian astronomer Abd al-Rahman al-Sufi wrote about the Andromeda Galaxy and remarked, "a little cloud", and was later discovered to be a galaxy. Early Chinese and Arabic astronomers also recorded how a supernova in 1054 led to the development of the Crab Nebula. The Orion Nebula, which resembles a star to the naked eye, was found in 1610, two years after the development of the telescope, by the French academic and naturalist Nicolas-Claude Fabri de Peiresc [8].

Figure 1: The Orion Nebula(M42) and the Running Man Nebula.

We must first understand that the space between the stars in the night sky is not entirely empty or vacuum before we can know how nebulas are created. Gases such as nitrogen, hydrogen, and other substances fill this void. Because of how evenly distributed these gases are, space seems to be empty. The interstellar medium, or ISM, is the name given to this area of space. The Interstellar Medium, or ISM, is where nebulas are formed. Nebulae fall into one of five categories, and each one is created differently [9].

2.3.1. Formation of Supernova Remnants

A nova or a supernova is created when a huge star in the last stages of its existence bursts because of gravity causing its core to collapse or because of nuclear fusion, leaving behind cosmic dust and gases. Then, this is referred to be a Supernova Remnant [9].

2.3.2. Formation of Emission, Reflection, and Dark Nebulae

Clouds of gas and dust that were already present in the interstellar medium are all that are needed to generate this kind of nebula. This substance just so happened to cluster together and wasn't drawn in by the gravity of any star system [9].

2.3.3. Formation of Planetary Nebulae

When a star like our Sun dies, it does not explode in a supernova or fall apart into a black hole. Instead, the dying star sheds its outer layers over time, forming a magnificent cloud known as a "planetary nebula" while its core develops into a white dwarf. The complex chemicals within the nebula shine in an intriguing spectrum when illuminated by the white dwarf, and gravity and other factors give the cloud distinctive patterns. Because of this, we frequently give planetary nebulas names based on how they seem to us, such as the Ring Nebula, Dumbbell Nebula, Stingray Nebula, and so on [10].

A protostar develops over time as the gas and dust particles in a nebula are drawn together by gravity and collapse into a core. Due to an increase in pressure and kinetic energy brought on by the gravitational collapse, the temperature in the core rises. These molecular gas clouds are also referred to as stellar nurseries since they are where stars are born [11]. Nuclear fusion begins when the temperature rises, combining hydrogen atoms to create helium. The core remains hot due to the nuclear fusion reaction, which releases energy. The star then enters a stable phase when the pull of gravity and the pressure of nuclear fusion's energy release are balanced. Like our sun, a star can remain in its stable phase for a number of billions of years. The star will gradually expand as all the hydrogen is eventually used up by nuclear fusion to create helium. The surface of this expanding star becomes red and gets considerably colder [12]. A red giant star is smaller and has a mass between 0.3-8 solar masses. Red supergiants are larger stars with masses between 8-40 solar masses. Solar mass is the mass of our sun. Red giants become white dwarfs by shedding their outer layer of gas and dust. What's left is a hot, dense core that has reached a higher surface temperature, heated up, and is now emitting white light, forming a planetary nebula. A red supergiant will continue to fuse heavier and heavier elements, burning hotter and growing until it bursts into a supernova, leaving a supernova remnant. A neutron star, or a black hole in the case of a massive star, is left behind after the explosion; dust and gas are expelled into space [12].

Nebulas play a key role in the evolution of stars since they are the source of star formation and the nebulas that result from dying stars. Planetary nebulae contribute significantly to the galaxy's chemical evolution because they recycle material into space that is rich in heavy metals and other by-products of nucleosynthesis (such as carbon, nitrogen, oxygen, and calcium). Only planetary nebulae in other galaxies may be visible well enough to yield accurate chemical abundance information [13]. Which helps in the study of distant galaxies.

2.4. Messier Catalogue

One of the most helpful resources in astronomy is the Messier Catalogue, sometimes referred to as the Messier objects. The re-emergence of Halley's comet in the middle of the eighteenth century contributed to the validation of Newton's theory and sparked renewed interest in astronomy. Charles Messier, a French astronomer, started his lifelong quest for comets during this period. He would finally find 15 of them. Messier discovered a tiny foggy object within the constellation of Taurus on August 28, 1758, while looking for comets. To avoid confusing them with comets, he started keeping a log of these hazy (cloudy) objects, which is now the Messier catalogue [14].

