university of groningen / faculty of science and engineering / bernoulli institute

Application of the Ant Colony Algorithm in the Identification of Globular Clusters

Bachelor's Project Thesis

Jördis Hollander, s2956543, j.hollander.6@student.rug.nl

Supervisor: Prof. Dr. H. Jaeger

October 30, 2021

Abstract

Globular clusters (*GC*s) are stellar agglomerates of about 10 000 to 100 000 stars. They provide an interesting ground to study stellar evolution. The complexity of the Universe makes the precise identification and classification of stellar structures challenging. In this paper, a pipeline for the identification of GCs is developed based on work by M. Mohammadi et al.[1] This pipeline consists of filtering candidate regions through the use of a blob-detection technique. The remaining regions are then processed by the Ant colony random-walk algorithm. This algorithm investigates a region and provides information on its stellar density in the form of pheromone values. Finally, these results are fed into a *gravity-inspired* clustering algorithm that was developed to interpret the pheromone values to determine potential GCs.

The aim of the research is to determine the accuracy of the pipeline in classifying GCs and investigate possible improvements. This pipeline is then applied on the Gaia DR2 data-set. Different regions consisting of a variety of stellar objects are selected. Some of these regions contain known GCs, while some do not. The accuracy of the pipeline is explored by running it on these different regions. The evaluation of the functionality of the pipeline firstly considers if it finds all known GCs and secondly looks at what other stellar structures it classifies as GCs.

The blob-detection technique filters away 87.5 % (813 out of 929) of the candidate regions and maintains 23 of the 29 regions that contain GCs. In constrast the Ant Colony algorithm coupled with the gravitation clustering filters away 86.0 % (799 of the 929 regions) and only maintains 10 of the 29 regions containing GCs. Across the 130 regions the Ant Colony algorithm identifies 164 distinct clusters. TODO: **Give numbers for whole pipeline**. This research shows that the blob-detection technique is effective as a pre-processing step. The Ant Colony algorithm and the graviational clustering technique seem to not be effective. However, deeper evaluation of the results highlights areas for systematic improvement. By their synergy with initial blob filteration, through further tweaking of the Ant Colony parameters coupled with further refinement of the filtration criterion for the clustured stars, this pipeline can likely be made robust.

[1] M. Mohammadi, N. Petkov, K. Bunte, R. Peletier, and F.-M. Schleif, "Globular Cluster Detection in the Gaia Survey," *Neurocomputing*, vol. 342, pp. 164–171, 2019.

Acknowledgements

This work has made use of data from the European Space Agency (ESA) mission *Gaia* (https://www.cosmos.esa.int/gaia), processed by the *Gaia* Data Processing and Analysis Consortium (DPAC, https://www.cosmos.esa.int/web/gaia/dpac/consortium). Funding for the DPAC has been provided by national institutions, in particular the institutions participating in the *Gaia* Multilateral Agreement.

Additionally, this work has made use of the VizieR catalogue access tool, CDS, Strasbourg, France (DOI: 10.26093/cds/vizier). The original description of the VizieR service was published in A&AS 143, 23.

Thanks is also given to the Center for Information Technology of the University of Groningen for providing access to the Peregrine high-performance computing cluster.

Contents

Chapter 1

Introduction

The speed of light in a vacuum is constant. This fact has enabled us to look into the past and observe how the Universe has evolved over time. From the cosmic soup, to the first stars, to the formation of galaxies; the evolution of the Universe has been a source of fascination across history. One area of particular interest is the evolution of galaxies. Early on most galaxies were small, but gradually these smaller galaxies merged and amalgamated to form larger galaxies [1]. Over time, as they continued to evolve, they began to manifest a great variety of galactic structures. However, this begins to raise a question: *How does one classify the stages of evolution for a galaxy or determine where it is in its evolution*?

Hubble's Tuning Fork was the first classification scheme that sought to answer this, based on a galaxy's structure and size [2]. However, this scheme proved to be insufficient in the face of the complexity of the Universe and the variety of possible galactic formations [2]. An alternative approach to this classification scheme is to attempt to determine the age of galaxies based on their composition and not their overall shape. One such method involves determining the ages of various clusters of stars within it, thereby providing insight into the origin of the stars making up the galaxy.

Of the various cluster types, the globular cluster (GC) is on average the oldest [3], and thus, is the most significant in gauging the age of galaxies. These types of clusters are stellar agglomerates [4] which formed in one of two ways [5]:

1. Through the compression of halo gas in the cosmic re-ionization phase early on in the formation of the Universe.
2. In the collapse of enormous molecular clouds triggered by events such as the collision of gas-rich galaxies.

They are typically composed of around 10 000 to 100 000 stars [6] bound tightly by gravity into a spherical formation. Some GCs are among the oldest objects in the Universe [7] and are thus an interesting ground to study galactic evolution [8]. These older GCs manifest some specific properties such as low metallicity [9], and through a combination of techniques, such as: horizontal branch morphology, analysis of white dwarf cooling sequences, and comparisons using the main-sequence turn-off location [7], [10], may have their age accurately determined, thereby bounding the age of the galaxy they are contained within. The question remains: *How are these GCs found?*

1.1 Prior Methods of Identifying GCs

Astronomy is primarily an observational science, which, for most of its history, has gleaned information by looking up to the night sky with nothing but the naked eye. It was in this manner that the first GC, Messier 22, was discovered. It was observed by Abraham Ilhe on August 6th, 1665 and as may be seen in Figure 1.1 demonstrates an unusually dense core [11]. This GC became the subject of much research across the 1900s and the statistics it has provided has been a basis for the identification of GCs today. The primary method for detecting news GCs has involved statistical analysis of photometric data across the Universe. Properties such as mean luminosity and metallicity are extracted from known GCs and are used to filter regions based on this spectroscopic information [12], [13]. To date, over 150 GCs have been discovered using such techniques in our Milky Way [14].

Figure 1.1: Crammed Center of Messier 22 taken by ESA/Hubble [15]

As a brief interlude, the advent of the large-scale tools used in modern astronomy have allowed for the collection of enormous amounts of information within our galaxy and far beyond. Telescopes (refractor, reflector, radiographic, spectrographic, and x-ray) allow us to extract a variety of information all without leaving the Earth's surface [16]. To collect information without the interference of the Earth's atmosphere we make use of satellites and space observatories that have been launched into orbit around our planet [17]. Occasionally, a space probe will be sent beyond our orbit to collect information from asteroids, planets, or their moons within our solar system [17]. However, for stellar observations, the sheer distances involved, means that, they are all made with telescopes.

Thus, the techniques used for the identification of GCs were predominantly limited to the evaluation of photometric data and performing statistical analysis. However, this has changed with the first Gaia data release in September 2016. In this data-set stars are represented by IDs and are coupled with a variety of astrometric, photometric, and spectroscopic readings [18] (see Chapter 2 for details). This allows processing to occur on a per-star basis and allows for a greater variety of techniques to be employed. The state-of-the-art in GC detection across such data-sets is described by Mohammadi et al. in 2018 [8].

In their paper, they make use of 3D kernel density estimations (KDE) across subdivisions of the data-set. Using these estimations they then evaluated two different techniques:

- Nearest-neighbors search.
- Kernel based anomaly detection through the training of a support-vector machine on random portions of the sky to search for outliers.

For both techniques, they then use a blob-detection technique, Difference of Gaussian (DoG), as a post-processing method to further filter the regions.

1.2 Objectives

The research in this paper follows from the work of Mohammadi et al. and seeks to automate the discovery of potential GCs using data from stellar catalogs such as the Gaia data-set. The objective is to produce a pipeline using:

1. Blob-detection techniques: specifically, Difference of Gaussian, as a crude initial filter that reduces the number of candidate regions for further inspection;
2. Ant Colony random-walk algorithm: to compute density information in the form of pheromone values;
3. Clustering: via an algorithm which pools these pheromone values in 3D space gravitationally to determine clusters.

This pipeline is optimized and tested on data from the Gaia DR2 data-set [18], [19]. Different regions consisting of a variety of stellar objects are selected. Some of these regions contain previously found GCs, while some do not. Since we cannot state with absolute certainty the total number of GCs (for any given region, more may yet to be found), the accuracy of the pipeline is explored by

running it on these different regions and then evaluating it in two ways. First, we determine if it finds all known GCs, and second we consider what other stellar structures (if any) it classifies as GCs.

A robust classifier for GCs would be a useful tool in exploring the Universe and this research is especially exciting with the full publication of the Gaia DR3 data-set slated for release in the first-half of 2022 [20].

It is important to note that due to the scope of this project, the emphasis does not lie in precisely optimizing the parameters associated with the pipeline. The aim is to identify the effectiveness and shortcomings of the individual stages of the pipeline, as well as to identify potential parametric relationships between the different stages.

1.3 Summary

The rest of this paper is structured as follows:

First, we take a closer look at the Gaia Data Release 2 in Chapter 2. We begin by depicting the stellar regions under evaluation. This is followed by a description of various astronomical objects that will be of interest in the analysis. Additionally, the parameters that will be made use of from the data-set are scrutinized.

In the chapter on methodology, Chapter 3, the overview of the pipeline is provided. This is coupled with in-depth descriptions of the three techniques in use, namely, Difference of Gaussian, the Ant Colony algorithm, and the gravitational clustering algorithm.

The results are explored in Chapter 4. It discusses the findings for the individual components of the pipeline, alongside the experimental variations of the parameters that are done. Additionally, it includes the statistical results and plots to display various interesting characteristics. The results for the four regions will be compared and contrasted to explore the differences in the outcomes.

The conclusion, Chapter 5, makes a statement on the effectiveness of the pipeline based on the results and discusses how the results compare against the objectives that were set forth.

The evaluation, Chapter 6, tries to interpret the results, provide an insight into how the components of the pipeline work, and attempts to address successes and shortcomings. It also describes interesting avenues for further research and improvement.

Chapter 2

Data

The stellar data selected for this project is a subset of the Gaia DR2 data-set [19]. This data was collected during the Gaia satellite mission [19] and is approximately 500 GB of data consisting of a variety of parameters collected on a per-star basis. This is in great contrast with typical telescopic data which is primarily raw radiometric imagery and greatly expands the types of processing that may be conveniently applied. The subset of the parameters that are of interest to the pipeline are described in Section 2.1. As a result of hardware limitations, it is necessary to limit the investigation to a set of smaller regions from within the Gaia data-set. Four distinct areas were selected, for which, the cosmic ranges as well as the number of stars within the region may be found in Table 2.1. These four areas are bounded by right ascension (RA) $\times$ declination (Dec) and represent regions of interest with varying stellar distributions.

Table 2.1: Areas Under Investigation

Area	RA	Dec	Number of Stars
A1	120° up to 246°	−2° up to 60°	25 486 556
A2	295° up to 308°	15° up to 25°	23 470 239
A3	0° up to 75°	−90° up to −30°	16 781 316
A4	0° up to 45°	30° up to 70°	32 333 936

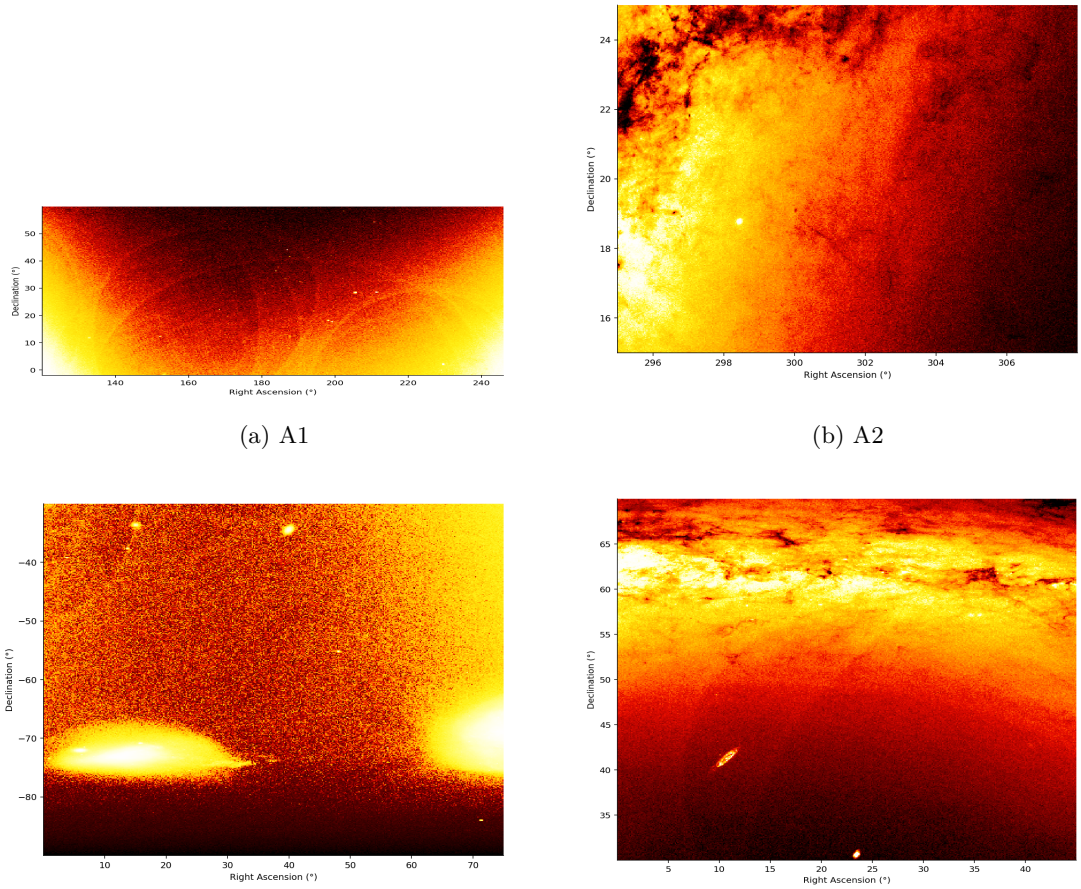
Why were these four regions selected?

Area 1 (A1): This region was used in the precursor work of Mohammadi et al. [8]. It is the largest region of the four in terms of their span across RA and Dec. Conversely, it also has the lowest density of stars across the region. This results in large areas that contain very few stars. These *dark* regions are very apparent in Figure 2.1a. It contains several known GCs, including a number in these *dark* regions, which are expected to be identified easily as they stand out from their dark and void stellar surroundings.

Area 2 (A2): The region was chosen because it is a much smaller area with a very high-density of stars across the whole span. Additionally, it contains only one GC. This is useful as a testing ground to see how the pipeline would handle regions featuring a high amount of stars.

Area 3 (A3): This area was also used in the paper by Mohammadi et al. and though its overall density is less than for A2, it features specific regions with an extremely high density of stars. Additionally, it contains 16 GCs, the Magellanic Clouds (the name given to a specific pair of Dwarf galaxies), a supernova remnant, open clusters, and galaxies. The Magellanic Clouds are the two extremely bright, very densely packed regions that may be seen in Figure 2.1c. These two very bright regions also contain some GCs and it would be interesting to see how the pipeline handles classifying these clusters with the interference of the Magellanic clouds.

Area 4 (A4): This area has no GCs, but contains the nearby and very bright Andromeda and Triangulum Galaxies. These galaxies lie across the range of 30°–50° Dec, and it would be interesting to see if the pipeline detects them and classifies them as GCs. The remaining range, from 50°–70° contains a large number of nubulae (large regions of very bright loosely-packed stellar gas).

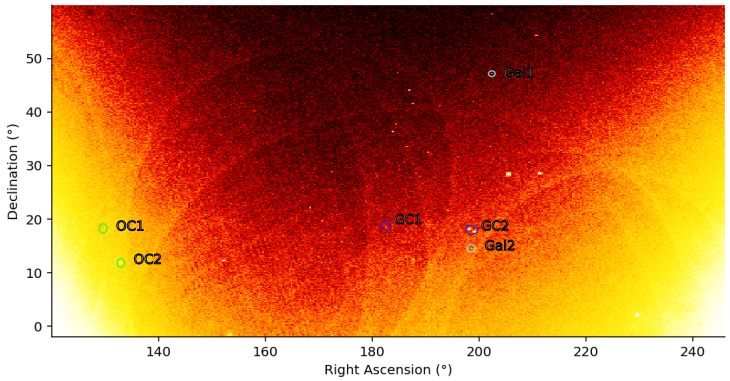


(a) A1 (b) A2

(c) A3 (d) A4

Figure 2.1: Stellar Distribution Heat-maps for the Four Areas

Figure 2.1 depicts heat-maps of the four regions synthesized from the stellar information in the Gaia DR2 data-set. It provides an insight into the population and density of the stars found within the four areas. The brighter areas (yellow–white) contain more stars than the darker areas (red–black). From these heat-maps, the spots of increased stellar density are very evident. However, it is not immediately apparent whether these spots are GCs, Open Clusters (OC), galaxies (Gal), or some other stellar structure. Figure 2.2, provides an example of some of these stellar structures for Area 1 and further highlights the difficulty these classifying stellar structures by eye alone.



○: Open Clusters ○: Globular Clusters ○: Galaxies

OC1: Messier 44	GC1: NGC4147	Gal1: The Whirlpool Galaxy
OC2: Messier 67	GC2: NGC5024 and NGC5053	Gal2: Malin 1

Figure 2.2: Stellar Structures Present in A1

Area 1 contains more than the three GCs encircled in Figure 2.2. It contains an additional nine GCs (for a total of 12), Area 2 contains just one known GC, and Area 3 contains 16 GCs [21], [22]. See Table 4.13 for information on the RA, the Dec, and the angular diameter (*DIA*) for these globular clusters where this information has been published. Note that the *DIA* is represented in arcminutes (′) which is a measure of angular distance where $1' = \frac{1}{60}°$ and is useful to provide an expectation of the size of a GC.

