Draft version May 25, 2023
Typeset using LaTeX twocolumn style in AASTeX631

Addendum: Creating a Pipeline to Search for Dwarf Nova Eruptions in Globular Cluster M15

Iman Behbehani,[1]

(Update to 705 Thesis Paper)

M. Shara,[2] D. Zurek,[2] E. Glikman,[1] P. Hess,[1]

[1] *Middlebury College Department of Physics*
14 Old Chapel Road
Middlebury, VT 05753, USA
[2] *Department of Astrophysics, American Museum of Natural History*
New York, NY 10024

1. UPDATES

In my previous report for my Physics 705 thesis paper, I developed a catalog of 169 potential variable stars. These candidates were found through the application of a differencing code I written in Python. I ran the code on five images pairs from the *uvm2* filter of the Ultraviolet Optical Telescope from NASA's *SWIFT* telescope in near Earth orbit.

In this extension, I narrowed down this list to determined which candidates were undiscovered. I compiled a list of known RR Lyrae and other variables in Messier 15 using Bhardwaj et al. (2021) and Hoffman et al. (2021). Combined, these papers contain 202 known variable stars, mostly rr lyrae. To reduce my catalog down to only undiscovered stars, I programmed an algorithm to remove all repeat candidates from the list, then, remove all stars present in the Bhardwaj et al. (2021) and Hoffman et al. (2021) catalogs.

This left me with a list of 39 undiscovered variable star candidates, significantly reduced from the original 169. Figure 3 shows the full list of remaining candidates after the data reduction. Figure 3 is divided into different columns based on right ascension and declination coordinates. They are sorted by units of hours, minutes, and seconds. Separating the coordinates into these categories helped sort between them. Due to the small area of the cluster, most only vary by seconds or minutes. The 'num' column represents the index of the star from the original catalog. I left this in to help keep track of stars before and after filtering.

I performed aperture photometry on a series of stars to measure their counts/second. Figure 1 shows the fluctuation in brightness of the faint star as it erupts. Figure 2 shows the brighter star eruption. This measures the amount of photons that hit the telescope detector per second. This is a quick way to measure the relative brightness of the stars before and after eruption. The rr lyra was 4x brighter than the unidentified star at eruption. This means this star is unlikely a dwarf nova, as the relative brightness should be much higher. The star goes from 0.19 to 0.12 counts/second, corresponding to a magnitude difference of 0.5 (note that this magnitude is specific to the Ultraviolet). A faint eruption that not much dimmer than an rr lyra is likely a blue straggler. These are hot, bright stars common in globular cluster, with fluctuations in brightness that are characteristically faint.

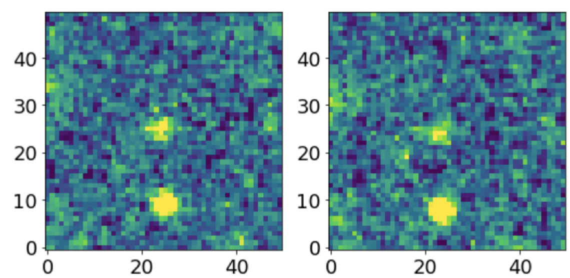


Figure 1. Cutout of an unidentified variable star identified in this study. Note the slight change in brightness between epochs.

The known rr lyra star went from 0.32 to 0.71 counts/second, a magnitude difference of 0.87. This is nearly twice as bright. Such a considerable final and initial brightness is characteristic of an rr lyra in the Ultraviolet. This adds credibility to my results, as the rr lyra acts as expected. This backs up the rest of my conclusion that most of the stars found are rr lyra, with some blue stragglers and other variables, with hopefully at least one dwarf nova.

The code to filter the table and find source magnitudes can be found in the appendix.

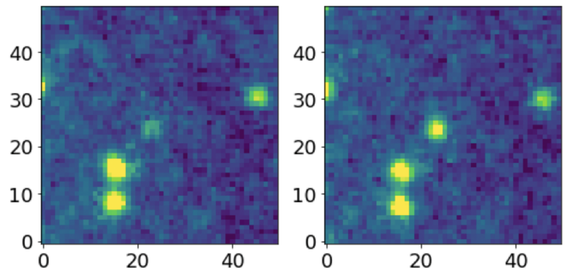


Figure 2. Cutout of a known rr lyra star visible in this study. Note the dramatic change in brightness between epochs.