2.5.Background Knowledge

We first need to understand a few things to plot an HR Diagram. The axis of the HR diagram we are plotting is the apparent magnitude and colour indices(B-G). The term magnitude indicates how bright a star, or an object is, so the brighter the object, the lower the magnitude because magnitude is a log scale quantity. There are two types of magnitude apparent and absolute magnitude. The apparent magnitude is the brightness of a star are an object as observed from Earth, whereas the absolute magnitude is a measure of the brightness of a star or an object if they were 10pc away from Earth [15]. The entire amount of energy emitted by a star or another object in the sky per second is expressed as luminosity, abbreviated L. So, this is how much power a star produces. A star's bolometric luminosity is its total power output at all wavelengths [16].

$$L = 4\pi r^2 \sigma T^4 \tag{1}$$

$$m - M = 5 \log\left(\frac{d}{10}\right) \tag{2}$$

In eq.1. L is luminosity, r is the radius of the star, σ is the Stefan-Boltzmann constant and T is the temperature of the star. In eq.2. m is the apparent magnitude of the star, M is the absolute magnitude of the star and d is the distance of the star from earth.

3. Method

3.1. Collecting Data

We used Celestron EdgeHD 11" Optical Tube Assembly, a flat field Schmidt-Cassegrain optics for pinpoint stars, to gather the data for M35, M36, M37, M3, and M42. The Schmidt-Cassegrain is a catadioptric telescope that integrates a Cassegrain reflector's optical pathway and a Schmidt corrector plate to create a small astronomical instrument that utilizes straightforward spherical surfaces [17]. The Cassegrain reflector, frequently used in optical telescopes, includes a primary concave mirror and a secondary convex mirror. The primary mirror of a Schmidt or Schmidt-Cassegrain telescope design introduces spherical aberration, which is corrected with a Schmidt corrector plate, an aspheric lens. This telescope is on the roof of the King's College London Strand building.

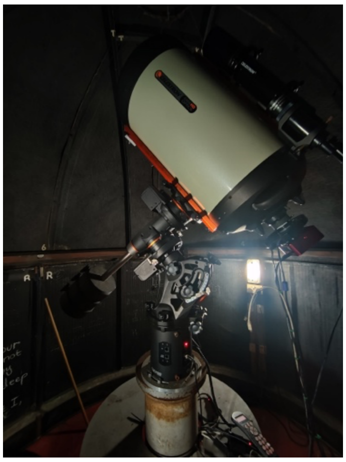

Figure 2: Celestron EdgeHD 11" Optical Tube Assembly

A guide scope for auto-tracking the star while taking images and a finder scope are attached to the telescope. A Charge-coupled device (CCD) camera is used with the telescope attached to a filter box for specific band imaging or narrow band imaging. The filters in the filter box are blue, green, Hα, OIII and SII filters. To image the objects, we first aligned the finder scope and the EdgeHD telescope so that they point to the same object. Then we put the date, time, and location to the handset to automatically polar align the telescope, which is needed for tracking the object. After the telescope is aligned, we put the object's name on the handset to auto-point itself to the thing. Then we focus the telescope and find a good spot of the object to image as the image is a zoomed image due to the large aperture of the telescope. Then we take ten images in Blue and Green Filter for M35, M36, and M37.

3.2. Processing Data

3.2.1. Fits File Handling

We use Python programming language to process the images, as the photos taken are saved as a FITS file format. A file format called FITS (Flexible Image Transport System) was created to store, transport, and alter data stored on files. A fits file contains multi-dimensional data that resembles a 2D array. Depending on the type of image, the camera's pixel data is saved as an array. For instance, if the data is RGB, it is saved as a three-layer 2D array, with one layer for each RGB colour. If the data is monochrome, it is kept as a two-dimensional array. The ASCII headers used to hold the image metadata are readable by humans. First, we must install AstroPy to handle the FITS file in Python. Then, using our Python compiler's command line function, AstroPy is installed. To install the most recent version of AstroPy, we use the pip function on the command line. We must import all required packages onto the Jupyter Notebook after installing astropy. The fits function is imported because handling fits a specific process of astropy. Then we import and open the pics for hot pixel removal, background removal, and stacking.

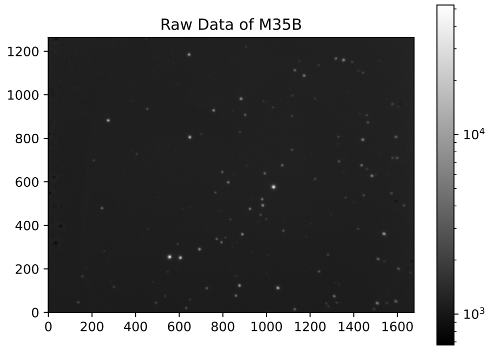

Figure 3: Raw Data of Image 1 of M35 in Blue Filter.