Table 2.2: Known GCs

GC	RA (°)	Dec (°)	DIA (′)
	A1		
M3	205.548 42°	28°22′38.2″	18′
M5	229.639 62°	2°4′54.9″	21.6′
NGC 5024	198.229 45°	18°1′5.4″	13′
NGC 4147	182.526 26°	18°32′33.5″	4.4′
NGC 5053	199.112 88°	17°42′0.5″	10′
NGC 5466	211.363 71°	28°32′4.0″	9′
Koposov 1	179.827 09°	12°15′36.0″	—
Palomar 3	151.382 92°	0°4′18.0″	1.6′
Palomar 4	172.319 99°	28°58′24.9″	1.3′
Palomar 5	229.021 87°	0°6′41.8″	8.0′
GCI 38	242.752 47°	14°57′28.0″	2.2′
Willman 1	162.35°	51°3′0.0″	7′
	A2		
M71	298.44°	18°46′45.1″	7.2′
	A3		
47 Tucanae	6.023 63°	−72°4′52.6″	50′
NGC 121	6.701 04°	−71°32′8.4″	–
NGC 1049	39.968 75°	−34°16′8″	–
NGC 362	15.809 42°	−70°50′55.6″	14′
NGC 1261	48.067 54°	−55°12′59.2″	6.85′
NGC 1629	67.404 17°	−71°50′18″	–
NGC 1644	69.415 00°	−66°11′49″	–
NGC 1651	69.386 25°	−70°35′8″	–
NGC 1652	69.595 42°	−68°40′23″	–
NGC 1696	72.125 00°	−68°14′35″	–
NGC 1756	73.707 92°	−69°14′15″	–
NGC 1783	74.786 67°	−65°59′7″	–
NGC 1786	74.782 91°	−67°44′43″	–
NGC 1795	74.945 83°	−69°48′5″	–
NGC 1841	71.346 25°	−83°59′49″	–
Arp Madore 1	58.761 25°	−49°36′52.0″	–

It is these GCs that are used in the evaluation of the results of the pipeline.

2.1 The Parameters

The Gaia DR2 data-set provides up to 88 parameters per star [19]. Of these parameters, there are six that are required for the pipeline:

1. Apparent Magnitude (`phot_g_mean_mag`): This is a unitless quantity and is the measure of a star's brightness when observed from Earth. Note, that a higher apparent magnitude corresponds to a less bright star.
2. Right Ascension (`ra`): This quantity is represented in degrees (°) and when coupled with the declination it provides a position for an astronomical body in the equatorial coordinate system.
3. Declination (`dec`): This quantity is represented in degrees (°) and when coupled with the right ascension it provides a position for an astronomical body in the equatorial coordinate system.
4. Parallax (`parallax`): This quantity is measured in milliarcseconds (mas) and is the difference in the apparent position of an object when viewed along two different lines of sight [23].
5. Proper Motion of Right Ascension (`pmra`): Expressed in $\mathrm{mas\,yr^{-1}}$ and is the motion of an astronomical body from the frame of the center of mass of the solar system in right ascension.
6. Proper Motion of Declination (`pmdec`): Expressed in $\mathrm{mas\,yr^{-1}}$ and is the motion of an astronomical body from the frame of the center of mass of the solar system in declination.

The details underlying the selection of these parameters alongside statistical analysis is provided in the sections that follow.

For the development of the pipeline, it is important to note, that the Gaia DR2 data-set does not include the `parallax`, `pmra`, or `pmdec` for all stellar objects. These parameters are missing for approximately 30 to 40% of the stars within the data-set [24]. Since these parameters are required for the pipeline, the stars that are missing these parameters have been filtered out. The details on how many stars are filtered out per area and the percentage that remain may be found in Table 2.3.

Table 2.3: Data Point Removals

Area	Total Number of Stars	Remaining Stars	Percentage of the Remaining
A1	25 486 556	17 933 864	70.4%
A2	23 470 239	14 268 513	60.8%
A3	16 781 316	9 961 034	59.4%
A4	32 333 936	22 243 660	68.8%

2.1.1 Apparent Magnitude

The apparent magnitude provides information on the brightness of a star. In isolation this is influenced by the proximity of the star to the observer and by the mass of the star. The closer the star is, the brighter it will appear. The more massive the star, the higher its black body radiation; thus, the more energy it releases in the form of brighter light.

This parameter is used alongside, RA and Dec, to create the 2-dimensional mapping that is fed into the Difference of Gaussian filters to detect blobs. Histograms of the apparent magnitude across the four areas may be seen in Figure 2.3.

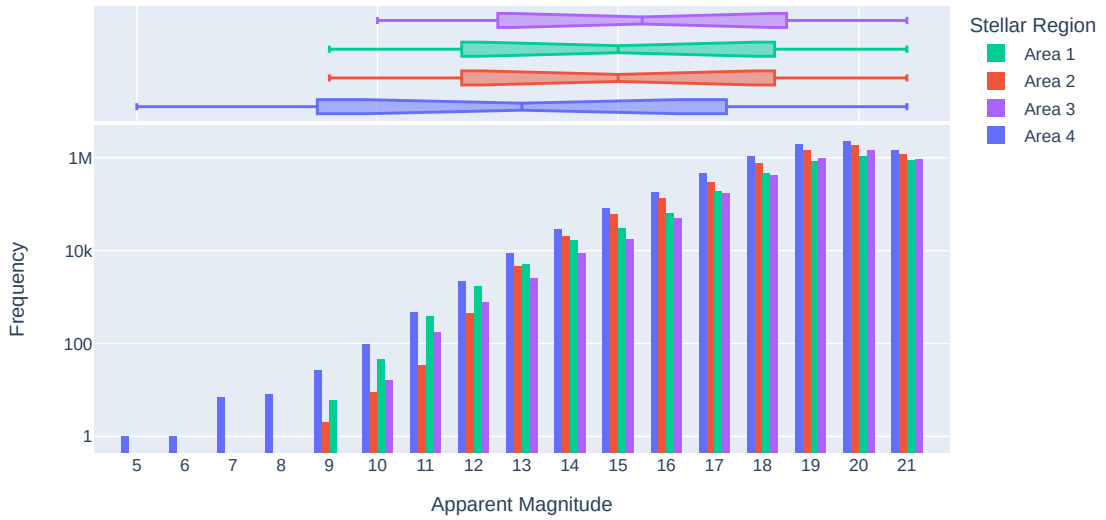


Figure 2.3: Distribution of Apparent Magnitude Across the Four Areas

A1 and A2 have very a similar distribution for their apparent magnitude. Area 1 contains more objects with a lower apparent magnitude (brighter stars) leading to a slightly broader tail being present. This may be explained by the fact that the data-set for Area 4 is larger (see Table 2.1).

2.1.2 RA, Dec, and Parallax

RA and Dec for some star are easy to retrieve. However, it is not so simple to retrieve the third dimension representing the distance from the Earth. By making use of trigonometry and angular shifts, astronomers were able to determine a value for distance referred to as a parsec. This works by virtue of our orbit around the sun. By measuring the relative position of distant stars at the opposite points of our planet's orbit, we may compare the position against some nearby star whose distance we know. The angular shift in position relative to that star provides us a parallax value. And $\frac{1}{\text{parallax}}$ provides us the distance in parsecs.

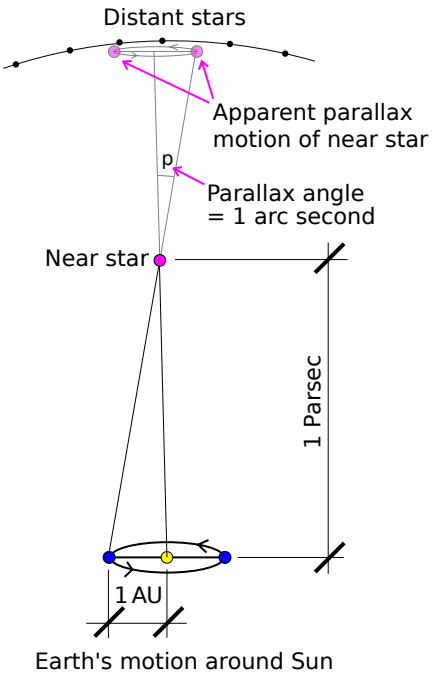


Figure 2.4: Depiction of Parallax Measurement [25]

TODO: **RA + Dec + Distance give you 3d coordinates of a sort. Important for the ant algorithm and clustering because that appear on top of each other may actually be many parsecs apart.**.

TODO: **Mention the negative parallax**.

The following comes from [24]: The parallax uncertainties are within the range of 0.04 mas for apparent magnitudes < 15. 0.1 mas for apparent magnitudes of 17. 0.7 mas for apparent magnitude of 20.

-Additionally, it is important to note that Gaia DR2 claims that for the stars with an apparent magnitude of 15 and higher that have mean errors for the parallaxes of 20-40 mas [26].
-For parallaxes, uncertainties are typically around 0.04 mas for sources brighter than 14 mag, around 0.1 mas for sources with a G magnitude around 17, and around 0.7 mas at the faint end, around 20 mag. The astrometric uncertainties provided in Gaia DR2 have been derived from the formal errors computed in the astrometric processing. Unlike for Gaia DR1, the parallax uncertainties have not been calibrated externally, i.e. they are known, as an ensemble, to be underestimated by 8–12% for faint sources (G> 16 mag) outside the Galactic plane and by up to 30% for bright stars (G<12 mag).[27]

TODO: **explain how it works based on the figure. and explain the content of this source [27]: the issue of the estimation of distances from parallaxes.**

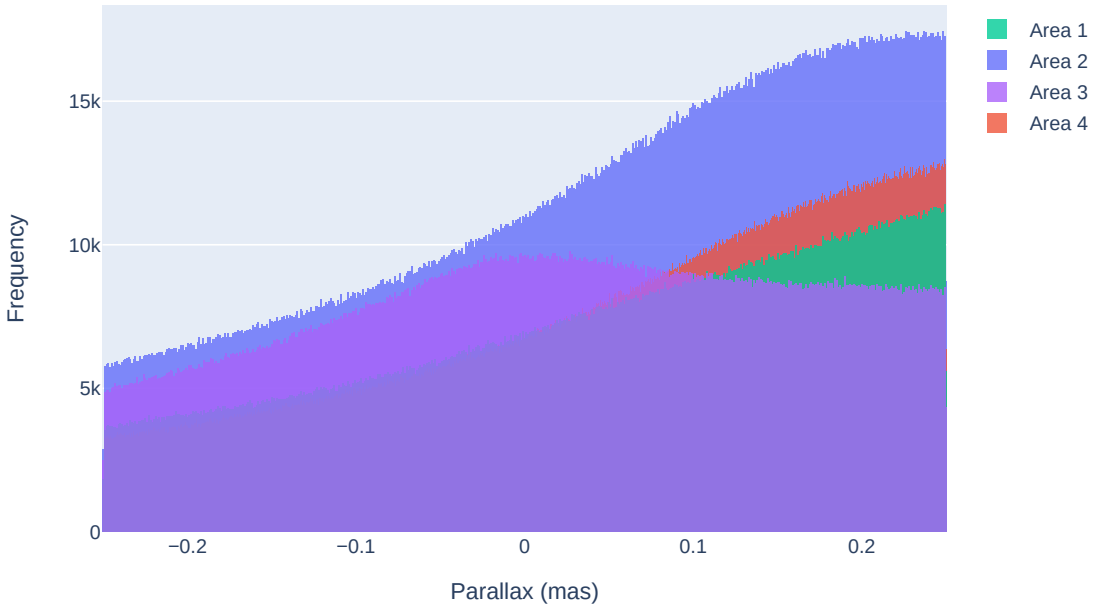


Figure 2.5: Distribution of Parallax Across the Four Areas

TODO: **explain the parallax chart.** Area 2 contains many more stars than the other regions. Stars that are closer too. Larger parallax values away from 0 indicates closer stars. TODO: **Deal with negative parallax?** TODO: **Add references to Mohammadi and that paper dealing with the error in these negative parallaxes.**

2.1.3 Proper Motion

It is also of interest to explore the spread of the data with respect to *proper motion of right ascension*, and *proper motion of declination*. Together they describe the angular shift that a star has over time. Essentially representing the drift that the star is facing. Stars that are drifting similarly are more likely to be related than stars that are drifting dissimilarly.

Histograms providing information on the distribution of PMRA and PMDec may be seen in Figure 2.7.

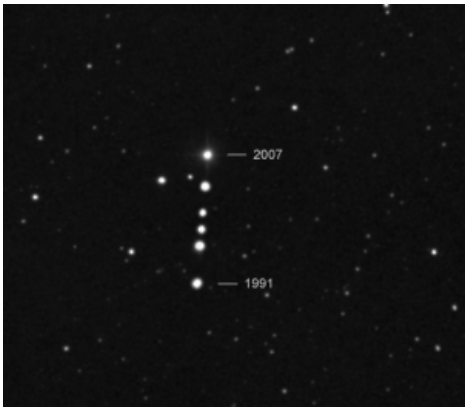
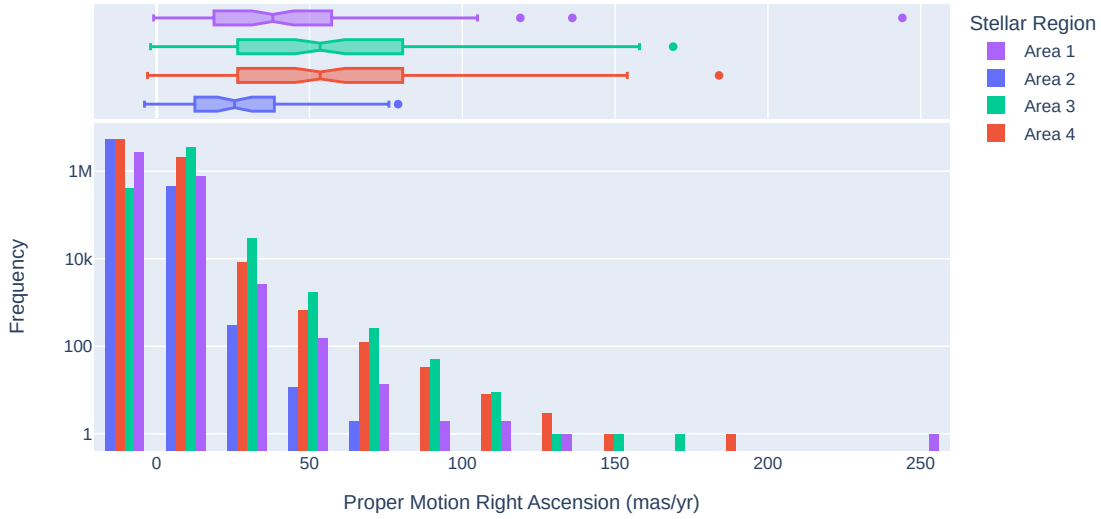


Figure 2.6: The proper motion of Barnard's Star between the years 1991 and 2007, an indication of its proximity to our own Solar System [28]

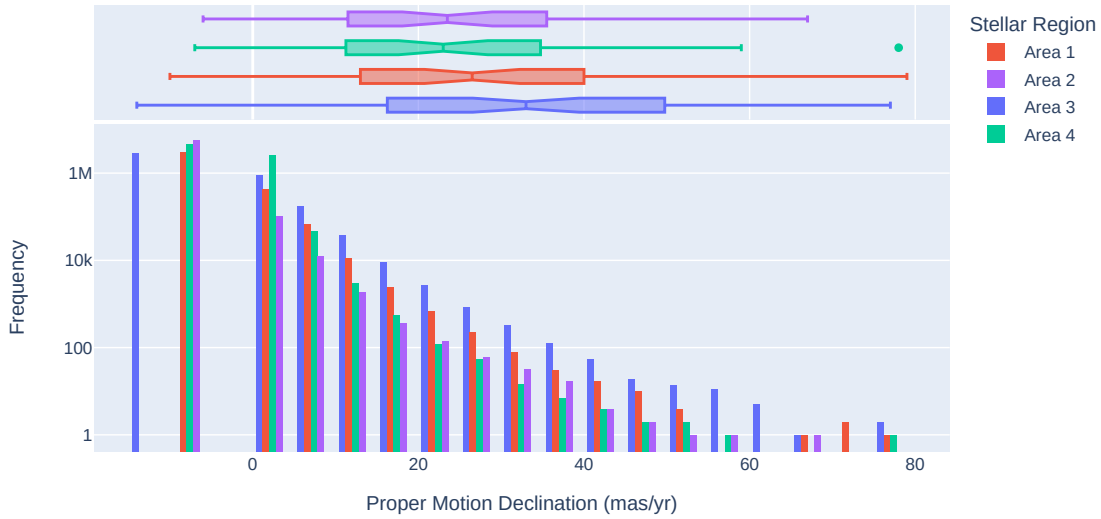
The following comes from [24]:

The parallax uncertainties are within the range of $0.06\,\mathrm{mas\,yr^{-1}}$ for apparent magnitudes < 15. $0.2\,\mathrm{mas\,yr^{-1}}$ for apparent magnitudes of 17. $1.2\,\mathrm{mas\,yr^{-1}}$ for apparent magnitudes of 20.

Figure 2.7: Distribution of Proper Motion Across the Four Areas



(a) Proper Motion In Right Ascension



(b) Proper Motion In Declination

TODO: **talk about the PMRA and PMDEC**

Chapter 3

Methodology

With each of the requisite parameters being described in Chapter 2, we may now begin to describe the process of identifying GCs. The central pipeline consists of three stages:

1. Blob-detection using the Difference of Gaussian algorithm.
2. Pheromone-based density mapping using the Ant Colony random-walk algorithm.
3. Gravitational clustering by using information from the previous stage to pool together related stars. The algorithm used for this stage was developed by the author for this paper.

However, as the Ant Colony algorithm functions better on smaller regions, it is first necessary to rasterize the data to create smaller windows to operate across. This pipeline then provides information, per raster, on the clusters contained within that raster. An overview of this whole process may be seen in Figure 3.1.

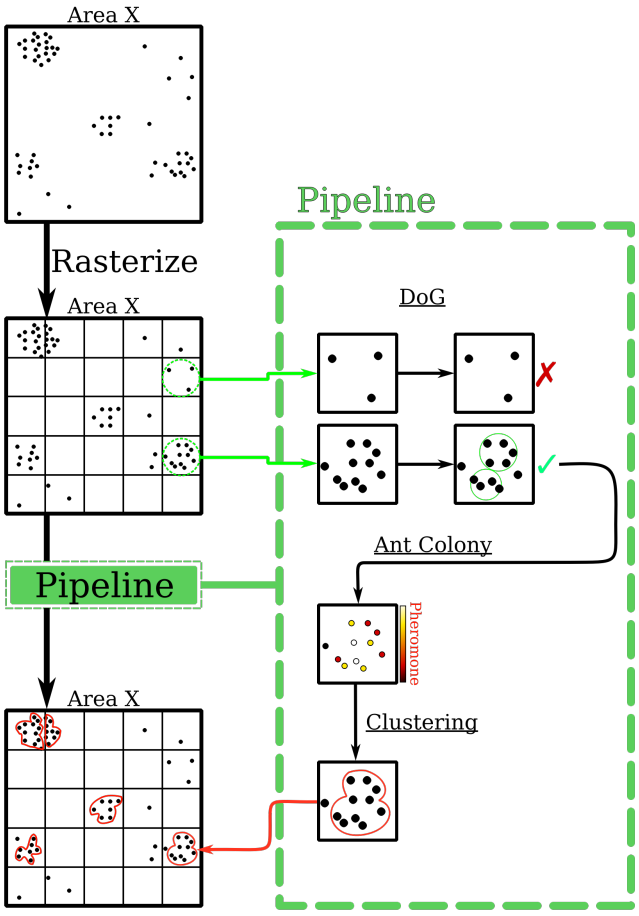


Figure 3.1: Overview of GC Identification

3.1 Rasterization

The process of rasterization simply involves the splitting of an area into smaller equally sized regions. Theoretically, this rasterization could occur across any set of parameters. However, in this instance the rasterization is applied across the equatorial coordinate system, and thus the splitting is based on the RA and Dec.

