Star Cluster Classification into Open and Globular Clusters with Interpretative Grad-CAMs

Jordan Stipp
Department of Computer Science
Stanford University
jstipp@stanford.edu

Jordi A. Montana-Lopez
Department of Physics
Stanford University
jmontana@stanford.edu

Abstract

We built a star cluster classifier to determine whether known star clusters are open or globular using their images. We used a VGG16 architecture for this and looked at the class activation map to recognize which features were more relevant.

We were able to tell apart globular clusters with an accuracy of 0.94 and an f1 score of 0.93 and implement the class activation maps. We observed that the network focused on the luminosity and number of clumps to determine whether it was a globular or open cluster.

1 Introduction

We created a star cluster classifier that can distinguish between open and globular star clusters according to pictures of the clusters. This is an interesting problem to solve because the location, distribution and age of globular clusters play a key role in determining the age of galaxies and the universe. But before dating the globular clusters, a first step is to be able to distinguish them given an image of a possible cluster. A common method of locating star clusters is based on the K-means algorithm (1). Once a database of clusters is created with that method, our algorithm would be able to tell which ones are globular and which ones are open. Solving this problem using deep learning will be even more relevant in the following years, since new astronomical instruments will provide very large amounts of data which will improve the accuracy of future iterations of our model and at the same time be able to classify newly discovered star clusters.

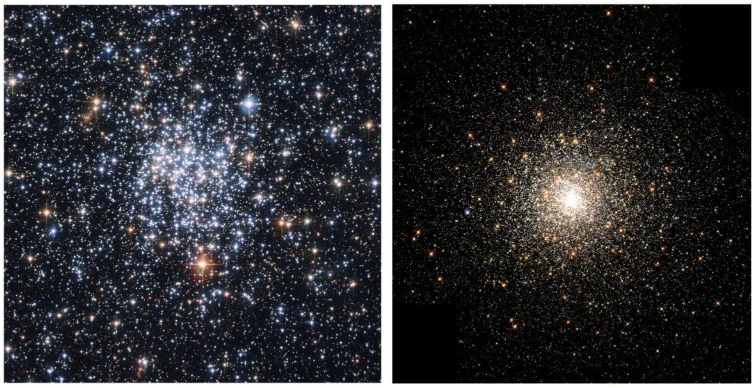

Figure 1: Open cluster (left) and globular cluster (right).

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

The main difference between the two is that stars in globular clusters are strongly gravitationally bound towards a center, and so the image looks like a bright spot in the middle. Open clusters have stars that are not so gravitationally bound together, so they appear spread out in the image.

2 Related work

Previous work concerning the classification of open and globular clusters used four-color photometry (2) and noticed a particularity of the Magellanic Clouds, in which the integrated spectra of populous star clusters can be arranged in such a way that they behave regularly along the star sequence. However, this particular regularity might not be present in other star clusters, so the reach of this technique is limited. Other work focused on locating star clusters using the K-means algorithm on near-infrared stellar spectra (1)(3). This allows them to identify star clusters as well as whether the individual stars are dwarfs, sub-giants, red-clump stars, etc. However, they don't determine whether the clusters are open or globular. A similar problem in imaging is that of classifying galaxy clusters, not star clusters. In this field there have already been successful implementations of machine learning algorithms to classify galaxy clusters (4) that use images of the galaxies, rather than four-color photometry, to classify them. Huertas et al. (5) are able to classify pre-blue nugget from post-blue nugget galaxies using deep learning and a simple sequential CNN with 3 convolutional layers and 2 fully connected layers. Since they also classify between few classes and only have 3 convolutional layers, we opted for using transfer learning but freezing most of the layers.

3 Dataset and Features

Our dataset consists of images from the Digitized Sky Survey 2 (DSS2)(6) of the locations in the sky that contain open and globular clusters according to the SIMBAD(7) database of the CDS Strasbourg. We will use 4000 images of open clusters and 4000 images of globular clusters. The ratio of the images for the training set, dev set, and test set are as follows: 80% in the training set, 10% in the dev set and 10% in test set. The original resolution of the images were 952×398. To train more efficiently and to fit the standard resolution often used in the architecture selected, $VGG16$, we cropped them to be of size 244×244. Before training, we normalised each element of the data by subtracting from each channel the required amount according to the transfer learning specifications of the VGG16 architecture.