2. FURTHER RESEARCH

Now that we can find the magnitude of a star using aperture photometry, we can determine the stellar types. Finding the magnitudes across two filters can give us all necessary information to plot the star on an HR diagram. Cataclysmic variables, blue stragglers and rr lyrae occupy highly distinct locations on the HR diagram. Plotting the stars will finalize our understanding of the catalog, resulting in a full list of previously unidentified variable stars alongside their stellar types.

3. ACKNOWLEDGEMENTS

I would like to start by thanking the Middlebury College Center for Community Outreach for the Academic Outreach Grant I was awarded to complete my thesis. This funding made it possible for me to conduct my work in person at the American Museum of Natural History, and get in touch with the greater community of astrophysicists.

Additionally, this project would not have been possible without multiple research advisors. Professor Eilat Glikmnan supervised this project and provided irreplaceable guidance on the coding and research process. Professor Paul Hess took on this project and made it possible for me to continue my research while Professor Glikman was on sabbatical. Without his help, I would not have been able to analyze and interpret my 705 data. Dr. Michael Shara and Dr. David Zurek took me on as a research student last summer for a 3 month project that turned into a one year thesis. I am grateful for their continued support, giving me the opportunity to do such engaging astrophysics research at the undergraduate level.

REFERENCES

Bhardwaj, A., Rejkuba, M., Sloan, G. C., Marconi, M., & Yang, S.-C. 2021, ApJ, 922, 20, doi: 10.3847/1538-4357/ac214d

Hoffman, A. M., Murakami, Y. S., Zheng, W., Stahl, B. E., & Filippenko, A. V. 2021, MNRAS, 502, 818, doi: 10.1093/mnras/stab010

	num	ra hh	ra mm	ra ss	dec hh	dec mm	dec ss
0	0.0	21.0	29.0	31.5624	12.0	11.0	35.0552
1	1.0	21.0	29.0	50.9979	12.0	19.0	3.0071
2	2.0	21.0	29.0	53.184	12.0	12.0	31.0676
3	4.0	21.0	29.0	47.72	12.0	11.0	32.0163
4	5.0	21.0	30.0	15.2738	12.0	11.0	35.0537
5	6.0	21.0	30.0	11.682	12.0	10.0	34.09
6	7.0	21.0	30.0	5.2819	12.0	10.0	45.0224
7	8.0	21.0	30.0	5.981	12.0	6.0	59.0769
8	11.0	21.0	30.0	1.0869	12.0	10.0	48.0037
9	12.0	21.0	30.0	6.8676	12.0	9.0	49.0078
10	13.0	21.0	30.0	0.9503	12.0	8.0	58.0761
11	14.0	21.0	30.0	0.8598	12.0	8.0	39.0554
12	16.0	21.0	29.0	53.4079	12.0	8.0	33.0603
13	17.0	21.0	29.0	59.2101	12.0	9.0	23.033
14	20.0	21.0	30.0	12.6076	12.0	8.0	35.0649
15	21.0	21.0	29.0	58.6266	12.0	5.0	49.0702
16	23.0	21.0	30.0	0.8723	12.0	14.0	24.0216
17	24.0	21.0	30.0	7.0533	12.0	11.0	55.0882
18	26.0	21.0	29.0	50.8489	12.0	9.0	1.0492
19	27.0	21.0	29.0	47.9457	12.0	8.0	45.001
20	28.0	21.0	30.0	3.8663	12.0	8.0	40.088
21	30.0	21.0	29.0	51.5984	12.0	5.0	51.0975
22	31.0	21.0	29.0	53.3774	12.0	5.0	1.0162
23	32.0	21.0	30.0	4.5136	12.0	11.0	15.0007
24	33.0	21.0	29.0	51.3683	12.0	14.0	58.954
25	34.0	21.0	30.0	2.4957	12.0	10.0	55.04
26	35.0	21.0	29.0	53.9536	12.0	11.0	37.9
27	36.0	21.0	29.0	46.9849	12.0	11.0	18.122
28	38.0	21.0	30.0	16.5424	12.0	9.0	6.218
29	41.0	21.0	30.0	0.1339	12.0	8.0	2.007
30	42.0	21.0	30.0	0.4702	12.0	6.0	29.72
31	43.0	21.0	30.0	4.4018	12.0	5.0	2.368
32	44.0	21.0	30.0	11.143	12.0	4.0	55.752
33	45.0	21.0	29.0	44.7379	12.0	9.0	4.632
34	47.0	21.0	29.0	41.7057	12.0	5.0	45.221
35	48.0	21.0	30.0	9.574	12.0	8.0	34.94
36	51.0	21.0	29.0	52.9878	12.0	7.0	33.0592
37	54.0	21.0	29.0	55.6849	12.0	7.0	13.0822
38	56.0	21.0	30.0	11.7522	12.0	3.0	44.0531