The largest known GC spans a RA × Dec of 1.5° × 1.5° [21] and thus this defines the minimum possible bound of the rasters. In, the the work of Mohammadi et al., rasters of 3.0° × 3.0° were used [8]. However, to provide leeway in the extraordinary case of two globular clusters of size 1.5° × 1.5° being next to each other, rasters of size 4.0° × 4.0° were chosen. Figure 3.2, shows an example of this rasterization scheme applied across the Small Magellanic Cloud in Area 3.

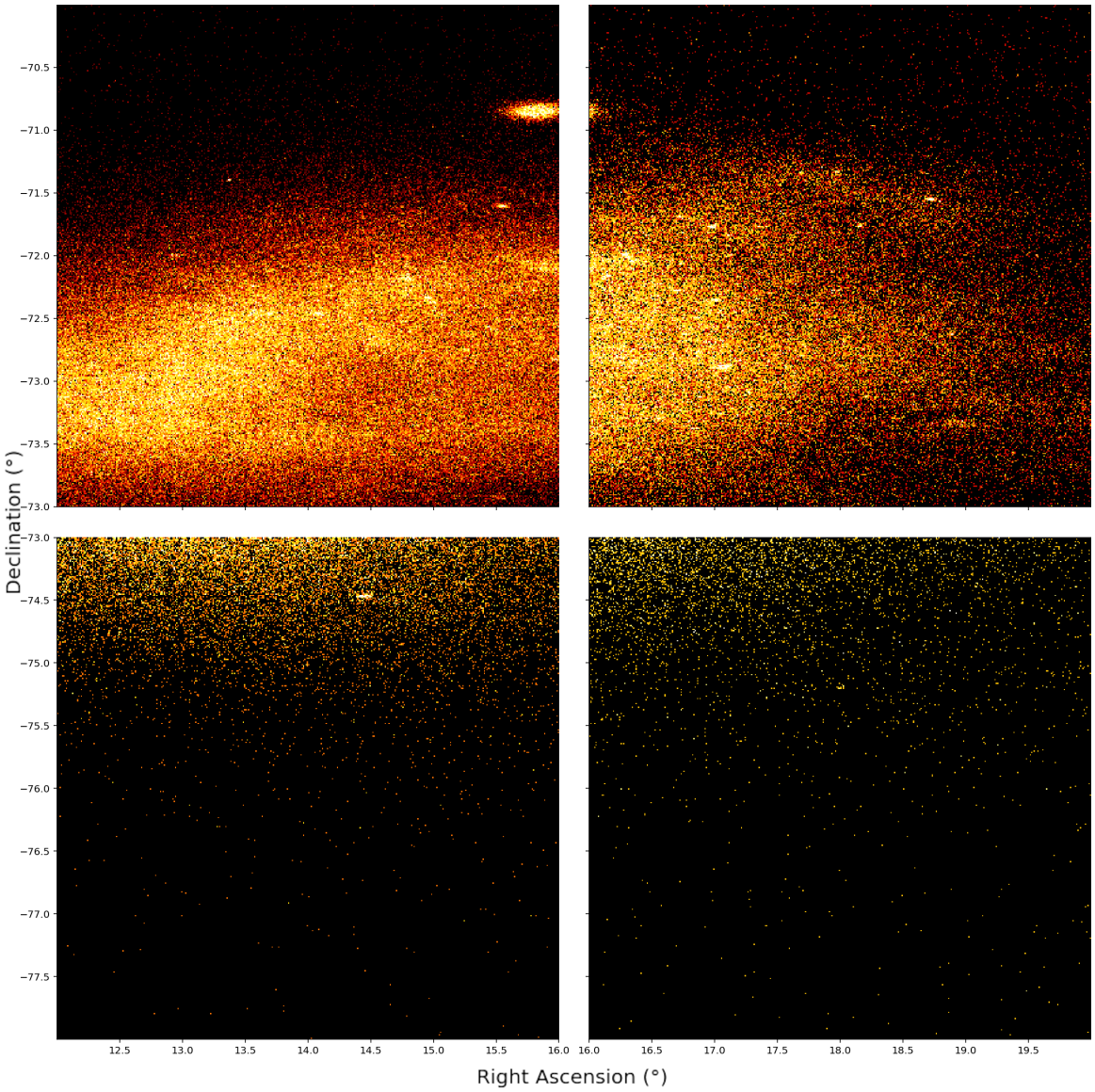


Figure 3.2: Example of Rasterization Across the Magellanic Cloud in Area 3

Rasterizing each area into smaller windows brings several benefits.

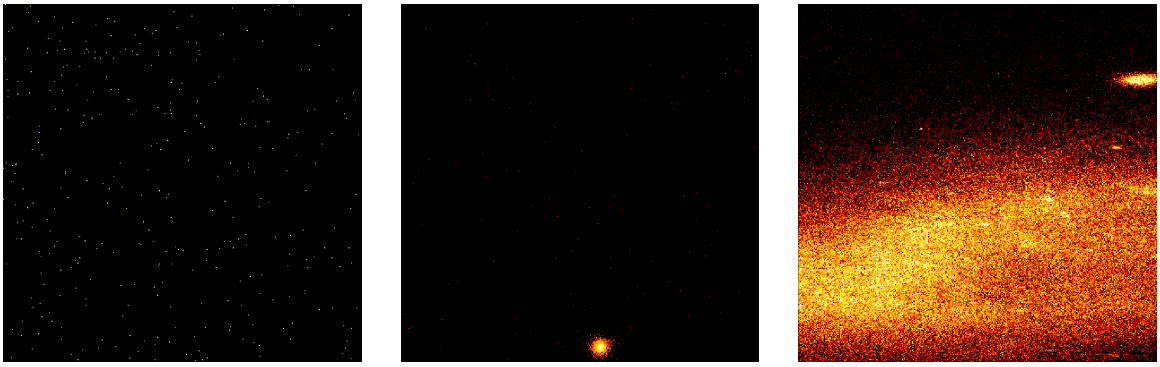
1. The Ant Colony algorithm functions better on smaller regions as it is more easily able to explore the state space (discussed in Section 3.3).
2. The sheer size of the data-set means that a significant amount of computer memory is required to completely load one of the four areas. The smaller regions generated by the rasterization significantly lowers the operating memory requirements.
3. More smaller regions are also more likely to be empty, and thus marked as such by the blob detection. This is useful and means that the more computationally expensive stages of the pipeline will have less data to process.

4. Finally, since each raster will be considered independently across the future stages of the pipeline, they may be operated on in parallel. This significantly reduces the amount of time needed to execute the pipeline.

However, this fixed rasterization scheme raises the issue of splitting a GC asunder at the raster boundary. This shortcoming and possible solutions are elaborated on in Section 6.4.

3.2 Blob Detection Using Difference of Gaussian

The rasterization results in many smaller regions, upon which, the central pipeline may then be run. However, the stellar distribution across these rasters is obviously non-uniform. Figure 3.3 displays the three different scenarios which dominate the description of the rasters.



(a) Dark Region (b) Dark Region with Bright Spot (c) Bright Region

Figure 3.3: Three Types of Rasters

Given, the significant time complexity of the Ant Colony algorithm and the clustering algorithm, it would be prudent to reduce the number of rasters to be processed. This may be done by filtering out the *dark* regions similar to the region presented in Figure 3.3a. The previous research performed by Mohammadi et al. made use Difference of Gaussian as a post-processing method [8]. However, as Difference of Gaussian is a blob detection technique, it may instead be applicable as a pre-processing step. Rasters for which DoG reports no blobs, would likely correspond to dark regions and may be filtered out.

3.2.1 Description

DoG is a blah ... It allows for blah ... Because of blah ...

Here's an example of part of blah ...

Convenient Api provided by Scipy. Basic description is as follows: Here's the kernel TODO: **This will just be the DoG equation with the right parameters filled in**, here's the algorithm applied with getting the coordinates at the end. TODO: **Add a footnote saying that under the hood Scipy seems to use another approximation using 1-dimensional kernels applied N times across N-dimensions instead of an N-d kernel.**

Mathematics

DoG is a feature enhancement algorithm

DoG manifests the SIFT property that from the paper in 2004 makes it elligable for use in blob detection.

Scale-invariance is important because the mechanism provides information on uniform clustering distributions in 2-D space in a way that facilitates continuous filtration on the N-Dimensional clustering property.

This then leads to the actual mathematical fundamental laplacian of the Gaussian which provides a scale-invariant mechanism. Brief Explanation then leads into DoG. TODO: **Showcase an image of the gaussian application to highlight at a high level what happens. Show Gaussian applied on some image at upper bound and lower bound, then show the subtraction**.

Subtract image at one scale with the image at another scale. Look for local extrema.

Talk only about Gaussian or about SIFT as well?

Reference the COSFIRE paper again because those circle-jerkers did.

TODO: **Figure out what the actual minimum size is based on the thresholds that were tested. The thresholds were tested and so that will be part of the results section.**

DoG is a feature enhancement algorithm for image data. In essence, it takes a grayscale image and produced blurred versions of that image. The blurring is made with respect to some property (which in our case is circularity). That image is then subtracted against the original which will cause the least matched regions to disappear and the most matched regions to remain.

The blurred images are obtained by transforming the original image with Gaussian kernels that use increased standard deviations [29]. The remainder of the difference between two successively blurred images are stacked up in a cube [29], which points out spacial information from between the range of blurring frequencies that are preserved in the images [30]. It can be viewed as a band-pass filter that discards all but a handful of spatial frequencies that describe features of interest within the original image [31]. As we are interested in circularity it is important that Gaussian Kernel is isotropic and behaves the same in any direction [29].

We convert the information of (RA, Dec, Magnitude) from each star within a raster, then convert it into a grayscale image representing (x,y,z). Where (x, y) are coordinates and z is a level of gray between black and white that represents the image.

In a DoG equation one finds the subtraction of Gaussian filters which explain the difference between an excitatory (positive) region and an inhibitory (negative) one [32]. A Gaussian filter is a normalized Gaussian Kernel which operates under some dimensionality. The Kernel [29] may be seen under Equation (3.1).

This pipeline uses an existing function from the library `SciKit` which uses `SciPys` 1D Gaussian kernel [33]. The equation of a DoG using a 1D Gaussian kernel is described here 3.2.

3.2.2 Mathematics

$$G_{ND}(\overline{x};\sigma) = \frac{1}{\left(\sqrt{2\pi}\sigma\right)^{N}} exp\left(-\frac{|\overline{x}|^2}{2\sigma^2}\right) \tag{3.1}$$

- TODO: **Explain N**
- TODO: **Explain x bar**
- TODO: **Explain x bar squared**
- TODO: **Explain that SciKit uses a 1-dimensional kernel and that you simply map your 2D image into 1D space.**

As the standard deviation (σ) increases so to does the resulting blurring of the image. This is because σ determines the width of the Gaussian Kernel and this referred to as the inner scale [31]. σ^2 is the variance. x is the windows of the input space from the image that gets convolved with the kernel.

$\frac{1}{\sqrt{2\pi}\sigma}$ represents the normalization constant and ensures that the average gray level of the image is maintained before and after the blurring with the kernel. This property is known as gray-level invariance [29].

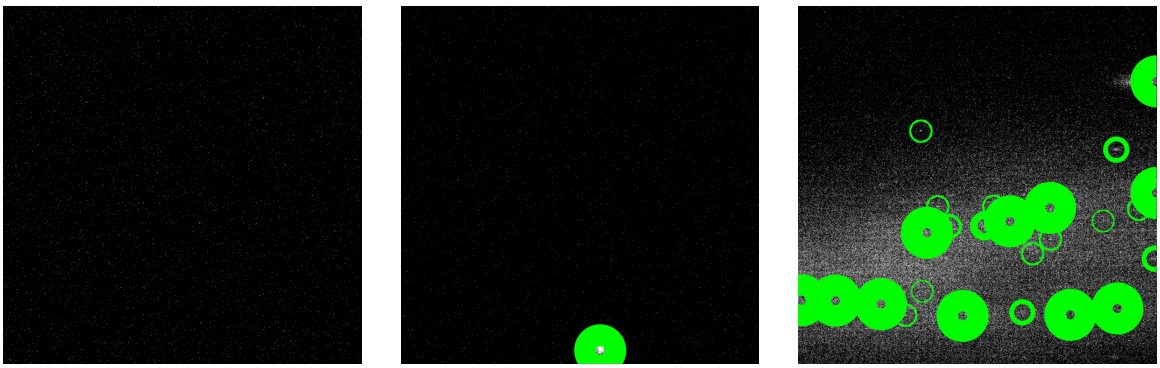
TODO: **Should I mention that the $2\pi sigma^2$ is a normalization step done differently in the SciPy implementation?**

$$DoG_{\sigma_1,\sigma_2}(x) = I * \left(\frac{exp(-\frac{x^2}{2(\sigma_1)^2})}{\sqrt{2\pi}(\sigma_1)^2} - \frac{exp(-\frac{x^2}{2(\sigma_2)^2})}{\sqrt{2\pi}(\sigma_2)^2}\right) \tag{3.2}$$

I is the image data.

Here is a matrix that represents the input gray scale image TODO: **rounded into 1 dimension**. Also for σ_1 and σ_2 there is a difference in the standard deviation, where $\sigma_1 < \sigma_2$.

3.2.3 Interpreting the Results of DoG



(a) Dark Region (b) Dark Region with Bright Spot (c) Bright Region

Figure 3.4: DoG Applied to the Three Types of Rasters

3.3 Ant Colony

After the dog filtration we are left with a number of raster to be tested for the existence of GCs. An exhaustive search of the regions would require TODO: **search every combinatorial group of stars and checking if the resulting group somehow conforms to being a GCs (a difficult computational task in its own right)**. The aim, instead, is to make use of the Ant colony random-walk algorithm to perform the combinatorial grouping through the leaving for pheromone. Then to use the clustering algorithm to pool those pheromone values into the potential clusters.

3.3.1 Bulletpoints

- the data is not used as an image but as a list of stars with their information on (ra,dec,mag,pmra,pmdec,distance, etc..)
- distance is computed and added.
- comment: Is the objective of the AC Alg. to detect *all* (rather: almost all) GCs in a region? then I don't see how it can be cheaper than exhaustive search. - The formal task spec for the AC is badly limiting my understanding - MUST be given.
- comment: what is the range of γ? is it used as a "decay rate"? what is the range of w's? If γ is a model parameter - how is it determined / optimized?
- implementation might highlight how it works in detail
- Question to answer in my report is about the constant C in the `update_pheromone` section.

3.3.2 Description

With the search space reduced through the DoG filter, we are left with candidate regions which demonstrate the property of circularity that GCs hold. Each region contains a set of stars belonging to that region. For each region we can start applying a random-walk algorithm.

TODO: ***Random-Walk Algorithm Background***

- [34]:
- A random-walk is known as a random process which describes a path including a succession of random steps in the mathematical space.
- Random-walks can be used to analyze and simulate the randomness of objects and calculate the correlation among objects, which are useful in solving practical problems.
- In the mathematical space, a simple random-walk model is a random-walk on a regular lattice, in which one point can jump to another position at each step according to a certain probability distribution.
- When it is applied on a specific network, the transition probability between nodes is positively relevant to their correlation strength. That is, the stronger their association is, the greater the

transition probability is. After enough steps, we can obtain a random path that can describe the network structure.

- examples: PageRank, lazy random-walk (LRW), Quantum walks (discrete time based algorithms and continuous time based algorithms),
- Compared with other alternative approaches, random-walk based algorithms can incorporate a great deal of contextual information. As same as collaborative filtering, link prediction and recommender system also aim to calculate the k-most-close nodes for the selected node. Hence, random-walks are also effective in link prediction and recommendation system.

Visualization

For this data the algorithm aims to identify the groups of stars that are densely packed among these candidate regions. Even with the previous candidate filtration, the state space is still likely to be huge, hence the use of a random-walk algorithm. An exhaustive method would simply take too long to compute.TODO: **mention that this method is also used by the comparison work of Mohammadi [35]** The random-walk algorithm that is released on the remaining regions is a swarm intelligence method called the ant colony algorithm TODO: **Change the citation [35] to a better source (background info on random-walk algorithm)**.

This swarm intelligence method makes use of a characteristic learned from ant colonies. Ants walk around in search for food while leaving behind a chemical substance called pheromones TODO: **REPLACE [35]**. These pheromones serve to guide other ants towards some reward. When a reward is visited by many ants the path to it will contain higher concentrations of the pheromones. Thus, ants are more likely to follow paths with higher pheromone concentrations TODO: **REPLACE [35]**. Interpreting these pheromone values with respect to GCs would mean that the levels of pheromone values within clusters are high and the levels on the paths between two clusters are low. This is practical for understanding what stars belong to a cluster and to provide distinction between overlapping clusters. This also provides information on the locations with the highest densities.

TODO: **work the text below into the implementation description** Before the ants start their random-walk, the feature vector $\mathbf{f}$ containing a pheromone for each star gets initialized. The outer loop runs the random-walk simulation N_{iter} amount of times. Each time it executes three things: The initialization of vector $\mathbf{N_v}$ keeping track of the number of visits for each star, the random-walk for N_a ants and the update of $\mathbf{f}$ by equation (3.10). The random-walk for each ant consists of the selection of a random starting position (for example the node indicating $\mathbf{x}_i$) and a series of steps depicting a path. On this path, the next step for the ant is determined in a random manner biased by the distribution of pheromones in its neighborhood. This step is governed by equation (3.4). When the ant takes the step it is logged in the vector $\mathbf{N_v}$.