3.2.2. **Hot Pixel and Background Removal**

Hot pixels, or single, sharp pixels, are seen infrequently across well-known photographs captured by digital cameras because they are locations where the lens-captured incident light does not cause a linear response from them. One of their key characteristics is that heated pixels always appear in the same spot no matter the frame, implying that they remain stationary and do not move. Electrical charges that develop in the camera's lens sensor will lead to hot pixels. They can't be seen while viewing the image since they are incredibly crisp and brilliant pixels. However, they are easily discernible when closely zooming in on the images during processing, mainly if the image's backdrop is exceptionally dark, as in pictures captured through a telescope. They may also become apparent at highly high ISOs or when the temperature of the sensor rises [18]. The weather at the time of the picture session also affects the presence of hot pixels since hotter environments are suitable for developing hot pixels on the camera's sensor. Finally, they frequently show up in long-exposure photos. The hot pixels were created because, even though the scene received less light at the time, the patterns that the camera sensor had picked up at that particular moment were considerably more durable. As a result, long exposures cause lenses and camera sensors to heat up more and more. We use find outliner pixel function to remove the hot pixels from the images.

Once the hot pixels are removed, we then remove the background. We use Background2D a photutils.background package function to estimate the background in the form of a gradient. We estimate it for all the images and then subtract it from the image without the hot pixels removed. Background removal is essential because the weather in London is hazy, mainly at night, leading to certain artefacts and problems later while registering the stars. Once the background is removed, we change any values that have dropped below zero to 500 so that it doesn't give artefacts.

3.3. Stacking and HR Diagram

To stack the images, we add the processed images together to get a stacked image. These processes are repeated for all the objects and in specific filters. Now we can plot an HR Diagram using these stacked images. We first import and open the files in a new Jupyter notebook to plot an HR diagram. We then register the stars in both filters for an object. Once the leads are registered, we pair the stars in both filters. After the pairing, we calculate the magnitude of the paired stars and then calculate the B-G using the magnitudes, which is the colour index. Then we plot the magnitude of a filter vs B-G which is the HR diagram.

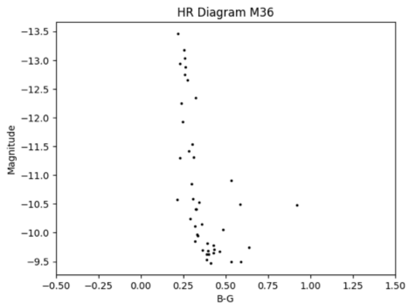

Figure 4: HR Diagram for M36.

4. Results

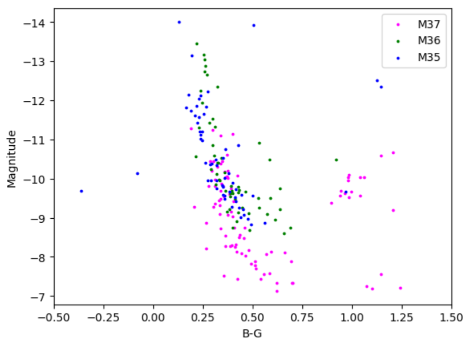

Figure 5: Combined HR Diagram for M36, M35 and M37.

The main sequence branch is readily visible on our HR diagram (Fig. 5). The apparent magnitude of stars in this region ranges from -7 to -14, with lower numbers indicating brighter stars. The stars present in the middle right of the diagram are red giants. On the y-axis, the red giant's branch is lower than anticipated. This is because the beginning of the red giant branch would probably coincide closely with the end of the main sequence since red giants were the main sequence stars that went off the main sequence as they became older.

The Main Sequence Turn-Off point is in the diagram's upper left corner. Where stars start to die and become Red Giants. A star's gravity, which keeps it together, and the pressure from continuing thermonuclear fusion processes at its core, which exerts an outward force on the star, balance each other out to keep the star stable [19]. That balance is destroyed, though, when a star's core runs out of hydrogen and the body starts to collapse. The surrounding plasma shell of the core heats up to the point where hydrogen fusion may begin when the core collapses [19]. The extra heat generated when fusion in this shell starts the star's outer layers to expand drastically, and the surface can grow up to several hundred times larger than the star's original size. The star's swollen surface cools, changing from white or yellow to red due to the energy at its surface dissipating much more [19].

Using the B-G, we can now calculate the temperatures of stars using the following formula:

$$T = \frac{5601}{((B-G)+0.4)^{2/3}} \quad (3)$$

In Python, by using the eq.3. gives us the following average temperature of the clusters.
Average Temp of the cluster M35: 6949.714594634932
Average Temp of the cluster M36: 6764.440440982434
Average Temp of the cluster M37: 6213.142702791388

The average apparent magnitudes of the clusters are as follows:
M35: -11.185200820823813
M36: -11.13077211327017
M37: -10.263532490048211

These averages were calculated using python.

We also collected data from M42 and M3. M42 is famously known as the Orion nebula. M42, a brilliant star in the constellation Orion, is thought by the Maya of Mesoamerica as the cosmic fire of creation [20]. Throughout history, several nations have been aware of this outstanding nursery. The nebula is the closest big star-forming zone to Earth and is only 1,500 light-years distant, giving it a moderately bright apparent brightness of 4 [20]. M42 may be seen with the naked eye thanks to its brightness and prominent placement just below Orion's belt, but it also provides a fantastic view of star birth for those using telescopes [20]. The Horsehead Nebula, M78, and Barnard's Loop are all components of the Orion Molecular Complex, a much larger nebula system that spans the whole Orion constellation. The Orion Molecular Complex also includes the M42 nebula [21].