Not all of our images were of the quality of figure 1. In fact, we found out that SIMBAD had mislabeled several star clusters. The proportion of these to good labels was very small, but sufficient that the task was unfeasible by hand.

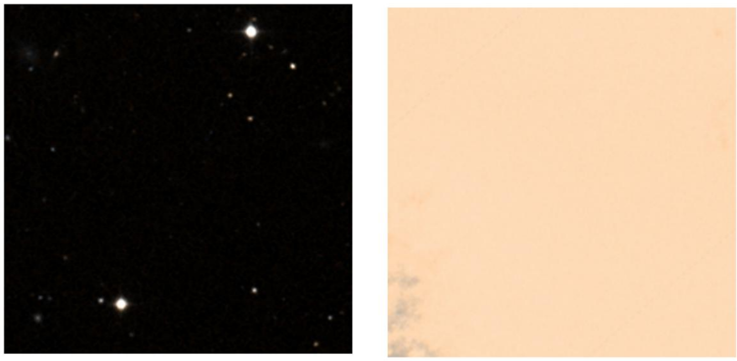

Figure 2: Poorly labeled images from SIMBAD. The left one is supposed to be a globular cluster and the right one an open one.

4 Methods

In selecting the architecture that we will use for what is fundamentally a classification task, we consider pretrained architectures to leverage the strength of preexisting classifiers. We do so with transfer learning. We select the VGG16 (with 138,099,386 trainable parameters) architecture since it is one of the best performing architectures trained on the vast ImageNet dataset. VGG16 architecture works by channeling output through a block of two convolutional layers with 64 filters, a block of two convolutional layers with 128 filters, a block of three convolutional layers with 256 filters, and a block of three convolutional layers with 512 filters. Each convolutional layer is preceded by padding layer of 1 pixel, and each block of convolutional layers are followed by a MaxPooling layer with a window size of 2×2 and a stride of 2. Following the convolutional layers are three fully connected layers, the first two have 4096 channels each. The final has 2 channels(one for each class). Through a Keras implementation, our VGG16 Architecture has 10 frozen layers to specify high level training for two classification outputs: open and globular. We tested from 10 to 13 frozen layers achieving similar results. Based on similar scientific literature on the use of VGG16, we settled on 10 frozen layers as had other research implementations. We used a standard dropout of 0.5, a low learning rate of 10^{-6} and 40 epochs to adjust for overfitting that we were getting in previous tests. Initially, our implementation used a higher learning rate, but the stagnation of low values accuracy and the loss function gave and indication of over fitting.

We trained our data in 16 item minibatches, and 40 epochs. Lengthening the number of epoch helped to raise accuracy drastically.

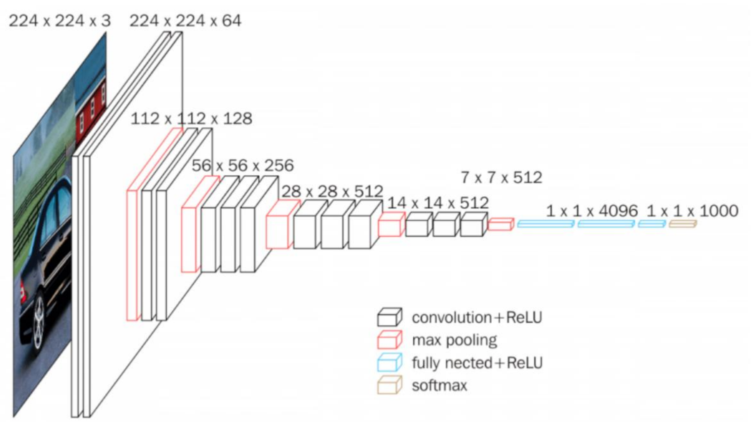

Figure 3: Visualization of the VGG16 layers. Source: Neurohive

We implemented a class activation map visualizer on top of the VGG16 that allowed us to extract the last convolutional network and calculate the guided Grad-CAM image.