Implementation Description

All files that DoG would approve for further inspection get run through the Ant algorithm. Where it returns a filled in pheromone column with for each star a pheromone value attached that expresses how dense how popular the area point (=star) was for the 'ants' to pass. Hence how dense the location is with stars. (as the ants try to take the shortest path, then where they took the shortest path is where the stars are closest together, and as a result you should see the highest pheromone value there as it was valued highly with pheromone values)

The Ant Algorithm as implemented consists of three loops. Where the first iterates an N_{iter} amount of times over the data (equivalent to redoing the experiment and getting accurate 'averaged' pheromone values). In the second loop all ants (N_a) are set loose randomly in the described landscape (=raster with regarding stars). The third loop describes the process of ants walking around a number of steps (N_s), taking the best paths laid out for them based on the math. When all the steps were taken the pheromone values are updated again

3.3.3 Algorithm

Algorithm 1 Ant Colony [35]

Output: The pheromone vector: $\mathbf{f} = [f_1, f_2, \ldots, f_{N_{\text{stars}}}]$
  $\mathbf{f}^0 = [0, 0, \ldots, 0]$ ▷ Length of N_{stars}
  for $t = 1, \ldots, N_{\text{iter}}$ **do**
    $\mathbf{N_v} = [0, 0, \ldots, 0]$ ▷ Length of N_{stars}
    for $a = 1, \ldots, N_a$ **do**
      $x_i \leftarrow$ Randomly select an initial position for the ath ant
      $\mathbf{N_v}[i]$ += 1
      for $s = 1, \ldots, N_s$ **do**
        $x_j \leftarrow$ Select the ant's next position using Eq. (3.4) ▷ This uses x_i and $\mathbf{f}^{t-1}$
        $\mathbf{N_v}[j]$ += 1
    Update $\mathbf{f}^{t-1}$ to $\mathbf{f}^t$ using Eq. (3.10) ▷ This uses t, $\mathbf{f}^{t-1}$, $\mathbf{N_v}$, and N_a
  return $\mathbf{f}^{N_{\text{iter}}}$

TODO: **Mention that the initial random selection is a uniform random distribution**

In this algorithm N_{iter} indicates the number of times that an ant randomly starts walking, $\mathbf{N_v}$ is the vector indicating the number of visits it received for each star (in the line $\mathbf{N_v}[j] = \mathbf{N_v}[j] + 1$ that number gets increased by 1 for the concerning star), N_a is the number of ants and N_s specifies the number of steps that an ant walks in each iteration.

Choice of constants explained:

Their default values are:

- $N_{iter} = 5$
  As the ants get initialized randomly in a uniform way, you need to give them the opportunity to test the parallel universes where they were initialized at a different location to give the varied outcome, possibly covering different dense regions that previously were not touched; because the ants would not make that jump as no star is attached to any of the stars neighborhoods the ants covered. As the ant location initialization is random, it is good to run it a couple of times (thought 5 times would cover it) instead of having more ants because that would affect not just the randomness of missing covering an area but other parts of the algorithm as well. Running Ant a large amount of times could take a long time so the choice of running it 5 times should be enough. However, it is unclear what else this constant might influence.
- $N_a = 30$
  Having too little ants one expects they would not cover a lot of ground and you wouldn't be able to see the effect of their influence of each other on the valuation of the stars in a given space. Having too many ants one expects that too many stars will be visited in the beginning which then influences the outcome sufficiently to not see clearly enough what the dense regions are. With the initial testing on some A1 rasters containing GCs this value seemed to result on something that would find the dense areas of the cluster. However, an influential number to the effectiveness of N_a might be the amount of stars that are present within the raster.
- $N_s = 1000$
  You should not have the ants walk around forever as they most likely will be walking in circles in the dense spots eventually, so the question is: at what point have they been walking around enough to paint a high pheromone map round these high density spots. I expect that with a lot of steps taken, it will concentrate the pheromone values in the dense regions in such a way that there will be multiple small dense regions to be seen. To give an insight into raster sizes of A1 it can go from about 5000 to 30 000 stars. I expect that one can get to the approximate center of a dense area in about 400 to 800 steps, so I picked N_s to be 1000.

3.3.4 Mathematics

After the ant is inserted at a random location it needs to know what options it has to go next (i.e. all neighbors in the neighborhood of $\mathbf{x}_i$). After the neighbors are known by their distance calculation we need a choice which are the calculations of the transition probabilities per star. To calculate these probabilities we need the normalized pheromone values of all neighboring stars and the weight values between $\mathbf{x}_j$ and the neighboring stars $\mathbf{x}_k$. After all steps have been taken, the pheromone

values need to be updated based on how often the stars have been visited this round an previous information.

Euclidean Distance for Calculating Neighboring Stars

One piece of the information that is needed is the Euclidean distance between stars. This is used to determine the nearest neighbors for each star. The intent is to create a list containing n of the nearest stars to the ith star. If the space were limited to a 2D plane (e.g. just RA and Dec) then computing the Euclidean distance is easy. One simply takes the difference between the RA and Dec values of the star and applies the Pythagorean theorem. However, to operate in 3D space one has to consider the third axis.

TODO: **Mention KDTree, especially because it is used in the clustering algorithm for speed.**

Third Axis: Depth The data-set does not directly report the depth. Instead it must be computed from the parallax (p) or the relation between the apparent and absolute magnitudes. For the stars that have a parallax value reported in the data-set, the distance (d) in parsecs (pc) may be calculated by taking the inverse of the parallax.

$$d = \frac{1}{p} \tag{3.3}$$

With these RA, Dec, and depth, one can then calculate the Euclidean distance between two stars $d(\mathbf{x}_i, \mathbf{x}_j)$.

TODO: **put the info on accuracy somewhere: maybe with the explanation of parallax** Thanks to the unprecedented performances of the Gaia astrometry, proper motions for these satellites could be measured with a typical accuracy of a few tens of μ as yr-1, which corresponds to an accuracy of 0.5-2kms-1 in tangetial velocity for a cluster located at 10 kpc from us. [36]

Transition Probability

To learn the **transition probability** (P) for the t-th iteration from the data point $\mathbf{x}_i$ to $\mathbf{x}_j$ we use equation (3.4) [35], which can be found below.

$$P^{(t+1)}(\mathbf{x}_i, \mathbf{x}_j) = \frac{(w(\mathbf{x}_i, \mathbf{x}_j))^{\gamma} \left(\hat{f}^{(t)}(\mathbf{x}_j)\right)^{1-\gamma}}{\sum_k (w(\mathbf{x}_i, \mathbf{x}_k))^{\gamma} \left(\hat{f}^{(t)}(\mathbf{x}_k)\right)^{1-\gamma}} \tag{3.4}$$

Here, γ controls the effect that the pheromones have on the resulting step taken. $w(\mathbf{x}_i, \mathbf{x}_j)$, denotes the weight of the TODO: **edge** that represents the path from $\mathbf{x}_i$ to $\mathbf{x}_j$. The weights get initialized (3.3.4) before the algorithm is run. Each ith star ($\mathbf{x}_i$) has its sorted collection of weights between itself and their nearest neighbors ($\mathbf{N}_i$). $\hat{f}^{(t)}(\mathbf{x})$, is the normalized (3.3.4) amount of pheromone on the respective star, $\mathbf{x}$, after t iterations.

Choice of constant

$\gamma = 0.9$

Weight Initialization

GCs are an agglomerate structure. When identifying them, it is necessary to consider the attributes of individual stars as well as the relationship between stars. To this end, the stars are represented on a graph, with the stars representing the nodes and the relationships between the stars represented as paths. On the paths a weight value is encoded which represents the similarity between pairs of stars in one value. Neighboring stars within the same cluster are expected to have a higher similarity than stars outside of the cluster. These weight values are either computed by a kernel function or through Principal Component Analysis (PCA) [35]. They require a one time initialization and are then used for calculating the transition probabilities in equation 3.4.

In the case that the weight is based on kernels the **Gaussian function** [35], is applied. The information incorporated for the kernel weight initialization is only based on the Euclidean distance,

and may be seen in equation (3.5).

$$K(\mathbf{x}_i, \mathbf{x}_j) = exp\left(-\frac{d(\mathbf{x}_i, \mathbf{x}_j)}{2\sigma^2}\right) \tag{3.5}$$

Here σ is the standard deviation and $d(\mathbf{x}_i, \mathbf{x}_j)$ is the Euclidean distance.

In the case that the weight is based on **PCA** then the method considers a geometric shape of a manifold to which the noise level of a data point can be interpreted as the distance that the data has to the manifold [35]. The aim of this is to encourage a random-walk staying close to the manifold [35]. The information incorporated for the PCA weight initialization is based on certain parameters that may be chosen from that data-set. These parameters would likely consist of the Euclidean distance plus RA, Dec, $PMRA$, and $PMDec$. This is because these five parameters encode the position of a star and its motion. Since stars in a GC move together neighboring stars within the cluster will have similar positions and motion.

PCA is used to find a vector $\mathbf{V}$ containing the set of eigenvalues and eigenvectors, i.e. $\{(\lambda_k, v_k)\}_{k=1}^{D}$ where D is the number of features considered in the PCA (in this case six: RA, Dec, $PMRA$, $PMDec$, $distance$, $magnitude$) [35]. The d-dimensionality of the manifold would make a separation of d planes. For which one than estimates the distance of the tangent space with formula (3.6) [35].

$$\Delta_j = ||\mathbf{x}_j - (\mathbf{V}\mathbf{V}^T(\mathbf{x}_j - \mu) + \mu)|| \tag{3.6}$$

Here $\mathbf{V} = [v_1, v_2, ..., v_d]$, $\mathbf{x}_j$ is the jth neighbor of $\mathbf{x}_i$ from the set of neighbors $\mathbf{N}_i$ and μ is the mean position of the stars in $\mathbf{N}_i$, calculated by equation (3.7).

$$\mu = \frac{1}{|\mathbf{N}_i|}\sum_{\mathbf{x}_j \in \mathbf{N}_i} \mathbf{x}_j \tag{3.7}$$

With Δ_j known we can initialize the weight values.

$$w(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1 - \frac{\Delta_j}{a} & a \geq \Delta_j \\ 0 & a < \Delta_j \end{cases} \tag{3.8}$$

Here, $a = max(\sigma_j)$ has the purpose of ensuring the weight values are positive [35]. It is important to note that when computing the weights using PCA that $w(\mathbf{x}_i, \mathbf{x}_j)$ does not need to be $w(\mathbf{x}_j, \mathbf{x}_i)$. As for each star, $\mathbf{x}_x$, has its own neighborhood of nearest stars $\mathbf{N}_x$ on which the calculation of Δ_x is based. When $\mathbf{N}_x$ is different μ is different and so is Δ_x.

Pheromone Normalization

To assure that the weights w and pheromones $\mathbf{f}$ are within the same scale, equation (3.9) is used for **normalizing the pheromones** of the neighborhood of $\mathbf{x}_j$ [35].

$$\hat{f}(\mathbf{x}_j) = \frac{f(\mathbf{x}_j)}{\sum_{\mathbf{x}_k \in \mathbf{N}_i} f(\mathbf{x}_k)} \tag{3.9}$$

$\mathbf{N}_i$, represents the set of the neighbors of the i-th star, $\mathbf{x}_j$ and $\mathbf{x}_k$ are the data points for the j-th and k-th star and $f(\mathbf{x})$ denotes the amount of pheromone on the regarding star.

3.3.5 Update Pheromone Levels

Equation (3.10) **updates the pheromone level** [35].

$$f^{(t+1)}(\mathbf{x}_i) = c \times \frac{\sum_{a=1}^{N_a} N_i^a(n)}{N_a \times n} + (1-\rho)f^{(t)}(\mathbf{x}_i) \tag{3.10}$$

Here, c indicates the amount of pheromone that is added to a point per visit, N_a is the number of ants, $N_i^a(n)$ is the number of times the a-th ant visits $\mathbf{x}_i$ over n steps in the $(t+1)$-th iteration and ρ is the hyper-parameter that controls evaporation of the pheromone, which happens because the recency of the pheromone update is important [35].

Choice of Constant Explanation:
$\rho = 0.1$

3.3.6 from Ant to Clustering section

ending: things that have not been mentioned and what is important in regard to what comes

3.3.7 Interpreting the Results of the Ant Colony Algorithm

The results of the algorithm are an updated pheromone vector **f** which contains adjusted pheromone values for each star in the given window. The pheromone values within clusters will be high and the values outside of (and between clusters) will be low. This provides a boundary separating a cluster from other stellar structures and is practical for providing a distinction between overlapping clusters. However, the actual clustering still has to be performed. The clustering algorithm can make use of the collected pheromone values and the available parameters. One possible technique that may be used is the use of spectral clustering[35].

3.4 Clustering

Though the pheromone value is technically computed using the coordinates of the stars relative to each other, a specific value within some clustering in a raster would be indistinguishable from that pheromone value within the some clustering within that raster. Thus, the clustering algorithm must take into account the coordinates of the stars being clustered while being guided by the pheromones of the stars.

The Ant Colony Algorithm has encoded the plethora of information related to the stars *RA* *Dec*, *PMRA*, *PMDec*, *magitude*, and *distance* TODO: **fix the rendering of the text here to be consistent across the whole project** into the pheromone trails that have been left. The next step is to then cluster these results such that stars with high pheromone values are grouped together. However, while the raw pheromone values do provide information on the density (based on the strength of the visitations by the ants), it is quite possible for two stars in different but equally dense regions to have the same pheromone values. Thus, the clustering algorithm must also make use of the position of the star coupled with its pheromone values to determine which cluster a star may belong to.

Research on a variety of clustering algorithms (such as those available within SciKit) presented no obvious candidate. Thus, this section describes an algorithm that was developed specifically for this pipeline. Firstly, to restate the issue, there are a number of points distributed in 3-dimensional space with a (pheromone) parameter attached to them that determines the strength of the attraction they feel to other stars. This problem formulation reveals the similarity of this problem description to the basic description of gravitational attraction.

3.4.1 Description

The basic intuitive description of the algorithm is that a set of clusters should be grown from the highest pheromone values outwards. These cluster have a *center-of-gravity* based on their pheromone values. This center-of-gravity coupled with the overall pheromone value (mass) of the cluster may be used to determine the strength of the attraction of the cluster with any other star. The method for computing the force of this attraction may be seen in equation TODO: **blah**.

TODO: **Equation here**

Then the star with the next highest pheromone value which is still available may be used as the starting point of the next cluster. This continues in the fashion until all stars have been clustered. It is important to note that the 0 pheromone value stars (those totally un-visited by the ants) do not contribute to increasing the gravitational attraction force of a cluster (this is used as an optimization by splitting the set of stars and performing a simpler computation to group the zero stars into the generated clusters).

However, there is an edge case to consider. Stars with a 0 pheromone value (those that were unvisited) will not result in a 0 gravitational attraction and thus never be clustered. Since, there is no information in this clustering other than the positions of these stars the mechanism is as follows. These stars coordinates are evaluated and are inserted into the clusters that have already been computed based on their centroid and the ellipsoid that is formed across the diameters for there values across RA, Dec, and Distance. A star that is within some cluster is simply inserted into the first cluster that contains it. This, also affords an optimization, where the set of stars are partitioned into non-zero and zero pheromone values with the initial clusters only being computed based on the stars with non-zero pheromone values. TODO: **the case were a star may actually lie across**

the bounding of 2 clusters is described in the shortcomings TODO: **keep this? and if so add ref to section**.

Finally, the set of generated set of clusters may then be filtered by various criterion defining the minimum requirements describing the target clusters. For GCs, this minimum requirement is the number of stars contained within the cluster.

Thus, the two parameters that may be varied with the execution of the algorithm are minimum number of stars that must be present within a cluster and the minimum attractive force. The smallest stellar clusters contain at least 100 stars and thus this is used as the minimum bound [37]. The minimum attractive force was determined through trial-and-error with a good value being determined at approximately 50. TODO: **rework this slightly and note that the value for the minimum attractive force likely varies based on the total mass of pheromone (as evidences by different clustering results with varied parameters for the ant colony algorithm)**

With all these steps in tow, the results are a set of clusters each of which contain a set of stars. These values may then be plotted to visualize the stars within a raster, highlighted those that are unclustered and those that are contained within differing clusters. See results.

3.4.2 Algorithm

Algorithm 2 Pheromone Clustering

Initialization: The *stars* with their pheromone values ($\mathbf{f}$) computed by the Ant Colony
Output: The clusters present in the stars according to their pheromone values

function PHEROMONE_CLUSTERING(stars, $F_{\text{min\_attraction}}$)
- $(\text{stars}_{\mathbf{f}=0}, \text{stars}_{\mathbf{f}\neq 0}) \leftarrow$ Partition stars by pheromone value($\mathbf{f}$)
- $\text{clusters}_{\text{initial}} = \text{NON\_ZERO\_CLUSTERING}(\text{stars}_{\mathbf{f}\neq 0}, F_{\text{min\_attraction}})$ ▷ See Alg. 3
- $\text{clusters}_{\text{all}} = \text{ZERO\_CLUSTERING}(\text{clusters}_{\text{initial}}, \text{stars}_{\mathbf{f}=0})$ ▷ See Alg. 4
- $\text{clusters}_{\text{filtered}} \leftarrow$ Filter out *cluster*s from $\text{clusters}_{\text{all}}$ that contain less than $N_{\text{min\_stars\_in\_GC}}$
- **return** $\text{clusters}_{\text{filtered}}$

Algorithm 3 Non-Zero Clustering

Initialization:
Output:

function NON_ZERO_CLUSTERING($\text{stars}_{\mathbf{f}\neq 0}$, $F_{\text{min\_attraction}}$)
- $\text{stars}_{\text{left}} = \text{stars}_{\mathbf{f}\neq 0}$
- clusters = {} ▷ Create an empty set of clusters
- **while** $\text{stars}_{\text{left}}$ is not empty **do**
  - Create an empty cluster, C
  - star $\leftarrow$ Extract the star with the greatest pheromone value from $\text{stars}_{\text{left}}$
  - Insert star into C
  - cluster_changed = `true`
  - **while** cluster_changed **do**
    - cluster_changed = `false`
    - **for all** star $\in \text{stars}_{\text{left}}$ **do**
      - cog $\leftarrow$ Compute COG of cluster,c, according to Eq. **??**
      - distance $\leftarrow$ Compute Euclidian distance between cog and position of star
      - pheromone_mass $\leftarrow$ Compute pheromone mass of cluster, c, according to Eq. **??**
      - star_pheromone $\leftarrow$ Extract pheromone value for star
      - $\text{attraction} = \dfrac{\text{star\_pheromone} \times \text{pheromone\_mass}}{\text{distance}^2}$ ▷ Based on Eq. **??**
      - **if** attraction $\geq F_{\text{min\_attraction}}$ **then**
        - Remove star from stars
        - Insert star in cluster, C
        - cluster_changed = `true`
  - Insert cluster, C, into clusters
- **return** clusters

Algorithm 4 Zero Clustering

Initialization:
Output:

```
function ZERO_CLUSTERING(clusters_initial, stars_f=0)
    clusters_all = clusters_initial
    for all star ∈ stars_f=0 do
        clusters_closest ← clusters_all sorted by Euclidian distance from centroid of cluster to star
        for all cluster ∈ clusters_closest do
            if cluster covers star based on Eq. (??) then
                Insert the star into the cluster
                break                                          ▷ Out of inner loop
    return clusters_all
```

TODO: **Stipulate the cluster covers star equation**

3.4.3 Mathematics

Pheromone Mass

We consider the pheromone mass to be ....