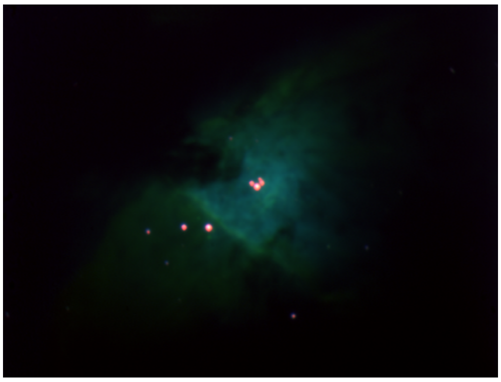

Figure 6: The Orion Nebula in Hubble Palate.

Fig.6 is a Hubble Palate image of the Orion nebula, meaning the photo was taken using narrowband filters. Photographs were taken in , Hα, OIII and SII filters and combined into an RGB Image. The SII was used as a red channel, Hα as green, and OIII as blue.

M3 was only pictured in one filter. Charles Messier's first find in his inventory, the globular cluster M3, was recorded. In 1764, Messier discovered the cluster, mistaking it for an empty nebula. In 1784, William Herschel sorted the stars in the cluster, erasing this misconception of the nature of M3. Over 500,000 stars have been identified in it as of late [22]. More variable stars than any other cluster are found in M3, which makes it noteworthy. When a star is variable, its brightness changes over time. The intrinsic brightness of some variable stars correlates with their period. These stars are very valuable for determining the distances of deep-sky objects because astronomers may infer the distances from brightness variations in those stars. The number of variable stars in M3 is at least 274 [22].

5. Conclusion

Star clusters are collections of stars with a common origin that have been gravitationally bound together for some time. There are two types of star clusters: open clusters and globular clusters. Open clusters are a collection of tens or hundreds of stars weakly linked by gravity, while globular clusters are more extensive and densely populated.

Nebula is a term used in astronomy to describe a cloud of gas and dust that inhabits the space between stars and newly forming stars. Astronomers have been intrigued by these strange regions in the night sky since Ptolemy. The interstellar medium, or ISM, is where nebulas are formed. Nebulae fall into one of five categories. Each is created differently. Supernova Remnants are begun when a massive starburst due to gravity or nuclear fusion, leaving behind cosmic dust and gases.

Emission, Reflection, and Dark Nebulae are created when clouds of gas and dust are already present in the interstellar medium. Planetary nebulae are formed when a star dies, and its core evolves into a white dwarf. The complex chemicals within the nebula shine in an intriguing spectrum when illuminated by the white dwarf, and gravity and other factors give the cloud distinctive patterns. A protostar develops over time as the gas and dust particles in a nebula are drawn together by gravity and collapse into a core. Nuclear fusion begins when the temperature rises, combining hydrogen atoms to create helium.

A protostar develops over time as the gas and dust particles in a nebula are drawn together by gravity and collapse into a core. Nuclear fusion begins when the temperature rises, combining hydrogen atoms to create helium. The star then enters a stable phase when the pull of gravity and the pressure of nuclear fusion's energy release are balanced. A red giant star is smaller and has a mass between 0.3-8 solar masses, while a red supergiant is larger and has a mass between 8-40 solar masses. A red supergiant will continue to fuse heavier and heavier elements, burning hotter and growing until it bursts into a supernova, leaving a supernova remnant.

Planetary nebulae contribute significantly to the galaxy's chemical evolution by recycling material into space that is rich in heavy metals and other by-products of nucleosynthesis. Only planetary nebulae in other galaxies may be visible well enough to yield accurate chemical abundance information.

The Celestron EdgeHD 11" Optical Tube Assembly and Schmidt-Cassegrain optics were used to gather data for M35, M36, M37, M3, and M42. The FITS file format was used to store, transport, and alter data stored on files. AstroPy was used to handle the FITS file in Python and import and open the pics for hot pixel removal, background removal, and stacking. Hot pixels are created by electrical charges in the camera's lens sensor and can be seen when zooming in on the images during processing.

Hot pixels are single, sharp pixels that appear in the same spot no matter the frame. They are created by electrical charges in the camera's lens sensor, which can be seen when zooming in on the images during processing. They can also become apparent at high ISOs or when the temperature of the sensor rises. They are often seen in long-exposure photos. Long exposures cause lenses and camera sensors to heat up, so we use the find outliner pixel function to remove hot pixels and backgrounds.