5 Results

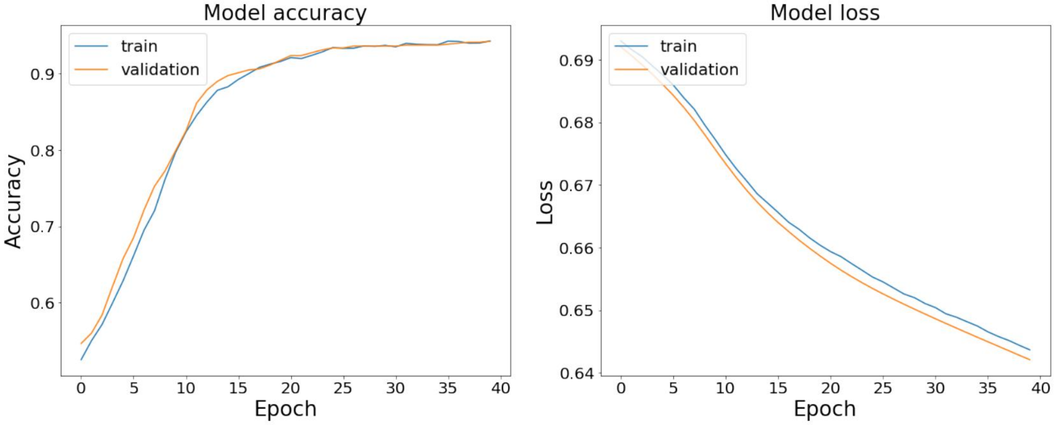

Figure 4: Evolution of the accuracy and the loss function with the epochs.

We achieve very good accuracy of 0.94 after 40 epochs. In an initial part of the project we were using data sets of imbalanced proportions: we had 5 times as many images of globular clusters than open clusters. Therefore, it made sense to calculate the f1 score as a metric to maximize. Since we ended up using 4000 images of open clusters and 4000 images of globular clusters, it did not make much sense to use the f1 score, but for reference we calculated it and it was 0.93. However, the loss function does not decrease as much as desired.

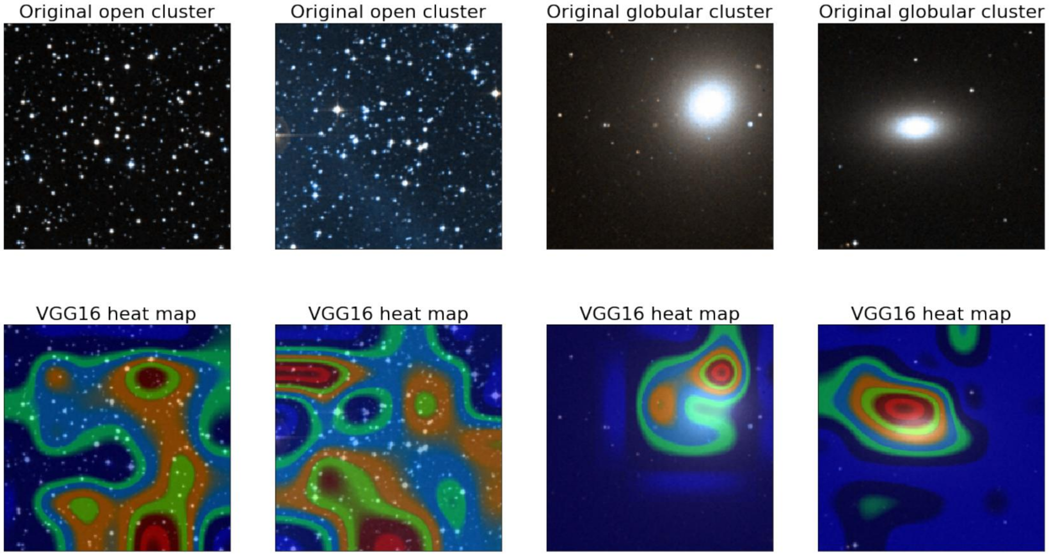

Figure 5: Heat maps of a sample of open and globular clusters from the last convolutional layer of the VGG16.