Center of Gravity

Attractive Forces

Centroid

3.4.4 Interpreting the Results of the Clustering

Chapter 4

Results and Findings

4.1 DoG

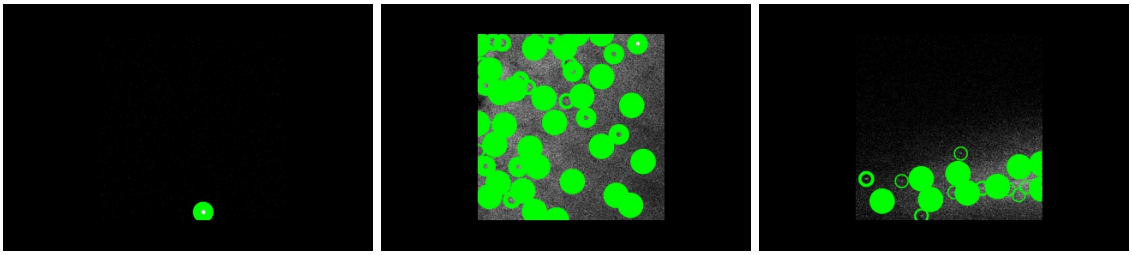
DoG is a blob-detection technique and in this data blobs are expected to be agglomerate stellar structures among which GCs. In Figure 4.1, you can see that this technique finds several blobs in the different areas. The green circles indicate where blobs were found, a small/thinner circle represents a smaller blob, and a larger/fuller circle illustrates a larger blob.

A1s Figure 4.1a finds only one blob, which is the globular cluster NGC 5024. In 4.1d you see a raster with a region of RA(228.0 – 232.0) and Dec(-2.0, 2.0). Visibly present are two blobs which are both GCs. Palomar 5, which, is clearly found in the center of the raster, and M5, with its center coordinates of (RA(229.6), Dec(2.02)) in another raster but with a large enough diameter such that its stars are also partially within the raster of Figure 4.1d. As the blob is so on and near the edge we only see half a green circle. You could see this as an indication that the raster might need to be made at a different cutoff point before running it through the next part of the pipeline. In the circumstances found in A1, where there are apart from the GCs not many stars present overall, DoG seems to work perfectly as a GC detection method. However, in areas that have a lot more stars overall, it does not only see GC's as blobs.

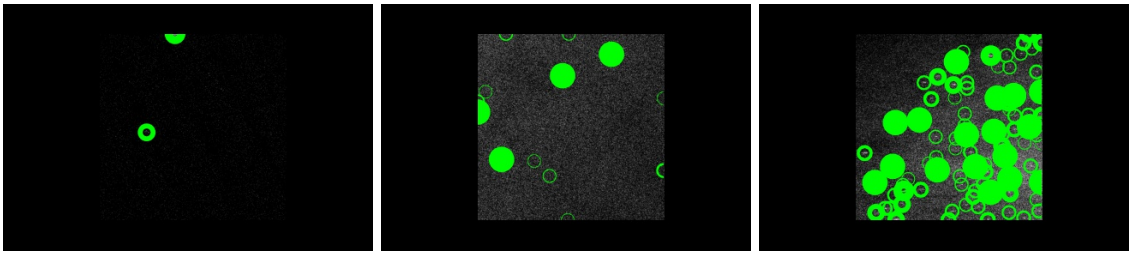
A2 has been rasterized and run DoG on twice, once with $4.0° \times 4.0°$ raster parameters and once with $2.0° \times 2.0°$ raster parameters. Figures 4.1b and 4.1e both show a variation of blobs that were found, among which the GC M71. The $2.0° \times 2.0°$ is one fourth the size of the $4.0° \times 4.0°$ raster making it a way smaller and easier area to look at as the number of blobs is less.

A3 is an area with an occasional extreme amount of stars in one large location, good examples of this situation are the Magellanic Clouds. DoG finds many blobs here, which is not unusual as there are many stellar objects present in these clouds. Many of the objects seem to be bright stars or galaxies. In 4.1c there are no GCs but it is a busy area where it finds many blobs. In 4.1f it finds an enormous amount of blobs. This raster contains four blobs (NGC 1696, NGC 1756, NGC 1786, and NGC 1795), however, it is not distinctly clear which ones in the image they could be, as so different blobs are found here as well.

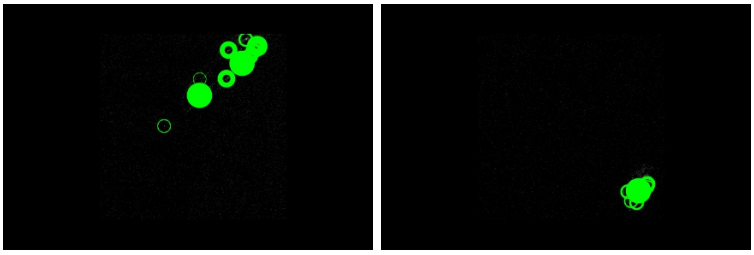
A4 has two large galaxies (which can be seen in Figures 4.1g and 4.1h). The galaxies themselves as a whole are not seen as blobs, however, the large singular parts that make up these galaxies, identify as blobs. For more detail on A4 see Table 4.3.



(a) In A1: NGC 5024 (b) In A2 (4by4): M71 (c) In A3: Small Magellanic Cloud



(d) In A1: Palomar 5 + M5 cutoff (e) In A2 (2by2): M71 (f) In A3: Large Magellanic Cloud



(g) In A4: Andromeda Galaxy (h) In A4: Triangulum Galaxy

Figure 4.1: Interesting examples of the blobs found by DoG

4.1.1 Remaining Rasters

The DoG filter was run across all the rasters of the area with varying thresholds. The number of remaining rasters depend on the threshold that was set regarding the intensity of the blob-detection on the data in the raster. In Table 4.1 the number of remaining rasters, from each area, are shown for the different thresholds.

Table 4.1: Rasters remaining after the execution of DoG at Varying Thresholds

Area	Threshold = 0.5	Threshold = 0.2	Total Number of Rasters
Area 1	7	8	512
Area 2 (4by4)	7	9	12
Area 2 (2by2)	16	28	35
Area 3	23	39	285
Area 4	19	60	120

When you set the threshold too low you will detect one or more blobs in every raster. For A1 the threshold was set to 0.1 and it found blobs in all 512 rasters. Then the intensity of the detection is too high, and so single stars are detected as blobs.

4.1.2 A1, A2, and A3 - Filtering Rasters with Known GCs

A1 has 7 out of 12 known GCs detected within a threshold of 0.2. Additionally, it detects blobs in another raster that does not contain a known GC. This raster is in the region with a starting RA and Dec (RA-start, Dec-start) of (152.0, 10.0) and a range of 4.0° × 4.0°, where it finds a blob at approximately *RA*: 152° and *Dec*: 12° at this position there is a dwarf galaxy. When searching between the thresholds of 0.2 and 0.1, we find that: In the case that the threshold=.15 it mostly does find the known GCs. The known GCs that are only detected then are for example: Koposov 1 which is a low-luminosity globular cluster [38], Palomar 3 which is one of the most distant GCs [39] with a magnitude of 14.26 and a distance of ∼96 kpc, and Palomar 4 which has an even larger magnitude of 15.65 and distance of ∼109 kpc [21] . They are not detected by DoG because the

size of the blob/cluster in these cases is quite small as they are located farther away or are of lower luminosity.

A2 has two similar outputs for the different types of rasterization bounds. They both detect the one GC present in thais area. A2 (4x4) detects blobs in 7 rasters with a threshold of 0.5 and 9 rasters with a threshold of 0.2. A2 (2x2) finds that in 16 rasters one or more blobs get detected when the threshold is set to 0.5 and 28 rasters remain at a threshold of 0.2. The raster in which this blob should be visible, has the region RA(297.0, 299.0), Dec(17.0,19.0)), and is present among the remaining rasters under threshold 0.5. A2 (2x2) is rastered into 35 windows, which is a much smaller amount than A1. However, the number of stars per raster is higher and the area that is split up in these files is smaller, so what you see in the image is a denser appearance of the stars.

A3 detects blobs in 39 out of 285 rasters with a threshold set to 0.2. At a threshold of 0.5 it will find one or more blobs in 23 rasters. And all rasters of the known GCs, but one (Arp Madore 1), have blobs detected under a threshold of 0.2. It is understandable that Arp Madore 1 does not get detected as a blob because it is one of most distant known GCs of the Milky Way [40].

The precise GCs that were found can be seen in the overview in Table 4.2.

Table 4.2: What known GCs are getting detected for a threshold of 0.2

GC	DoGs Present GCs	GC	DOGs Present GCs	GC	DoGs Present GCs
Area 1		Area 2		Area 3	
M3	Present	M71	Present	47 Tucanae	Present
M5	Present			NGC 121	Present
NGC 5024	Present			NGC 1049	Present
NGC 4147	Present			NGC 362	Present
NGC 5053	Present			NGC 1261	Present
NGC 5466	Present			NGC 1629	Present
Koposov 1				NGC 1644	Present
Palomar 3				NGC 1651	Present
Palomar 4				NGC 1652	Present
Palomar 5	Present			NGC 1696	Present
GCI 38				NGC 1756	Present
Willman 1				NGC 1783	Present
				NGC 1786	Present
				NGC 1795	Present
				NGC 1841	Present
				Arp Madore 1	

4.1.3 A4 - Finding Other Stellar Structures

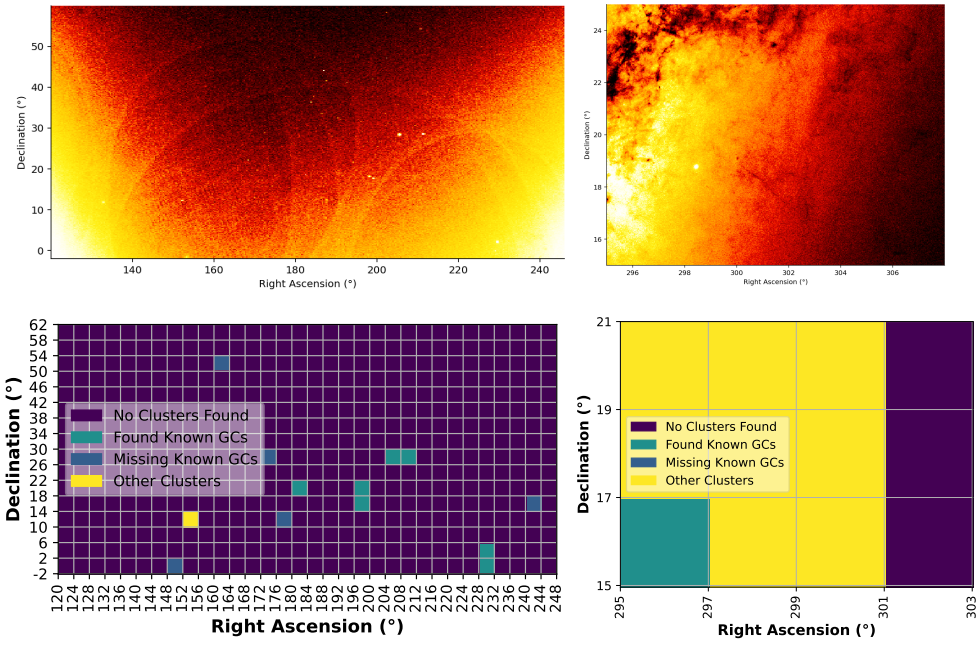
A4 does not contain any known GCs but has 60 out of 120 raster files containing blobs according to DoG using a threshold value of 0.2. And 19 at a threshold of 0.5. Some of the stellar structures are described in Table 4.3.

Table 4.3: Some Identifiable Rasters found Area 4

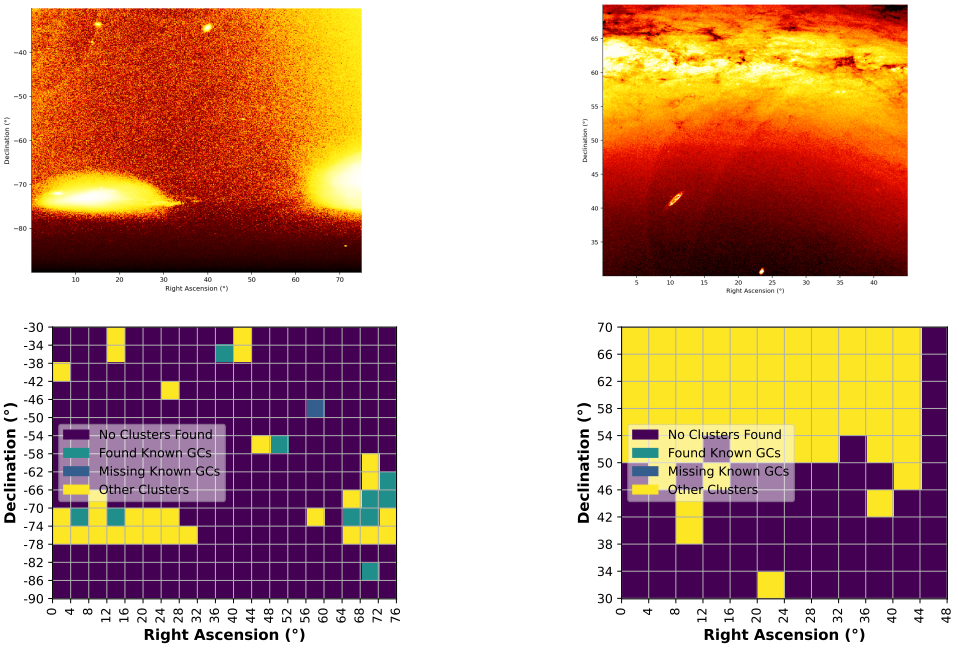
coordinates in `ra - dec`	**Identified As**	**Blob Description**
`10.0 - 41.0`	Andromeda	Massive stars and the dwarf galaxy NGC 205 within Andromeda are seen as blobs but Andromeda itself is not, probably because of the shape.
`23.4 - 30.0`	Triangulum Galaxy M33	Massive circular shape is detected which is circled by more subtle blobs. The main shape is the spiral galaxy.
`(0 to 4)-(62 to 66)`	Supernova Remnant SNR G116.9+00.1, open star cluster st 18 within the Little Rosette Nebula, dark nebula LDN 1268	The raster is quite busy, it contains a super nova remnant, an open star cluster, a dark nebula and many stars. Ten large blobs and seven smaller ones are detected.

4.1.4 Rasters Graph Comparing DoG Findings to Respective Areas

Figure 4.2 shows rasters containing blobs as identified by dog across the four areas. All four areas are represented as a plot of rasters indicating where DoG finds *known GCs*, *other blobs*, and *no blobs*. It also shows what rasters should have found GCs but instead have it be the *missing known GCs* locations. To give a clear idea of where you can picture these locations, it is displayed below the scatter-plot of stars.



(a) A1 (b) A2



(c) A3 (d) A4

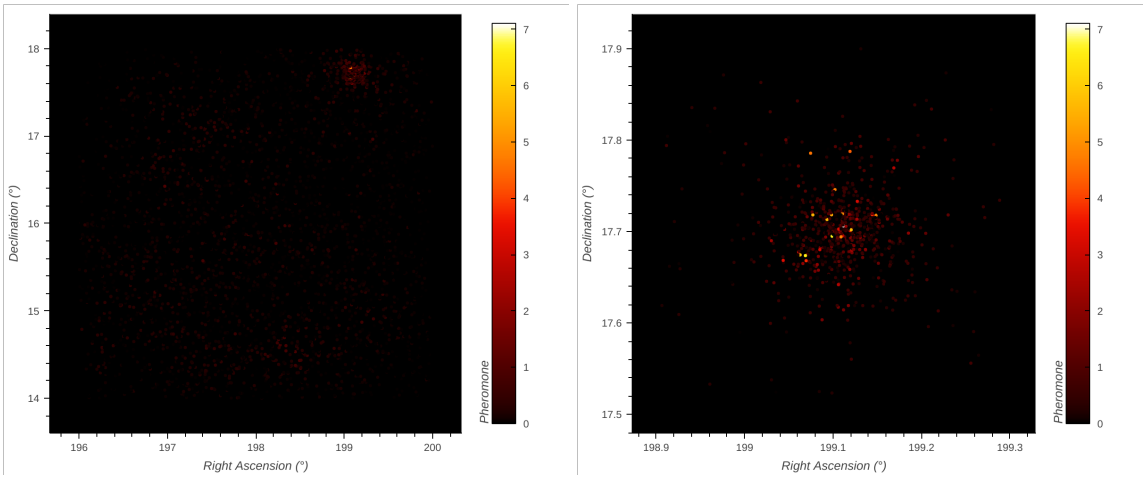
Figure 4.2: Rasters-and scatter-plot comparison where DoG was run with threshold set to 0.2

This Figure makes clear that areas that have a high number of stars or a high contrast in star density (i.e. A1, A2, and A3) has many blobs detected by DoG. From the A2 and A4 raster figure to scatter-plot comparison we see that it finds blobs at the yellow bright parts of the scatter-plots. A3 has blobs detected at the locations of the Magellanic Clouds. We can also see that ate the blacker dark regions in the scatter-plots represented in the raster graph, there are near to no blobs found. The scatter-plot of A1 has the yellow brightest parts in the corners, however, in comparison to the number of stars present in the brightest parts of the other areas it contains less stars.(plot might not be representative in comparison to other areas scatter-plots)

4.2 Ant

The pheromone traces from the Ant Colony algorithm can be nicely visualized in the heatmaps from Figure 4.3. Figure 4.3a depicts a pheromone heatmap of the values of the raster containing

two GCs. The bar to the left is a bar indicating the spread of the pheromone values in this raster, where it is indicated from black to white as going from a low to a high pheromone value. The points in the scatter plot represent the stars in the raster by a color that indicates the pheromone value. There are minor pheromone values spread out all over the raster yet it is clear that a concentrated amount is found at the top right corner with the regional bounds of RA(198.9 – 199.3) and Dec(17.5 – 17.9). This is where the clusters are located. When honing in on this region (see Figure 4.3b) it becomes more clear that this is where the major pheromone values are located. The closer to the center, the higher valued the stars are. These figures are an indication of the Ant Colony algorithm functioning as they are intended, zoning in on star clusters through appointed pheromone values to stars in dense areas.