Background2D estimates the background and subtracts it from the image with the hot pixels removed. Stacking and HR Diagrams are used to plot an HR diagram. We register stars in both filters, pair them, calculate the magnitude of the paired stars, and plot the magnitude of a filter vs B-G. The main sequence branch is visible on the HR diagram (Fig. 5). Red giants are in the middle right, with lower numbers indicating brighter stars. The Main Sequence Turn-Off point is in the upper left corner, where stars start to die and become Red Giants. The average temperature of stars is calculated using the B-G formula.

The Orion Nebula is the closest big star-forming zone to Earth and is 1,500 light-years distant. The Horsehead Nebula, M78, and Barnard's Loop are all components of the Orion Molecular Complex, including the M42 nebula. Fig.6 is a Hubble Palate image of the Orion Nebula taken using narrowband filters. Charles Messier's first find in his inventory, the globular cluster M3, was recorded in 1764. Over 500,000 stars have been identified in it as of late, with more variable stars than any other cluster.

This project was conducted in hazy London weather unsuitable for astrophotography and photometry. Furthermore, the images must be taken from a dark, low-light polluted area to get the best results. To minimize any errors, we must ensure that we do not touch the telescope when clicking images and keep phones or bright light sources away from the telescope. We could have used better background removal methods for better results to get the best star pairs for the HR diagram.

References

1. **OPT Telescopes**. n.d. *The Basic Types of Telescopes*. Accessed April 17, 2023. https://optcorp.com/blogs/telescopes-101/the-basic-telescope-types.
2. **Royal Museums Greenwich**. n.d. *The history of the telescope*. Accessed April 17, 2023. https://www.rmg.co.uk/stories/topics/history-telescope.
3. **Library of Congress**. n.d. *Finding Our Place in the Cosmos: From Galileo to Sagan and Beyond*. Accessed April 17, 2023. https://www.loc.gov/collections/finding-our-place-in-the-cosmos-with-carl-sagan/articles-and-essays/modeling-the-cosmos/galileo-and-the-telescope#:~:text=While%20there%20is%20evidence%20that,Jacob%20Metius%20independently%20created%20telescopes.
4. **Australia Telescope National Facility**. n.d. *Star Clusters*. Accessed April 18, 2023. https://www.atnf.csiro.au/outreach/education/senior/astrophysics/stellarevolution_clusters.html.
5. **National Schools' Observatory**. n.d. *Open Cluster*. Accessed April 18, 2023. https://www.schoolsobservatory.org/learn/astro/stars/clusters/openclus.
6. **ESA/Hubble**. n.d. *Globular Clusters*. Accessed April 18, 2023. https://esahubble.org/wordbank/globular-cluster/.
7. **Space Center Houston**. n.d. *What is Nebula?* Accessed February 14, 2023. https://spacecenter.org/what-is-a-nebula/.
8. **Stewart, Suzy**. *"Nebula Facts – A Guide To Nebulae"*. The Planets. Accessed on February 14, 2023. https://theplanets.org/nebula-facts/.
9. **Stone, Elena**. 2022. *How are Nebulae Formed*. July 18. Accessed February 14, 2023. https://littleastronomy.com/how-is-a-nebula-formed/.
10. **Center for Astrophysics**. n.d. *Planetary Nebulas*. Accessed February 15, 2023. https://www.cfa.harvard.edu/research/topic/planetary-nebulas#:~:text=When%20a%20star%20like%20our,core%20becomes%20a%20white%20dwarf.
11. **Comunale, Joseph**. n.d. *What is a Protostar?* Early Stellar Evolution. Accessed February 15, 2023. https://study.com/academy/lesson/protostar-definition-formation-facts.html.
12. **The Science Hive**. n.d. *Stellar Evolution*. Accessed February 15, 2023. https://www.thesciencehive.co.uk/stellar-evolution-gcse.