On the other hand, we achieve very good results for the guided Grad-CAM, since it is able to tell exactly where in the picture the globular cluster is and it finds different features for open and globular clusters. Through the analysis of the results of the Grad−CAM projections, we have found a notable emphasis from the CNN on the intensity and proximity of light sources in an image to aid in binary classification. Dispersed yet relatively clumped stars tend towards an open classification while images with an intense adjacent clustering are classified as globular. This generally matches our expectations. What we find to be a profound result from the Grad-CAM is its ability to seemingly recognize the relationship between stars within a cluster based on their proximity. Heat signatures within the Grad-CAMs highlight what the algorithm deems most significant in the classification. With improvement, this application could also help to define whether a star or group of start belong to a specified cluster or whether they were mischaracterized in the first place.

6 Conclusion/Future Work

We observed that the network focused more on one big clump of light for globular clusters when compared to open clusters, where it found a disperse amount of smaller clumps of luminosity. Future iterations of this project could look into gathering data from other database and include parameters such as the metallicity. Furthermore, if our current implementation was combined with K-means algorithms for finding star clusters, we could find and characterize the new clusters as they are found.

The performance of our network is remarkable given that our initial data was not 100% well labeled. Therefore, our method could also be used to discard those mislabeled images from the original database and be retrained iteratively.

7 Contributions

Data acquisition: both Jordan and Jordi worked on using different methods and packages for data acquisition, until we settled for Aladin.
Architecture: Jordan and Jordi worked on both the preliminary versions of the architecture (simple CNN for binary classification) and the final VGG16. This also includes class activation maps and running the code on the GPU.
Report and poster: both Jordan and Jordi contributed to writing the report and the poster.

8 Acknowledgements

This research has made use of the SIMBAD database, operated at CDS, Strasbourg, France and the DSS2 database from the Space Telescope Science Institute.

Source code available at: **https://bit.ly/2HxLSwh**

References

[1] Y.-m. Cheung, “k*-means — a generalized k-means clustering algorithm with unknown cluster number,” in *Intelligent Data Engineering and Automated Learning — IDEAL 2002* (H. Yin, N. Allinson, R. Freeman, J. Keane, and S. Hubbard, eds.), (Berlin, Heidelberg), pp. 307–317, Springer Berlin Heidelberg, 2002.

[2] L. Searle, A. Wilkinson, and W. G. Bagnuolo, “A classification of star clusters in the Magellanic Clouds.,” , vol. 239, pp. 803–814, Aug 1980.

[3] R. Garcia-Dias, C. A. Prieto, J. S. Almeida, and I. Ordovás-Pascual, “Machine learning in apogee: Unsupervised spectral classification with k-means,” 2018.

[4] J. De La Calleja and O. Fuentes, “Machine learning and image analysis for morphological galaxy classification,” *Monthly Notices of the Royal Astronomical Society*, vol. 349, pp. 87–93, 03 2004.

[5] M. Huertas-Company, J. R. Primack, A. Dekel, D. C. Koo, S. Lapiner, D. Ceverino, R. C. Simons, G. F. Snyder, M. Bernardi, Z. Chen, H. Domínguez-Sánchez, Z. Chen, C. T. Lee, B. Margalef-Bentabol, and D. Tuccillo, “Deep learning identifies high-z galaxies in a central blue nugget phase in a characteristic mass range,” 2018.

[6] Space Telescope Science Institute, “Digitized Sky Survey 2 (DSS2),” 2019.

[7] SIMBAD, Centre de données astronomiques de Strasbourg, “Open cluster (OpCl), Globular cluster (GlGl),” 2019.