(a) Complete Raster perspective

(b) Zoomed in on the GCs perspective

Figure 4.3: The 2D pheromone heatmaps of the overlapping GCs NGC5024 and NGC5053

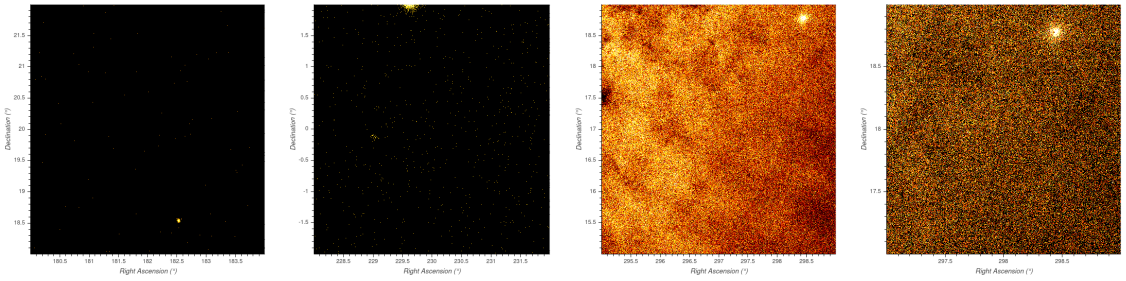
To understand how this Algorithm looks like in different types of regions, we take a closer look at some example areas in the Figures of 4.4 and 4.5. These Figures compare the densities of stars in different raster areas and how the Ant Colony algorithm handles them is visualized in the heatmaps depicted below the scatter-plots of the stars in the rasters.

Both Figure 4.4e and Figure 4.4f representing A1, have yellow spots depicting stars with high pheromone values. In these rasters you can see relatively clearly that the Ant Colony algorithm marks the locations of the GCs with higher pheromone values.

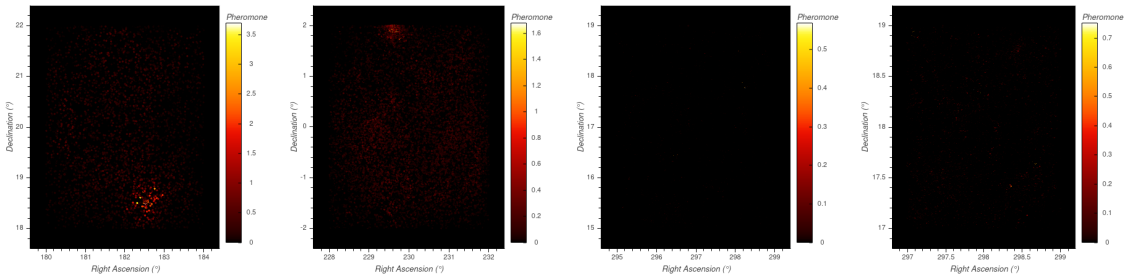
A2 (2by2) is a more zoomed in version of the (4by4) raster that focuses on the present GC of this area. The heatmaps of both Figure 4.4g and Figure 4.4h are very dark. Figure 4.4h shows some small and nearly invisible pheromone spots and has a slightly higher maximum pheromones of 0.75 than the Figure 4.4g of (4by4) with a maximum pheromone of 0.55. **As the (2by2) raster gets a better result here we will mainly be looking at the (2by2) raster information from now on.**

A3 showing its two Magellanic Clouds has a similar occurrence of an incredibly dense stellar region that does not get represented well in the pheromone Heatmap, showing (in black) how the Ant Colony algorithm is handling these type of areas. Table 4.5 displays the number of stars (N_{stars}) for each sub-figure in Figure 4.4 and Figure 4.5, showing that the N_{stars} of A2 and A3 are more than a two-hundred-thousand and even go up to a million stars, and the rasters shown of A1 and A4 are less than thirty-thousand. This difference is huge and would explain the behavior of the Ant Colony algorithm depicted in the heatmaps that for A3 are incredibly dark.

A4 has two examples of the algorithms effect on huge Galaxies. The Galaxies locations in the heatmaps are depicted as more dense red smudges, that are noticeable but not the brightest areas in the heatmaps. The highest pheromone values emerge in a cluster of stars that occurs for an unknown reason. This might be a group of stars worth looking into and worth further investigating with the third part of this pipeline (4.3)

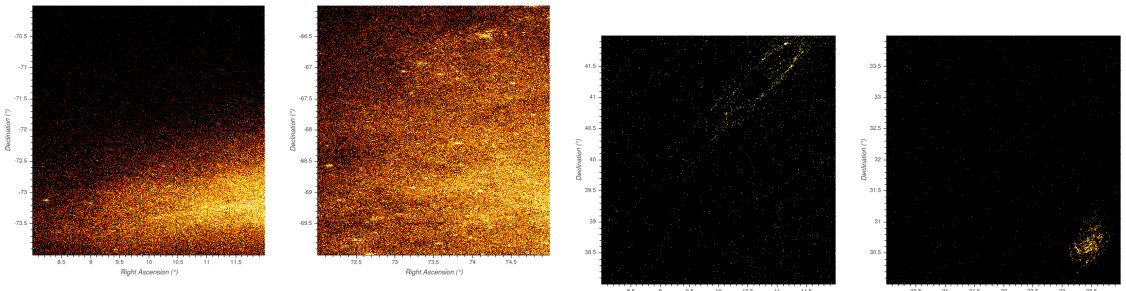


(a) A1: NGC 4147 (b) A1: Palomar 5 (c) A2 (4by4): M71 (d) A2 (2by2): M71

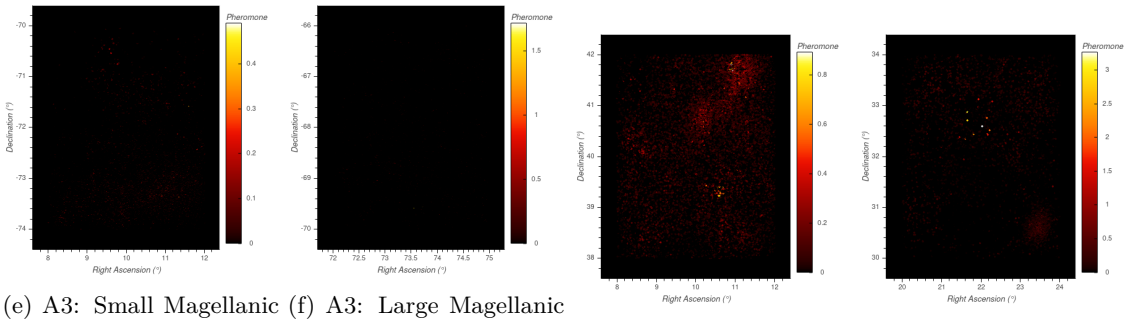


(e) A1: NGC 4147 (f) A1: Palomar 5 (g) A2 (4by4): M71 (h) A2 (2by2): M71

Figure 4.4: Scatterplots (top) vs. Heatmaps (below) of varying rasters of A1 and A2



(a) A3: Small Magellanic Cloud (b) A3: Large Magellanic Cloud (c) A4: Andromeda Gal (d) A4: Triangulum Gal



(e) A3: Small Magellanic Cloud (f) A3: Large Magellanic Cloud (g) A4: Andromeda Gal (h) A4: Triangulum Gal

Figure 4.5: Scatterplots (top) vs. Heatmaps (below) of varying rasters of A3 and A4

Table 4.4 helps visualize a possible relation between the number of stars present in a raster and the maximum pheromone values present.

Table 4.4: Visualization of the number of stars and maximum pheromone values of Figure 4.4 and Figure 4.5

Figure	N_{stars}	max pheromone bar
4.4a A1 NGC 4147	5612	3.75
4.4b A1 Palomar5 + M5	19290	1.50
4.4c A2 (4x4)	1391301	0.55
4.4d A2 (2x2)	331916	0.75
4.5a A3 Small Magellanic Cloud	239828	0.50
4.5b A3 Large Magellanic Cloud	523931	1.75
4.5c A4 Andromeda	21587	0.90
4.5d A4 Triangulum	14568	3.25

4.2.1 Stats on the pheromone distribution per area:

The Ant Colony algorithm outputs a pheromone value for every star in the data-set. The stars that were not visited by an ant have a pheromone value nearly set to zero and the ones that were visited have a pheromone value somewhere between zero and nine. The pheromone distribution can be seen in Figure 4.6

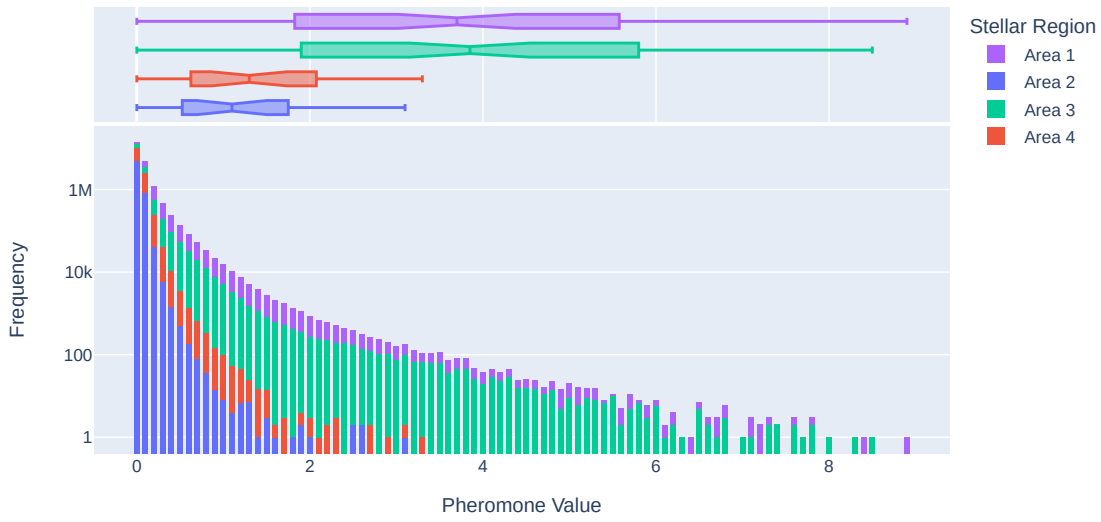


Figure 4.6: Logarithmic Distribution of Pheromone for the Four Areas

Most pheromone values in A2 and A4 lie between zero and two. For A1 and A3 the values are spread out more an lie between 2 and 6. The expectation for this distribution, following from what relation we suspect based on Table 4.4, is that A1 has the widest spread of pheromone values and that A2 has a smaller spread. This seems to be an accurate thought looking at the histogram in Figure 4.6. It is a bit surprising to see that A3s distribution is similar to A1s and A4s distribution is similar to A2. The distribution of A3 could be explained by the fact that next to bright rasters showing the small and large MC it also has regions that are darker with a lower number of stars. The rasters of A4 display a more homogeneous distribution of stars. Though on a macro level, A4 displays heterogeneous stellar structures, per raster, the distribution appears homogeneous. This helps to explain why A4 displays a uniform distribution of pheromone values.

When looking at Table 4.5, it becomes clear that not only the distribution of A1 and A3, and A2 and A4 are similar but also the mean (μ) pheromone values. The μ pheromone values of the visited stars in A1 and A3 lie around 0.16 and in A2 and A4 they lie around 0.04.

Area	μ pheromone value of the visited stars
A1	0.1654
A2 (2x2)	0.0345
A3	0.1545
A4	0.0515

Table 4.5: The mean (μ) pheromone values per area

The Ant Colony algorithm was run on A2 (2by2) multiple times with different settings for the constants N_{iter}, N_s, and N_a. The constants were adjusted one at a time and had an increase to four time the default values; $N_{\text{iter}} = 20$, $N_s = 4000$, and $N_a = 120$. The μ pheromone values for the different adjustments can be found in Table 4.6.

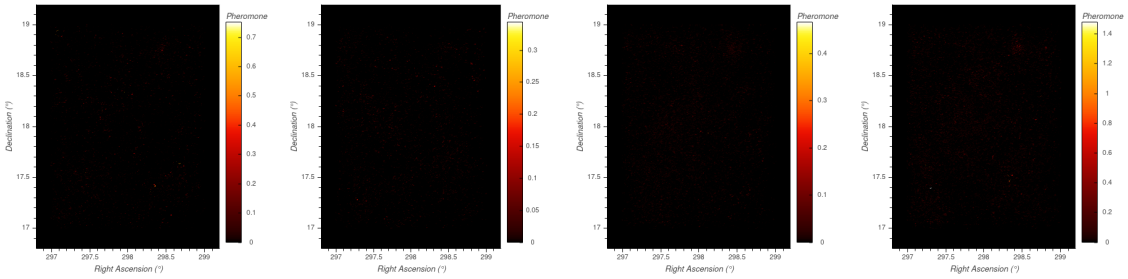
Adjustment	μ pheromone value for all visited stars in A2	μ pheromone value of all stars in A2
No Adjustment	0.0345	0.0052
N_{iter}	0.0158	0.0029
N_a	0.0140	0.0052
N_s	0.0587	0.0209

Table 4.6: The mean pheromone values for A2 with the adjusted constants N_{iter}, N_s, and N_a.

Making the number of iterations that the Ant Colony goes through, four times as high ($4\times N_{\text{iter}}$=20), results in a lower μ pheromone value for all visited as well as stars in general. A similar decrease of the μ pheromone values for the visited stars happens when the number ants increases to 120. However, the μ pheromone value of all stars stays the same. Unlike the adjustments of N_{iter} and N_a the increase of the number of steps, N_s, from 1000 to 4000 makes the μ pheromone value for all stars increase to four times the value and when looking at only the visited stars the increase of the μ value is 70%.

What does this mean? (analysis)
TODO: **I believe I am missing the max pheromone values here, as I am looking at the heatmap figure I see a difference that pops out**

There are visible differences (in Table 4.6) for the μ pheromone values of the stars for each adjustment of a constant. However, when looking at all the heatmaps of the example of A2s GC (in Figure 4.7), run for the adjustments, there barely seem to be any differences among them. This comes out differently after running the Gravitational Clustering 4.3 on it though.



(a) No adjustment (b) N_{iter} adjustment (c) N_a adjustment (d) N_s adjustment

Figure 4.7: A2s GC raster for all adjusted versions of running Ant Colony

From the information we have looked at so far, it becomes clear that the N_{stars} is important as well as the pheromone distribution. When observing Figure 4.7 about the raster of A2 including the GC M71, it looks almost empty even though the N_{stars} (=331 916) is a large number. Hence we should look at some statistics of the actual visited stars in this and other areas.

4.2.2 Stars Visited per Area

TODO: **compute the percentage visited per area for A2 (4x4) comment on why it is or is not different**.

The total number of visitations ($N_{\text{visitations}}$) per raster that can happen is determined by the number of ants, steps, and iterations; $N_a \times N_s \times N_{\text{iter}} = N_{\text{visitations}}$ total. With the default values for the ants, steps, and iterations $N_{\text{visitations}}$, $30 \times 1000 \times 5$, makes 150 000. This number is not spread out evenly over all stars as the focus of the algorithm is to have the ants visit the most dense parts of the rasters, which is where the visitations are high and where the ants will hang around. As the number of stars per area and the approximation of stars per raster is of different size and the total possible $N_{\text{visitations}}$ is the stays the same, the number of visited stars and the percentage of stars visited per area is diverse for the different areas.

Area	Number of Stars	Approximation of Stars per Raster	Visited Stars	Percentage (%)
A1	3 540 521	6915	2 658 150	75.1
A2 (2x2)	5 745 034	164 144	871 919	15.1
A3	4 031 209	14 145	1 583 441	39.3
A4	7 374 125	61 451	2 002 229	27.2

Table 4.7: The percentage of stars visited in the Ant Algorithm, per area

The average percentage is about 34%. However, when looking at each area you can see that the percentage visited in A1 is much higher than A2. This is likely explained by the difference in star population density for these areas. The number of stars that reside in A2 is nearly to double the number that reside in A1 while A1 is much larger than A2. Which means there are more stars per raster but the algorithm still is set the same, has a set number of ants, with a set number of steps, and a set number of rounds that it will do. The low percentage is not unexpected and has been tried to be accounted for by having a smaller raster for A2, however, this still led to an average of 15% of visited stars in A2. A smaller raster means that the algorithm gets run on a smaller set of stars each time, increasing the number of visited stars, however the danger is that when splitting up the stars, one might divide the GC into pieces. It depends on where the dataset is cut on wether this GC division happens.

How will the visitation percentage change when altering the parameters N_{iter}, N_a, and N_s?

After changing the parameters **separately** by multiplying the value by **four** (so N_{iter} becomes 20, N_a becomes 120, and N_s becomes 4000), and running the ant algorithm on it, the percentage of visitations have gone up.

Adjustment	Percentage (%)
No Adjustment	15.1
N_{iter}	18.2
N_a	37.4
N_s	35.6

Table 4.8: The visited percentage of stars in A2 (2x2) with adjusted values for N_{iter}, N_a, and N_s.

The adjustment of N_{iter} from 5 to 20 results in a visitation of 18.2 %, which is barely any higher than the default of 15.1%. This increase in iterations show us is that iterating over this area more times will not visit a larger variety of stars, but mainly increase the number of visitations for the same stars.

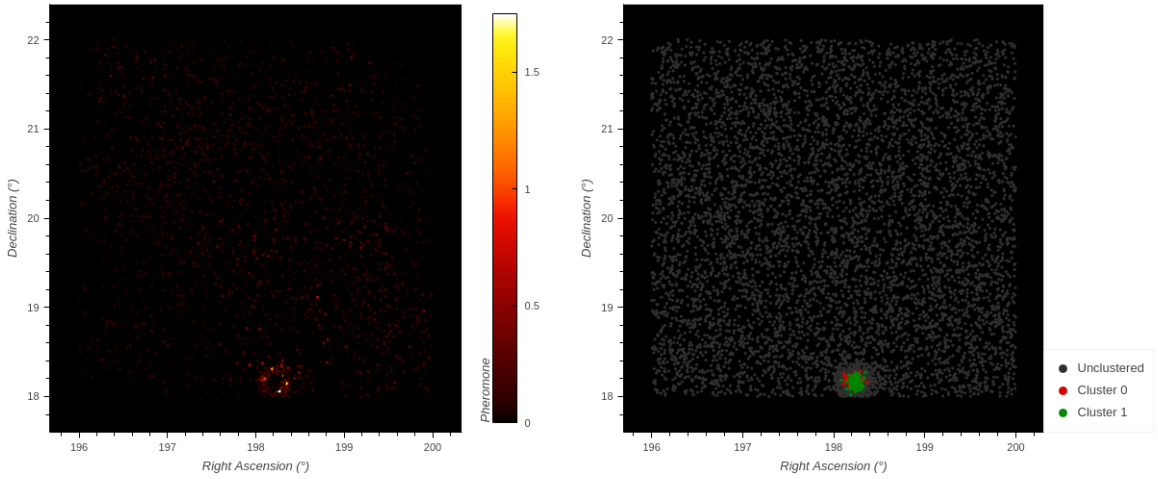
The adjustment of N_a from 30 to 120 results in a visitation of 37.4%, which is more than double with the default settings. This increase in the number of ants shows us when you have more ants in the field, there could arise a larger spread of visited neighborhoods, in search for the densest parts. As there are more ants walking around every where, there could occur a moment of confusion in the beginning at where the dense parts lie, as it is crawling of more ants leaving pheromone in more places, possibly confusing each other what direction to go into. TODO: **not the best possible reason or explanation, yet**

The adjustment of N_s from 1000 to 4000 results in a visitation of 35.6.4%, which is more than double with the default settings. This increase in the number of steps shows us that when the ants walk around for longer, it gives them more opportunity to include more stars in their dense area. They could also be walking more neighborhoods in the proximity of their densest part, by having the a higher probability, through more steps, to stumble on them.