13. **Karen B. Kwitter and R. B. C. Henry** 2022 PASP 134 022001**DOI** 10.1088/1538-3873/ac32b1
14. **Sea and Sky**. n.d. *The Messier Catalog of Deep Sky Objects*. Accessed April 18, 2023. http://www.seasky.org/astronomy/astronomy-messier.html.
15. **Perkins School of Blind**. n.d. *Apparent vs. Absolute Magnitude of Stars- Interactive Model*. Accessed April 18, 2023. https://www.perkins.org/resource/apparent-vs-absolute-magnitude-stars-interactive-model/#:~:text=absolute%20magnitude%20%E2%80%93%20a%20measure%20of,star%20as%20seen%20from%20Earth.
16. **Australia Telescope National Facility**. n.d. *Luminosity of Stars*. Accessed April 18, 2023. https://www.atnf.csiro.au/outreach/education/senior/astrophysics/photometry_luminosity.html#:~:text=Luminosity%2C%20L%2C%20is%20a%20measure,is%20called%20its%20bolometric%20luminosity.
17. **Alan**. 2022. *Maksutov-cassegrain vs Schmidt-cassegrain: Compared*. July 20. Accessed April 18, 2023. https://www.odysseymagazine.com/maksutov-cassegrain-vs-schmidt-cassegrain/
18. **MANSUROV, NASIM**. 2022. *What is ISO? The Complete Guide for Beginners*. September 15. Accessed April 19, 2023. https://photographylife.com/what-is-iso-in-photography.
19. **ESA/Hubble.** n.d. *Red Giant*. Accessed April 19, 2023. https://esahubble.org/wordbank/red-giant/#:~:text=A%20red%20giant%20forms%20after,taking%20place%20at%20its%20core.
20. **Garner, Rob, ed**. 2021. *Messier 42 (The Orion Nebula)*. August 25. Accessed April 20, 2023. https://www.nasa.gov/feature/goddard/2017/messier-42-the-orion-nebula.
21. **AstroBackyard**. n.d. *The Orion Nebula*. Accessed April 20, 2023. https://astrobackyard.com/orion-nebula/.
22. **Garner, Rob, ed**. 2021. *Messier 3*. August 25. Accessed April 20, 2023. https://www.nasa.gov/feature/goddard/2017/messier-3.