More insight on these adjustments can be found in the results Clustering Section 4.3, in Table 4.9.

4.3 Clustering

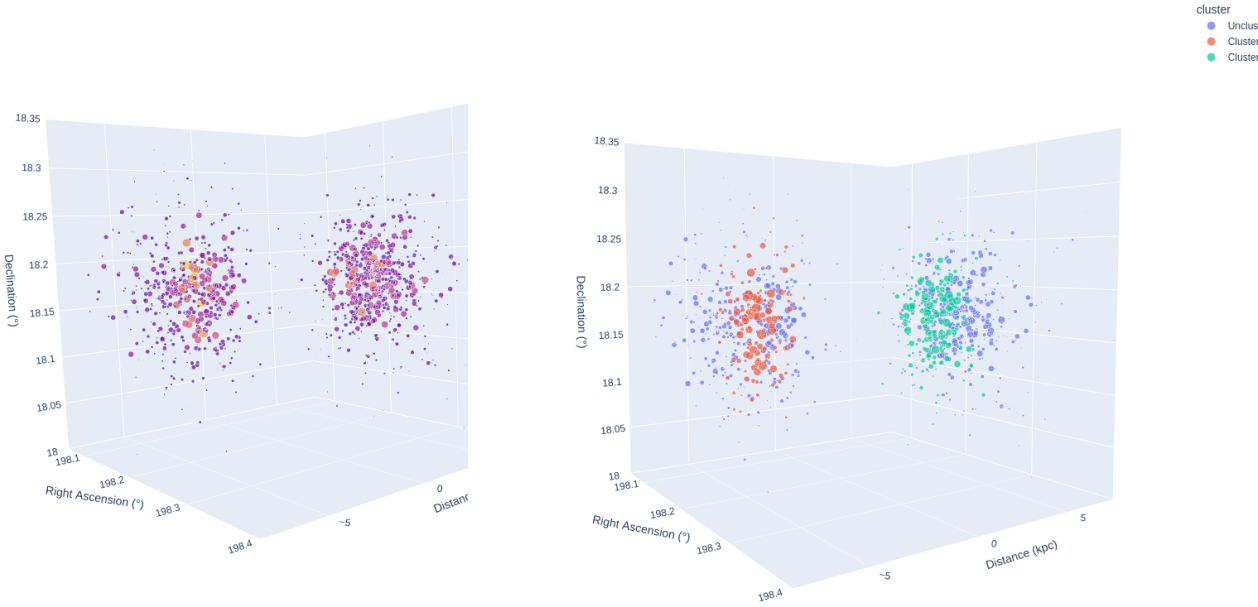
The output of the clustering will show the final results of the pipeline. Where it has detected a cluster and what stars belong to this cluster. In Figure 4.8 it becomes clear how the pheromone values have an influence on the clustering output. The pheromone values are clearly visible in the right upper corner of the pheromone heatmap, and so is the cluster also found in the same location (right upper corner), as can be easily seen in this 2D cluster plot.



(a) Pheromone Heatmap (b) Pheromone Heatmap

Figure 4.8: Pheromone heatmap and Cluster Plot of NGC5025

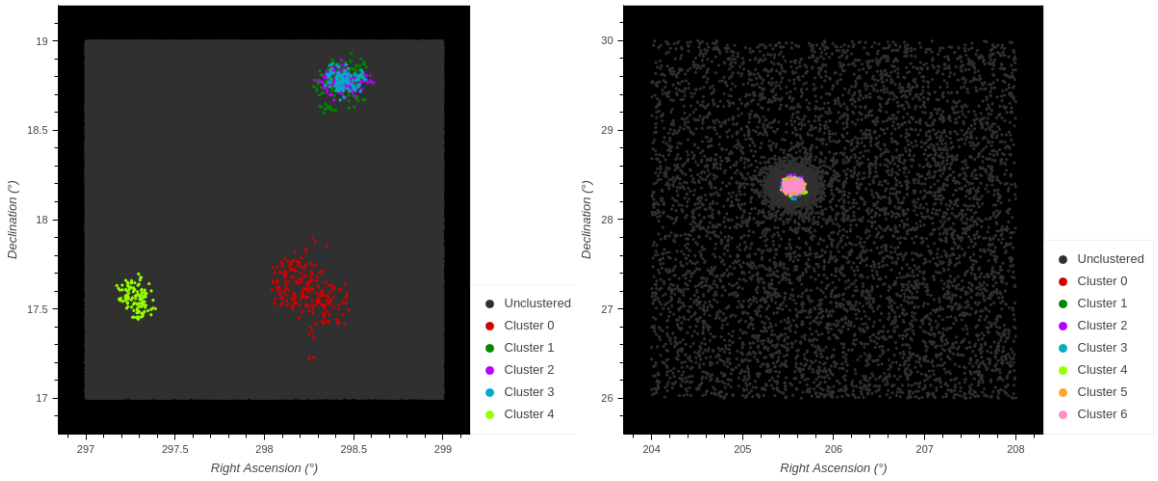
The 2D angle which shows the *RA-Dec* side of the clusters location makes it easy to pinpoint its location. However, it cannot give insight on depth and it is unclear about how the stellar sets that were found belong together as cluster formations. To visualize the clustering within a raster, involving information on the depth, 3D plotting is needed.



(a) Pheromone Heatmap (b) 3D Cluster Plot

Figure 4.9: 3D Plots of NGC 5025

It specifically gives insight into what is going on in the raster when multiple clusters are found that are overlapping. As the stars with a negative *parallax* have been kept in the dataset, overlapping clusters might be the same cluster. This assumption could be drawn when the clusters are overlapping and located approximately at each others mirroring *distance*, for example at 5 kpc and -5 kpc, as can be seen in Figure 4.9. To know what GCs might be overlapping you first should check if the clusters are overlapping on the *RA-Dec* side. What this side can indicate is if the clusters are distinct or overlapping clusters. In Figure 4.10a you can see two clearly distinct clusters: the yellow cluster Cluster 4 and the red cluster Custer 0. In Figure 4.10b there are six other clusters hidden behind pink cluster Cluster 6. These could be distinct but will only become clear from investigating the clusters depth locations.



(a) Distinct Clusters example

(b) Overlapping Clusters example

Figure 4.10: 2D Distinct and Overlapping Clusters

TODO: **For each area an example plot of the 2d pheromone heatmap and the clusters it has found, for area 4 show a plot of what it thinks is a gc. For area 3 show 47 Tucane and the distant cluster which most likely is not found: Arp - Madore 1, maybe NGCs that are really close: 1651 and 1652 (will have to make a table to view which files they are in.) for area 2 show the variations on the GC (`a2_297.0_17.0.bin`) so default (2d and 3d) and the Ns variation on the same raster containing the GC. Also the 3d cluster plot of `a2_307.0_21.0.bin` in default version. Area 1 has the interesting overlapping clusters and otherwise idkn.**

Area	Total Clusters Detected	Detected in number of Rasters
A1	104	77
A2 (2x2)	2	2
A3	52	46
A4	6	5

Table 4.9: The number of clusters detected in a number of rasters for each area

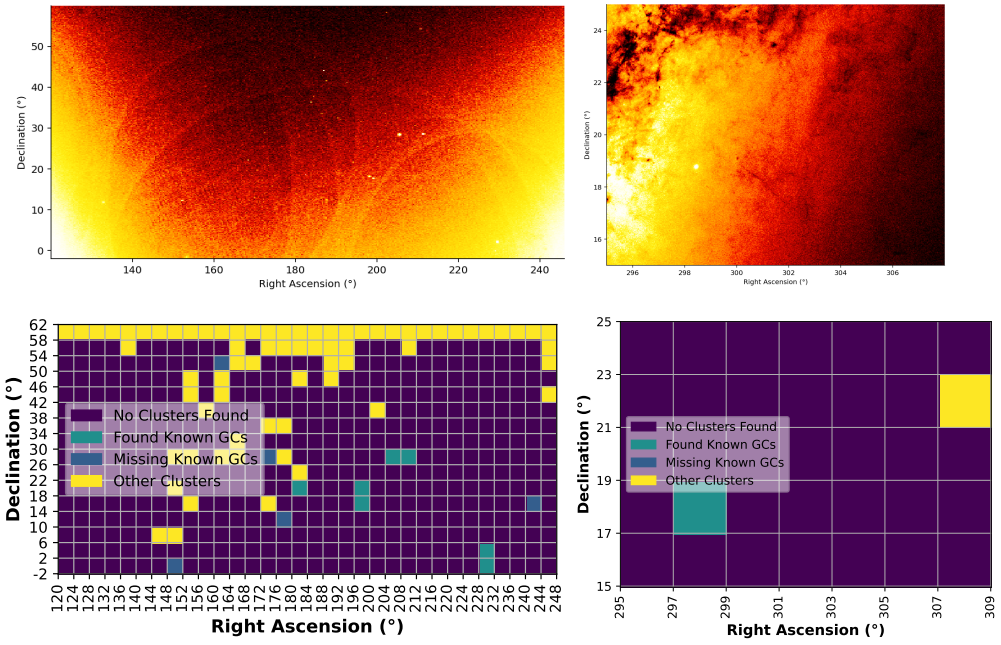
Adjustment	Total Clusters Detected	Detected in number of Rasters
No Adjustment	2	2
N_{iter}	0	0
N_a	0	0
N_s	83	30

Table 4.10: The number of clusters detected in a number of asters for each area

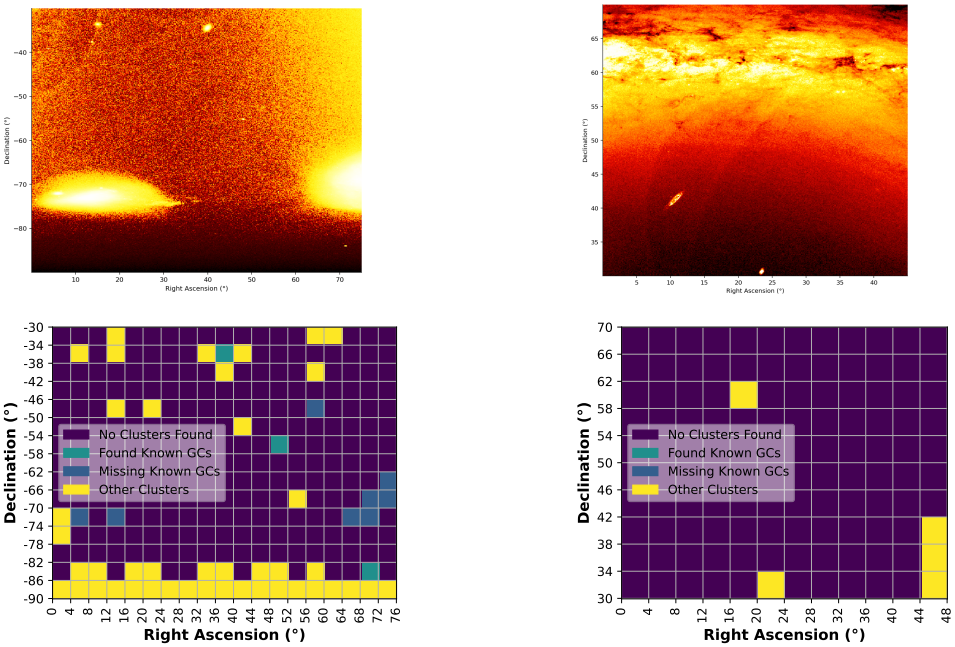
TODO: **fill in**

Table 4.11: What known GCs are getting found through clustering

GC	DoGs Present GCs	GC	DOGs Present GCs	GC	DoGs Present GCs
Area 1		Area 2		Area 3	
M3	Present	M71	Present	47 Tucanae	
M5	Present			NGC 121	
NGC 5024	Present			NGC 1049	Present
NGC 4147	Present			NGC 362	
NGC 5053	Present			NGC 1261	Present
NGC 5466	Present			NGC 1629	
Koposov 1				NGC 1644	
Palomar 3				NGC 1651	
Palomar 4				NGC 1652	
Palomar 5	Present			NGC 1696	
GCI 38				NGC 1756	
Willman 1				NGC 1783	
				NGC 1786	
				NGC 1795	
				NGC 1841	Present
				Arp Madore 1	



(a) A1 (b) A2



(c) A3 (d) A4

Figure 4.11: Remaining Rasters of Areas After Clustering

Results of Complete Pipeline

Area	Total Clusters Detected	Detected in number of Rasters
A1	16	7
A2 (2x2)	1	1
A3	12	8
A4	3	2

Table 4.12: The number of clusters detected in a number of rasters for each area

Table 4.13: Pipeline Cluster Results

Cluster Type	Name	RA (°)	Dec (°)	distance (kpc)	bounds of distance (kpc)
		A1			
GC	M3	205.55°	28°22′12″	6.47	6.07 – 6.88
GC	M3	205.56°	28°22′12″	4.17	4.01 – 4.39
GC	M3	205.55°	28°22′48″	4.77	4.41 – 5.10
GC	M3	205.56°	28°22′12″	5.27	4.99 – 5.52
GC	M3	205.57°	28°22′12″	7.17	6.93 – 7.45
GC	M3	205.55°	28°21′36″	-4.92	-5.09 – -4.75
GC	M3	205.56°	28°22′12″	-4.17	-4.33 – -4.03
GC	M5	229.63°	2°5′12″	-4.14	-4.32 – -4.00
GC	M5	229.64°	2°4′12″	-5.30	-5.53 – -5.10
GC	NGC 4147	182.53°	18°42′0″	4.76	4.01 – 5.95
GC	NGC 5053	199.11°	17°42′0″	4.76	4.02 – 5.79
GC	NGC 5024	198.23°	18°10′12″	-5.17	-5.73 – -4.64
GC	NGC 5024	198.23°	18°9′36″	4.55	4.10 – 5.03
GC	NGC 5466	211.36°	28°31′48″	-4.79	-5.41 – -4.18
GC	NGC 5466	211.37°	28°31′48″	4.47	4.07 – 5.04
GC	Palomar 5	229.62°	1°57′36″	4.54	4.01 – 5.17
		A2			
GC	M71	298.42°	18°47′24″	6.36	6.19 – 6.53
		A3			
unknown	0.0 -78.0	3.52°	−74°22′48″	-4.85	-5.71 – -4.01
unknown	0.0 -78.0	3.58°	−74°34′48″	4.45	4.01 – 5.06
unknown	0.0 -74.0	3.70°	−73°30′36″	4.40	4.01 – 4.95
Galaxy	Southern Pinwheel Galaxy	13.75°	−37°21′0″	4.65	4.00 – 5.79
Galaxy	Sculptor Dwarf Galaxy	15.02°	−33°42′36″	4.83	4.01 – 6.06
GC	NGC 1049	39.81°	−34°33′36″	4.70	4.00 – 5.53
GC	NGC 1049	39.84°	−34°33′36″	-4.41	-4.96 – -4.01
Galaxy	Fornax Dwarf Galaxy	40.16°	−34°25′48″	4.63	4.02 – 5.42
Galaxy	Fornax Dwarf Galaxy	40.12°	−34°29′24″	-4.32	-4.71 – -4.02
GC	NGC 1261	48.08°	−55°12′36″	4.67	4.01 – 5.51
GC	NGC 1261	48.07°	−55°13′12″	-4.49	-5.05 – -4.00
GC	NGC 1841	71.38°	−84°0′36″	-5.63	-7.63 – -4.06
		A4			
Galaxy	Triangulum Galaxy	23.48°	30°39′0″	4.43	4.00 – 5.04
Galaxy	Triangulum Galaxy	23.45°	30°36′0″	-4.31	-4.72 – -4.01
unknown	16.0 58.0	16.22°	60°39′0″	4.16	4.01 – 4.39

The raster of M3 contains seven clusters that overlap at coordinates (205.5°,28°22). The mirrored clusters could lie at distances of: 4.17 and -4.17, and 4.77 and -4.92. The raster of M5 contains two

clusters that both lie at a negative distance of -4.14 and -5.30 kpc.

Except for the raster of the known GCs M3 and M5, all overlapping clusters are in pairs that are at opposite parallaxes, one negative and one positive,that almost are at opposite distance values form zero.

If all distances would be assumed as positive the distance of the found clusters lies between: 4.00 and 7.63 kpc.

Koposov 1 has a distance of 48.3 kpc, palomar 4 a distance of 109 kpc, GCI 38 a distance of 74.7 kpc, NGC 121 a distance of 61 kpc, ngc 362 8.5 kpc, NGC 1783 49 kpc, arp madore 1 123.3 kpc.

47 Tucanae with a distance of 4.5 kpc has not been found, the couse however, is likely that this GC lies in or near the megalanic clouds so through the density the pipeline has trouble spotting it.

Chapter 5

Conclusion

5.0.1 graphics

- Answer the main questions on what it does
- and if it work as intended

5.1 DoG

GCs are stellar agglomerates with a radius varying from 0.5 pc to 10 pc, typically centered at 3 pc to 5 pc [41]. The DoG method can find these agglomerate structures, or blobs. However, what variety of blob sizes it finds depends on the set parameters. This method generally has more difficulty finding larger blobs [42], but from the list of known GCs in A1, A2, and A3 we see it doesn't have trouble finding the larger GCs, rather finding the ones that are smaller and blend into the background with the rest of the brighter stars. Also from A4 we learn that other structures are indeed hard to separate with this method. Hence the follow up with an Ant algorithm based clustering method.

The DoG method does not work perfectly as a pre-processing method, as it filters out known GCs. In A1 it only finds 7 out of 12 of the known GCs when the threshold is set to 0.2. When setting the threshold of the blob size really low (say a threshold of 0.12) it would include almost all raster files but it also would not filter those areas that are absolutely dark, failing to decrease the amount of data.