Appendix

```
#Stacking Part

import numpy as np
import matplotlib.pyplot as plt
from astropy.io import fits
from scipy.optimize import curve_fit
import pandas as pd
from IPython.display import clear_output
from tqdm.notebook import tqdm
from astropy.stats import sigma_clipped_stats
from photutils.detection import DAOStarFinder
from matplotlib.colors import LogNorm as log
from astropy.stats import SigmaClip
from photutils.background import Background2D, MedianBackground

def rb(data):
    for i in range(data.shape[0]):
```

```
        for s in range(data.shape[1]):
            if data[i][s]<0:
                data[i][s]=500
    return data

Img1 = ('M35_1306B.fit')
Img2 = ('M35_1307B.fit')
Img3 = ('M35_1308B.fit')
Img4 = ('M35_1309B.fit')
Img5 = ('M35_1310B.fit')
Img6 = ('M35_1311B.fit')
Img7 = ('M35_1312B.fit')
Img8 = ('M35_1313B.fit')
Img9 = ('M35_1314B.fit')
Img10 =('M35_1315B.fit')
Imgo = ('M35_1316Boff.fit')

hdul1 = fits.open(Img1)
hdul2 = fits.open(Img2)
hdul3 = fits.open(Img3)
hdul4 = fits.open(Img4)
hdul5 = fits.open(Img5)
hdul6 = fits.open(Img6)
hdul7 = fits.open(Img7)
hdul8 = fits.open(Img8)
hdul9 = fits.open(Img9)
hdul10 = fits.open(Img10)
hdulo = fits.open(Imgo)

Data1 = hdul1[0].data.astype('int32')
Data2 = hdul2[0].data.astype('int32')
Data3 = hdul3[0].data.astype('int32')
Data4 = hdul4[0].data.astype('int32')
Data5 = hdul5[0].data.astype('int32')
Data6 = hdul6[0].data.astype('int32')
Data7 = hdul7[0].data.astype('int32')
Data8 = hdul8[0].data.astype('int32')
Data9 = hdul9[0].data.astype('int32')
Data10 = hdul10[0].data.astype('int32')
Datao = hdulo[0].data.astype('int32')


def find_outlier_pixels ( data , tolerance=3 , worry_about_edges=True ):
    #This function finds the hot or dead pixels in a 2D dataset.
    #tolerance is the number of standard deviations used to cutoff the hot pixels
    #If you want to ignore the edges and greatly speed up the code, then set
    #worry_about_edges to False.
    #
    #The function returns a list of hot pixels and also an image with with hot pixels removed
```

```
from scipy.ndimage import median_filter
blurred = median_filter ( data , size=2 )
difference = data - blurred
threshold = 1*np.std(difference)

#find the hot pixels, but ignore the edges
hot_pixels = np.nonzero( (np.abs (difference [1:-1,1:-1] ) > threshold) )
hot_pixels = np.array(hot_pixels) + 1 #because we ignored the first row and first column

fixed_image = np.copy (data) #This is the image with the hot pixels removed
for y,x in zip (hot_pixels[0],hot_pixels[1]):
    fixed_image [y,x] = blurred[y,x]

if worry_about_edges == True:
    height,width = np.shape (data)

    ###Now get the pixels on the edges (but not the corners)###

    #left and right sides
    for index in range (1,height-1):
        #left side:
        med  = np.median (data[index-1:index+2,0:2])
        diff = np.abs (data[index,0] - med)
        if diff>threshold:
            hot_pixels = np.hstack(( hot_pixels, [[index],[0]]  ))
            fixed_image[index,0] = med

        #right side:
        med  = np.median(data[index-1:index+2,-2:])
        diff = np.abs(data[index,-1] - med)
        if diff>threshold:
            hot_pixels = np.hstack(( hot_pixels, [[index],[width-1]]  ))
            fixed_image[index,-1] = med

    #Then the top and bottom
    for index in range(1,width-1):
        #bottom:
        med  = np.median(data[0:2,index-1:index+2])
        diff = np.abs(data[0,index] - med)
        if diff>threshold:
            hot_pixels = np.hstack(( hot_pixels, [[0],[index]]  ))
            fixed_image[0,index] = med

        #top:
        med  = np.median(data[-2:,index-1:index+2])
        diff = np.abs(data[-1,index] - med)
        if diff>threshold:
            hot_pixels = np.hstack(( hot_pixels, [[height-1],[index]]  ))
            fixed_image[-1,index] = med
```

```
        ###Then the corners###

        #bottom left
        med  = np.median(data[0:2,0:2])
        diff = np.abs(data[0,0] - med)
        if diff>threshold:
            hot_pixels = np.hstack(( hot_pixels, [[0],[0]]  ))
            fixed_image[0,0] = med

        #bottom right
        med  = np.median(data[0:2,-2:])
        diff = np.abs(data[0,-1] - med)
        if diff>threshold:
            hot_pixels = np.hstack(( hot_pixels, [[0],[width-1]]  ))
            fixed_image[0,-1] = med

        #top left
        med  = np.median(data[-2:,0:2])
        diff = np.abs(data[-1,0] - med)
        if diff>threshold:
            hot_pixels = np.hstack(( hot_pixels, [[height-1],[0]]  ))
            fixed_image[-1,0] = med

        #top right
        med  = np.median(data[-2:,-2:])
        diff = np.abs(data[-1,-1] - med)
        if diff>threshold:
            hot_pixels = np.hstack(( hot_pixels, [[height-1],[width-1]]  ))
            fixed_image[-1,-1] = med

    return hot_pixels,fixed_image




hot_pixels1,fixed_image1 = find_outlier_pixels (Data1, worry_about_edges=True)
print(np.count_nonzero(hot_pixels1))
hot_pixels2,fixed_image2 = find_outlier_pixels (Data2, worry_about_edges=True)
print(np.count_nonzero(hot_pixels2))
hot_pixels3,fixed_image3 = find_outlier_pixels (Data3, worry_about_edges=True)
print(np.count_nonzero(hot_pixels3))
hot_pixels4,fixed_image4 = find_outlier_pixels (Data4, worry_about_edges=True)
print(np.count_nonzero(hot_pixels4))
hot_pixels5,fixed_image5 = find_outlier_pixels (Data5, worry_about_edges=True)
print(np.count_nonzero(hot_pixels5))
hot_pixels6,fixed_image6 = find_outlier_pixels (Data6, worry_about_edges=True)
print(np.count_nonzero(hot_pixels6))