A reason for wanting to decrease the number of rasters is that when it comes to stellar data it can consist of big data and so the computations to get to the results are vast and it takes time to get there. So if this method of filtration does not work another way to go is to make it computationally run fast. This can be done by the optimization of the implementation (the Julia language, Parallelization and Peregrine solve this issue in a different way).

However, when observing what DoG finds of the known GCs of A3 at a threshold set to 0.2 the detection technique seems to almost work perfectly as it finds 15 out of 16. Having a closer look at the GC it doesn't find, called Arp Madore 1, it turns out that this is one of the most distant GCs of the Milkey Way galaxy [40]. Looking at the GCs not found in A1 it becomes clear that they are like Arp Madore 1 either have a large distance, have low luminosity, or both. If the GC is far away due to a large distance, its overall magnitude would be hard to pick up in an image, as is a low luminosity. So we can conclude that the bolb-detection technique works well as a pre-processing method given the GC has a bright enough overall magnitude visible for the image that gets processed by DoG.

5.2 Ant

Through experimentation on the starting conditions of the Ant Colony random-walk algorithm, it is very evident that the results of this algorithm are heavily influenced by the overall stellar density of a region, and the Ant Colony adjustable parameters (number of iterations of the core algorithm, number of ants initialized, and the number of steps they can take).

The Ant Colony algorithm has the most influence on whether the pipeline outputs the most insightful result. Its adjustable parameters determine whether they have the functional setting for the most insightful result for the type of regional situations. The type of regions can be split into dark

and bright regions, whose main difference is the number of stars present. In this report the adjustable parameter values that work for the dark regions with GCs are used on bright regions to evaluate if that works and whether adjustments are needed. It shows that adjustments are needed and the adjustable parameters settings for the type of bright region needs more experimentation.

5.3 Clustering

Chapter 6

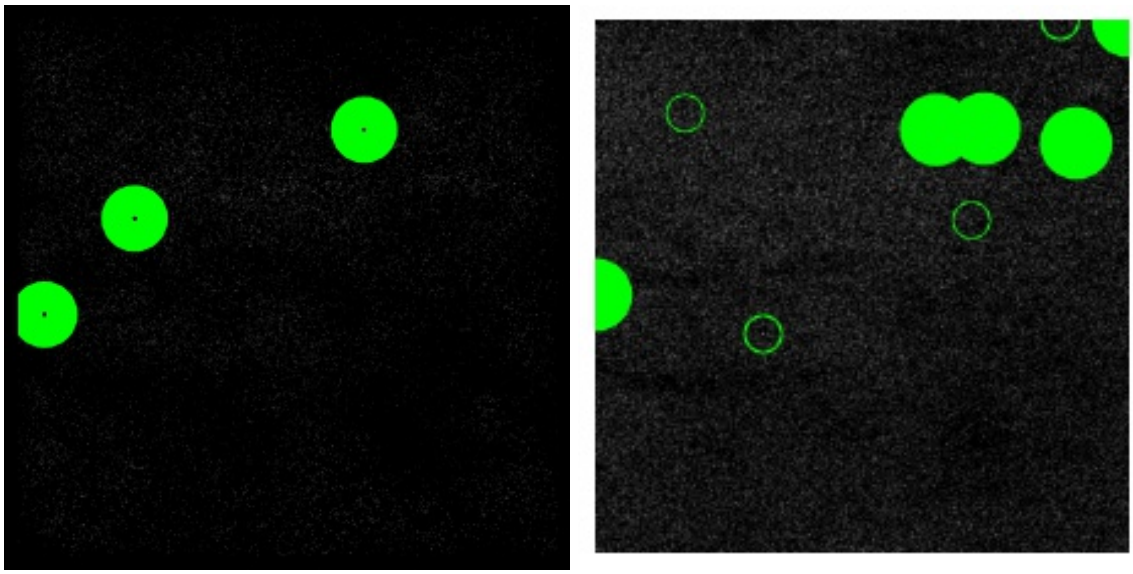
Evaluation

- try answering why it does what it does
- contrast to other work
- what might work, followup research

6.1 DoG

- What rasters remain at which threshold?
- does it filter out known GC's
- As a pre-processing method it doesn't work but how about post processing, why would that work?
- What about the busy areas vs. the empty ones? - what makes this behavior occur

The DoG filter applied as a post-processing method might help give more information. See Figure 6.1 to observe the difference in blob detection based as a pre- and post-processing method. The post-processing method can make use of the computed pheromone values of the Ant algorithm. This is different from basing it on the the absolute magnitude. Basing the image on the magnitude one looks at the brightness and not at the stellar geography, which might even be misleading.



(a) Pheromone based (b) Magnitude based

Figure 6.1: Blobs found through DoG, based on the Pheromone and Magnitude values, making up the image.

These images show that if one would focus on the pheromone values instead of the brightness (or magnitude) you get different results on what would constitute as blobs. Which shows that it might

have a different insight as a post-processing method. As less small blobs that are actually lonely bright stars would remain, as the magnitude longer of influence on where the image is bright.

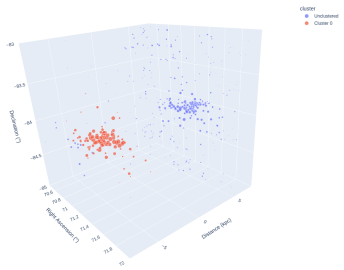
6.2 Ant

- How many stars are visited (in %)

6.3 Clustering

- One can give a percentage on how many known clusters it found.
- One can then say it also found more objects of which some can be recognized as dwarf galaxies, galaxies and other parts.

Keeping the negative parallax as was suggested by the paper of **negative_paralax** results in finding more known GCs.



(a) A1

Here, with the current set parameters, it finds the cluster in a negative distance and not in a positive one.

6.4 Shortcomings and Further Research

1. Cutoff point for the rasterization may slice through a GC.
2. The
3. Negative parallax fix-up.
4. Gaussian KDE instead of regular Gaussian Kernel.
5. Rasterize across distance as well so the rasterization is 3D.
6. Different units for RA, Dec, and Distance. Euclidean distance may not be the best distance metric to us in the ant colony and in the clustering. Perhaps something from spherical coordinates may be more appropriate? Certainly the issue of scaling concerns when considering a Unit distance in RA or Dec vs a Unit distance in distance.
7. Cut off points for rasterization at the boundary of an area can be weird.
8. Sizes of rasters could be dynamic, if the ant's can figure out their own parameters based on size and if this info can be passed into the clustering.
9. Random raster size for Area 2.

6.4.1 NOTES

TODO: **Helmi's research as an addition of insight and a source**

Looking at stars like fossils is what the Gaia data-set makes possible. Its key question is: 'How did the Milkey Way form?' Looking at pmra and pmdec one can look at the motions of stars and one finds that a group moves into the opposite direction than the rest of the Milky Way galaxy (Enceladus). They were dwarf galaxies that are now part of our Galaxy.

Place in the milkey way that is seemingly emptier is called halo. vs bulge bright center with its thick and thin disk on the sides.

Bibliography

[1] M. Giavalisco, *Galaxy Evolution*. 2000, p. 2142. DOI: 10.1888/0333750888/2142.

[2] *The Hubble Tuning Fork - Classification of Galaxies*, ESA/Hubble. [Online]. Available: https://www.spacetelescope.org/images/heic9902o/ (visited on 04/08/2020).

[3] R. Jimenez and P. Padoan, "A New Self-consistency Check on the Ages of Globular Clusters," *The Astrophysical Journal*, vol. 463, no. 1, pp. L17–L20, May 1996. DOI: 10.1086/310053. [Online]. Available: https://doi.org/10.1086/310053.

[4] R. Gratton, A. Bragaglia, E. Carretta, V. D'Orazi, S. Lucatello, and A. Sollima, "What is a Globular Cluster? an Observational Perspective.," *The Astronomy and Astrophysics Review*, vol. 27, no. 1, pp. 1–136, 2019. DOI: doi:10.1007/s00159-019-0119-3.

[5] S. van den Bergh, "How Did Globular Clusters Form?" *The Astrophysical Jorunal Letters*, vol. 559, no. 2, pp. L113–L114, Oct. 2001. DOI: 10.1086/323754. arXiv: astro-ph/0108298 [astro-ph].

[6] D. Bamberger, *How do Astronomers Count Stars in a Globular Cluster?* [Online]. Available: https://www.quora.com/How-do-astronomers-count-stars-in-a-globular-cluster?share=1 (visited on 04/06/2020).

[7] B. C. Erin M. O'Malley Christina Gilligan, "Absolute Ages and Distances of 22 GCs Using Monte Carlo Main-sequence Fitting," *The Astrophysical Journal*, vol. 838, no. 2, p. 162, Apr. 2017. DOI: 10.3847/1538-4357/aa6574. [Online]. Available: https://doi.org/10.3847/1538-4357/aa6574.

[8] M. Mohammadi, N. Petkov, K. Bunte, R. Peletier, and F.-M. Schleif, "Globular Cluster Detection in the Gaia Survey," *Neurocomputing*, vol. 342, pp. 164–171, 2019, Advances in artificial neural networks, machine learning and computational intelligence, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2018.10.081. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231219301341.

[9] A. Dotter, A. Sarajedini, and J. Anderson, "Globular Clusters in the Outer Galactic Halo: New Hubble Space Telescope / Advanced Camera for Surveys Imaging of Six Globular Clusters and the Galactic Globular Clusters Age-metalicity Relation," *The Astrophysical Journal*, vol. 738, no. 1, p. 74, Aug. 2011. DOI: 10.1088/0004-637x/738/1/74. [Online]. Available: https://doi.org/10.1088/0004-637x/738/1/74.

[10] D. R. Soderblom, "The Ages of Stars," *Annual Review of Astronomy and Astrophysics*, vol. 48, no. 1, pp. 581–629, 2010. DOI: 10.1146/annurev-astro-081309-130806. [Online]. Available: https://doi.org/10.1146/annurev-astro-081309-130806.

[11] L. Monaco, E. Pancino, F. R. Ferraro, and M. Bellazzini, "Wide-field photometry of the galactic globular cluster m22," *Monthly Notices of the Royal Astronomical Society*, vol. 349, no. 4, pp. 1278–1290, Apr. 2004, ISSN: 1365-2966. DOI: 10.1111/j.1365-2966.2004.07599.x. [Online]. Available: http://dx.doi.org/10.1111/j.1365-2966.2004.07599.x.

[12] G. D. Bothun, H. C. Harris, and J. E. Hesser, "Detection of the globular-cluster population around ngc 7814," *Publications of the Astronomical Society of the Pacific*, vol. 104, p. 1220, Dec. 1992. DOI: 10.1086/133111.

[13] M. G. Lee, H. S. Park, and H. S. Hwang, "Detection of a Large-Scale Structure of Intracluster Globular Clusters in the Virgo Cluster," *Science*, vol. 328, no. 5976, pp. 334–336, 2010, ISSN: 0036-8075. DOI: 10.1126/science.1186496. eprint: https://science.sciencemag.org/content/328/5976/334.full.pdf. [Online]. Available: https://science.sciencemag.org/content/328/5976/334.

[14] W. E. Harris. (Dec. 2010). "Catalog of Parameters for Milky Way Globular Clusters: the Database," McMaster University, [Online]. Available: https://physics.mcmaster.ca/~harris/mwgc.dat (visited on 08/25/2021).

[15] (Apr. 10, 2015). "The Crammed Center of Messier 22," ESA/Hubble, [Online]. Available: https://upload.wikimedia.org/wikipedia/commons/7/76/The%5C%5Fcrammed%5C%5Fcentre%5C%5Fof%5C%5FMessier%5C%5F22.jpg (visited on 08/16/2021).

[16] Y.-S. Ting, A. Vaz, and G. Narayan, *How Do We Study the Stars? - Yuan-Sen Ting*, Ted-Ed. [Online]. Available: https://ed.ted.com/lessons/how-do-we-study-the-stars-yuan-sen-ting (visited on 07/16/2021).

[17] *Space Technology*. [Online]. Available: https://clarkscience8.weebly.com/space-technology.html (visited on 07/16/2021).

[18] Gaia Collaboration et al., "Description of the Gaia Mission (Spacecraft, Instruments, Survey and Measurement Principles, and Operations)," *Astronomy and Astrophysics*, vol. 595, Nov. 2016. DOI: 10.1051/0004-6361/201629272. [Online]. Available: https://doi.org/10.1051/0004-6361/201629272.

[19] Gaia Collaboration et al., "Gaia Data Release 2 - Summary of the Contents and Survey Properties," *Astronomy and Astrophysics*, vol. 616, Aug. 2018. DOI: 10.1051/0004-6361/201833051. [Online]. Available: https://doi.org/10.1051/0004-6361/201833051.

[20] (2021). "Gaia Data Release Scenario," European Space Agency, [Online]. Available: https://www.cosmos.esa.int/web/gaia/release (visited on 08/25/2021).

[21] *List of Globular Clusters*. [Online]. Available: https://en.wikipedia.org/wiki/List%5C%5Fof%5C%5Fglobular%5C%5Fclusters (visited on 04/07/2020).

[22] (Aug. 16, 2021). "List of NGC Selected from the VizieR Service," Strasbourg Astronomical Data Center, [Online]. Available: https://vizier.u-strasbg.fr/viz-bin/VizieR-3?-source=VII/118/ngc2000 (visited on 08/16/2021).

[23] (Aug. 14, 2021). "Parallax," [Online]. Available: https://en.wikipedia.org/wiki/Parallax (visited on 08/24/2021).

[24] (Apr. 27, 2018). "Gaia Data Release 2 (Gaia DR2)," European Space Agency, [Online]. Available: https://www.cosmos.esa.int/web/gaia/dr2 (visited on 05/27/2020).

[25] C. Martin, *Stellarparallax Parsec1*, Sep. 24, 2006. [Online]. Available: https://commons.wikimedia.org/wiki/File:Stellarparallax%5C%5Fparsec1.svg.

[26] Gaia Collaboration et al., "Gaia Data Release 2. Summary of the Contents and Survey Properties," *Astronomy Advisory Panel*, vol. 616, A1, A1, Aug. 2018. DOI: 10.1051/0004-6361/201833051. arXiv: 1804.09365 [astro-ph.GA].

[27] X. Luri, A. G. A. Brown, L. M. Sarro, F. Arenou, C. A. L. Bailer-Jones, A. Castro-Ginard, J. de Bruijne, T. Prusti, C. Babusiaux, and H. E. Delgado, "Gaia Data Release 2," *Astronomy & Astrophysics*, vol. 616, A9, Aug. 2018, ISSN: 1432-0746. DOI: 10.1051/0004-6361/201832964. [Online]. Available: http://dx.doi.org/10.1051/0004-6361/201832964.

[28] Z. Modrono. (Jul. 18, 2017). "Dreaming Peter van de Kamp's Dream," [Online]. Available: https://reddots.space/dreaming-peter-van-de-kamps-dream/ (visited on 08/19/2021).

[29] *Diffusion Gaussian Kernel*, University of Wisconsin-Madison, 2007. [Online]. Available: http://pages.stat.wisc.edu/~mchung/teaching/MIA/reading/diffusion.gaussian.kernel.pdf (visited on 06/15/2021).

[30] *Difference of Gaussians*. [Online]. Available: https://en.wikipedia.org/wiki/Difference%5C%5Fof%5C%5FGaussians#cite%5C%5Fnote-micro.magnet.fsu.edu-1 (visited on 06/24/2021).

[31] M. A. Michael W. Davidson. (May 17, 2016). "Molecular Expressions Microscopy Primer: Digital Image Processing – Difference of Gaussians Edge Enhancement Algorithm," [Online]. Available: https://micro.magnet.fsu.edu/primer/java/digitalimaging/processing/diffgaussians/index.html (visited on 06/24/2021).

[32] B. Gecer, G. Azzopardi, and N. Petkov, "Color-blob-based Cosfire Filters for Object Recognition," *Neurocomputing*, vol. 342, pp. 164–171, 2019. DOI: doi:10.1016/j.imavis.2016.10.006.

[33] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[34] F. Xia, J. Liu, H. Nie, Y. Fu, L. Wan, and X. Kong, "Random Walks: A Review of Algorithms and Applications," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. PP, pp. 1–13, Nov. 2019. DOI: 10.1109/TETCI.2019.2952908.

[35] M. Mohammedi, "The Ant Colony," Unpublished, 2020.

[36] L. Posti and A. Helmi, "Mass and Shape of the Milky Way's Dark Matter Halo with Globular Clusters from Gaia and Hubble," *Astronomy & Astrophysics*, vol. 621, p. 10, Jan. 2019. DOI: https://doi.org/10.1051/0004-6361/201833355.

[37] E. Dugan. (Jun. 19, 2019). "What is the Difference Between a Globular Star Cluster and an Open Star Cluster?" [Online]. Available: https://astronomy.com/magazine/ask-astro/2019/06/what-is-the-difference-between-a-globular-star-cluster-and-an-open-star-cluster (visited on 08/25/2021).

[38] S. Koposov, J. T. A. de Jong, V. Belokurov, H.-W. Rix, D. B. Zucker, N. W. Evans, G. Gilmore, M. J. Irwin, and E. F. Bell, "The Discovery of Two Extremely Low Luminosity Milky Way Globular Clusters," *The Astrophysical Journal*, vol. 669, no. 1, pp. 337–342, Nov. 2007. DOI: 10.1086/521422. [Online]. Available: https://doi.org/10.1086/521422.

[39] M. E. Sharina, M. V. Ryabova, M. I. Maricheva, and A. S. Gorban, "The Stellar Population and Orbit of the Galactic Globular Cluster Palomar 3," *Astronomy Reports*, vol. 62, no. 11, pp. 733–746, Nov. 2018, ISSN: 1562-6881. DOI: 10.1134/s1063772918110069. [Online]. Available: http://dx.doi.org/10.1134/S1063772918110069.

[40] M. Aaronson, R. A. Schommer, and E. W. Olszewski, "AM-1 : a Very Distant Globular Cluster.," *Astrophysical Journal*, vol. 276, pp. 221–228, Jan. 1984. DOI: 10.1086/161605.

[41] R. Gratton, A. Bragaglia, E. Carretta, V. D'Orazi, S. Lucatello, and A. Sollima, "What is a Globular Cluster? an Observational Perspective.," *The Astronomy and Astrophysics Review.*, vol. 27, no. 1, pp. 1–136, 2019. DOI: doi:10.1007/s00159-019-0119-3. [Online]. Available: https://www.e-education.psu.edu/astro801/content/l4%5C%5Fp6.html (visited on 08/24/2021).

[42] *Blob Detection*. [Online]. Available: https://scikit-image.org/docs/0.17.x/auto%5C%5Fexamples/features%5C%5Fdetection/plot%5C%5Fblob.html (visited on 06/15/2021).