hot_pixels7,fixed_image7 = find_outlier_pixels (Data7, worry_about_edges=True)
print(np.count_nonzero(hot_pixels7))
hot_pixels8,fixed_image8 = find_outlier_pixels (Data8, worry_about_edges=True)
```

```
print(np.count_nonzero(hot_pixels8))
hot_pixels9,fixed_image9 = find_outlier_pixels (Data9, worry_about_edges=True)
print(np.count_nonzero(hot_pixels9))
hot_pixels10,fixed_image10 = find_outlier_pixels (Data10, worry_about_edges=True)
print(np.count_nonzero(hot_pixels10))
hot_pixelso,fixed_imageo = find_outlier_pixels (Datao, worry_about_edges=True)
print(np.count_nonzero(hot_pixelso))

sigma_clip = SigmaClip(sigma=3.0)
bkg_estimator = MedianBackground()
bkg1 = Background2D(fixed_image1, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg2 = Background2D(fixed_image2, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg3 = Background2D(fixed_image3, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg4 = Background2D(fixed_image4, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg5 = Background2D(fixed_image5, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg6 = Background2D(fixed_image6, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg7 = Background2D(fixed_image7, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg8 = Background2D(fixed_image8, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg9 = Background2D(fixed_image9, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)
bkg10 = Background2D(fixed_image10, (20, 30), filter_size=(1, 1), sigma_clip=sigma_clip, bkg_estimator=bkg_estimator)

BR1 = fixed_image1 - bkg1.background
BR2 = fixed_image2 - bkg2.background
BR3 = fixed_image3 - bkg3.background
BR4 = fixed_image4 - bkg4.background
BR5 = fixed_image5 - bkg5.background
BR6 = fixed_image6 - bkg6.background
BR7 = fixed_image7 - bkg7.background
BR8 = fixed_image8 - bkg8.background
BR9 = fixed_image9 - bkg9.background
BR10 = fixed_image10 - bkg10.background

BR1 = rb(BR1)
BR2 = rb(BR2)
BR3 = rb(BR3)
BR4 = rb(BR4)
BR5 = rb(BR5)
BR6 = rb(BR6)
BR7 = rb(BR7)
BR8 = rb(BR8)
```

```
BR9 = rb(BR9)
BR10 = rb(BR10)

Stack = (BR1+BR2+BR3+BR4+BR5+BR6+BR7+BR8+BR9+BR10)

plt.style.use('default')
plt.figure()
plt.title('Stack M35B')
plt.imshow(Stack, cmap='gray', vmin=0, vmax=30000)
plt.gca().invert_yaxis()
plt.colorbar()

image_format = 'svg' # e.g .png, .svg, etc.
image_name = 'M35B Stack2.svg'

plt.savefig(image_name, format=image_format, dpi=1200)

NewImg= fits.PrimaryHDU(Stack)
NewImg.writeto('New_M35B.fits', overwrite=True)

#HR Plot
import numpy as np
import matplotlib.pyplot as plt
from astropy.io import fits
from scipy.optimize import curve_fit
import pandas as pd
from IPython.display import clear_output
import IProgress
from tqdm import tqdm
from astropy.stats import sigma_clipped_stats
from photutils.detection import DAOStarFinder
from photutils.detection import find_peaks
from astropy.stats import SigmaClip
from scipy.optimize import curve_fit
from photutils.background import Background2D, MedianBackground

Img1 = ('Stacked_M36B.fits')
Img2 = ('Stacked_M36G.fits')

hdul1 = fits.open(Img1)
hdul2 = fits.open(Img2)

DataB = hdul1[0].data.astype('int32')
DataG = hdul2[0].data.astype('int32')

meanB = np.mean(DataB)
medianB = np.median(DataB)
stdB = np.std(DataB)
print(stdB)
starfind = DAOStarFinder(fwhm=4, threshold=stdB)
```

```
starsB = find_peaks(DataB-medianB,threshold=2*stdB)
print(starsB)

meanG = np.mean(DataG)
medianG = np.median(DataG)
stdG = np.std(DataG)
print(stdG)
starfind = DAOStarFinder(fwhm=4, threshold=stdG)
starsG = find_peaks(DataG-medianG,threshold=2*stdG)
print(starsG)

pairsbx=[]
pairsby=[]
pairsgy=[]
pairsgx=[]
PeakB=[]
PeakG=[]
for i in tqdm(range(len(starsB))):
        x_position1=starsB['x_peak'][i]
        y_position1=starsB['y_peak'][i]
        Pb=starsB['peak_value'][i]
        x_values=[]
        y_values=[]
        for s in range(len(starsG)):
            x_value=starsG['x_peak'][s]
            y_value=starsG['y_peak'][s]
            Pg=starsG['peak_value'][s]
            x_values=np.append(x_values,x_value)
            y_values=np.append(y_values,y_value)
            if x_position1==x_value or y_position1==y_value:
                pairsbx=np.append(pairsbx,x_position1)
                pairsby=np.append(pairsby,y_position1)
                pairsgx=np.append(pairsgx,x_value)
                pairsgy=np.append(pairsgy,y_value)
                PeakB=np.append(PeakB,Pb)
                PeakG=np.append(PeakG,Pg)
                np.delete(starsG,s)
                break
            else:
                continue

pairsbx=np.array(pairsbx)
pairsby=np.array(pairsby)
pairsgx=np.array(pairsgx)
pairsgy=np.array(pairsgy)
PeakB=np.array(PeakB)
PeakG=np.array(PeakG)

print(PeakB)
print(PeakG)
```

```
print(np.count_nonzero(PeakG))

mag_B = -2.5 * np.log10(PeakB)
mag_G = -2.5 * np.log10(PeakG)
colour = mag_B - mag_G
print (colour)

print(np.mean(mag_G))

plt.scatter(colour,mag_B, s=3, c='black')
plt.title('HR Diagram M36')
plt.xlabel('B-G')
plt.ylabel('Magnitude')
plt.xlim(-0.5,1.5)
plt.gca().invert_yaxis()
plt.show()
image_format = 'svg' # e.g .png, .svg, etc.
image_name = 'M36B HR.svg'
plt.savefig(image_name, format=image_format, dpi=1200)

df = pd.DataFrame({"Mag B" : mag_B, "Colour" : colour})
df.to_csv("M37.csv", index=False)

T = (5601/((colour+0.4)**(2/3)))
print('Temperatures of the Stars','\n',T)
colour= colour[np.logical_not(np.isnan(T))]
mag_B = mag_B[np.logical_not(np.isnan(T))]
mag_G = mag_G[np.logical_not(np.isnan(T))]
T = T[np.logical_not(np.isnan(T))]
MT = np.mean(T)
print('\nAverage Temp of the cluster: ',MT)
```