Figure 3. Table representing the coordinates of new unidentified variable stars in Messier 15.

APPENDIX

A. CODE

In [1]:
```
# The following code will allow the user to compare a list of stellar objects with
# a list of known objects. The intention is to identify which sources from
# the original list are undiscovered. This is written for a case in which
# the comparison dataset is a list of known rr lyra.
#
# Repeat sources will be removed, resulting in a final table of
# unique objects.


# Author: Iman Behbehani
# Date: Spring 2023
```

In [2]:
```
# import necessary packages
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

In [3]:
```
# open the catalog and comparison files
text_file = open('/Users/iman/Documents/Midd_Coursework/catalog.csv')
text_file2 = open('/Users/iman/Documents/Midd_Coursework/rrlyrae.csv')
```

In [4]:
```
table = pd.read_csv(text_file)
```

In [5]:
```
catalog = table[['ra hh', 'ra mm','ra ss','dec hh','dec mm','dec ss']].copy()
```

In [6]:
```
table2 = pd.read_csv(text_file2)
```

In [7]:
```
comparison = table2[['ra hh', 'ra mm','ra ss','dec hh','dec mm','dec ss']].copy()
```

remove duplicates

In [8]:
```
### This double for loop removes repeat candidates within the specified error

#creating a new dataframe for non repeated stars
df2 = pd.DataFrame(columns = ['num','ra hh', 'ra mm', 'ra ss', 'dec hh', 'dec mm', 'dec ss'
#specify x and y errors
xerr=3
yerr=5

#sorting through
for i in range(len(catalog)):
    rep = False

    for j in range(len(catalog)): # check with rest of catalog
        #ra min in col 2, dec min in col 6
        if((i!=j)and((catalog['ra ss'][j]-xerr)<catalog['ra ss'][i]<(catalog['ra ss'][j]+xe
           and((catalog['dec ss'][j]-yerr)<catalog['dec ss'][i]<(catalog['dec ss'][j]+yerr)
           and((catalog['ra mm'][i]==catalog['ra mm'][j])and(catalog['dec mm'][i]==catalog[
                rep = True

    if(rep == False):
        df2 = df2.append({'num': i,
        'ra hh' : catalog['ra hh'][i], 'ra mm' : catalog['ra mm'][i], 'ra ss' : catalog['ra
        'dec hh': catalog['dec hh'][i], 'dec mm' : catalog['dec mm'][i], 'dec ss' : catalog

```

remove known stars. in this case, rr lyra

In [10]:
```
# compare with known stars

# create new dataframe for undiscovered stars
df3 = pd.DataFrame(columns = ['num','ra hh', 'ra mm', 'ra ss', 'dec hh', 'dec mm', 'dec
# define error
xerr=3
yerr=5


for i in range(len(df2)): #rows
    rep = False

    for j in range(len(df2)): #rows
    #ra min in col 2, dec min in col 6
        if(((comparison['ra ss'][j]-xerr)<df2['ra ss'][i]<(comparison['ra ss'][j]+xerr)
           and((comparison['dec ss'][j]-yerr)<df2['dec ss'][i]<(comparison['dec ss'][j]
           and((df2['ra mm'][i]==comparison['ra mm'][j])and(df2['dec mm'][i]==compariso
                rep = True
                print(i)

    if(rep == False):
        df3 = df3.append({'num': i,
        'ra hh' : df2['ra hh'][i], 'ra mm' : df2['ra mm'][i], 'ra ss' : df2['ra ss'][i]
        'dec hh' : df2['dec hh'][i], 'dec mm' : df2['dec mm'][i], 'dec ss' : df2['dec s


pd.set_option("display.max_rows", None)
```

this section shows how to zoom in on a spectific star

In [12]:

```
# import
import astropy

from astropy.io import fits
from astropy import wcs
from astropy.nddata import Cutout2D
from astropy import units as u
import numpy as np
import matplotlib.pyplot as plt
import astropy.io.fits as fits
from astropy.visualization import simple_norm
```

In [13]:

```
def cutout (x, y, data, hdu, filename):

    #center of cutout area coordinates
    xcoord = x
    ycoord = y

    #size of cutout image
    xsize = 50
    ysize = 50

    #cutout function (see astropy.nddata for documentation)
    cutout = Cutout2D(data, (xcoord,ycoord),(ysize,xsize))

    #Show our new image
    norm = simple_norm(cutout.data, 'sqrt', percent=98.1) #linear, log, sqrt

    plt.imshow(cutout.data, norm=norm, origin='lower', cmap='viridis')
    plt.show()

    #Writing to FITS file
    fits_newfile_name = filename

    vhdr = hdu[0].header
    print(type(cutout))

    fits.writeto(fits_newfile_name,cutout.data,vhdr,overwrite=True)

    return cutout.data

```

In [14]:

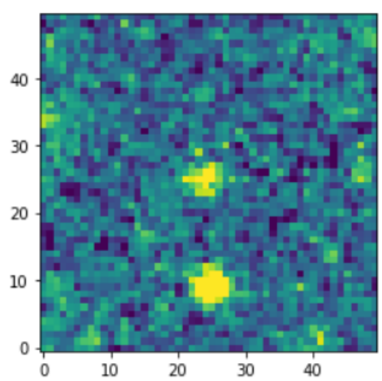
```
# open the two epochs

hdu1010 = fits.open("/Users/iman/Documents/Midd_Coursework/Fall_2022/704/amnh/swiftuvot
hdu1010[1].header['EXTNAME'] = 'SCI'
hdu1010[1].header['EXTVER'] = 1
data1 = hdu1010[1].data #this is already a numpy.ndarray
header10 = hdu1010[0].header
header11 = hdu1010[1].header

hdu3010 = fits.open("/Users/iman/Documents/Midd_Coursework/Fall_2022/704/amnh/swiftuvot
hdu3010[1].header['EXTNAME'] = 'SCI'
hdu3010[1].header['EXTVER'] = 1
data2 = hdu3010[1].data
header20 = hdu3010[0].header
header21 = hdu3010[1].header

data = np.array(data1)
```

In [15]:
```
#making backup. mapping 1 onto 2
hdu1010a = hdu1010
data1a = hdu1010[1].data
header10a = hdu1010[0].header
header11a = hdu1010[1].header

hdu3010a = hdu3010
data2a = hdu3010[1].data
header20a = hdu3010[0].header
header21a = hdu3010[1].header
```

In [16]:
```
# scale for exposure
header11['TELAPSE']
exp = header11["TELAPSE"]
data11a = data1a/exp

header21['TELAPSE']
exp2 = header21["TELAPSE"]
data22a = data2/exp2
```

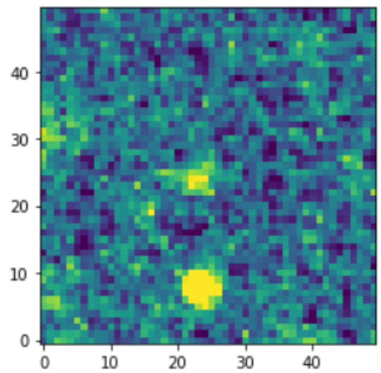
plot star

In [17]:
```
#making the cutout
# replace with fits file directory
cutout35 = fits.open("/Users/iman/Documents/Midd_Coursework/Fall_2022/704/amnh/swiftuvo

#picking the center
one = np.zeros((1, 2))
one[0] = [755.82003, 396.32866]

#saving to drive
filename = "/Users/iman/Documents/Midd_Coursework/Fall_2022/704/amnh/swiftuvot2/cutty1.
cut1 = cutout(one[0][0], one[0][1], data11a, cutout35, filename)
```




```
<class 'astropy.nddata.utils.Cutout2D'>
```

In [18]:
```
#second cutout
cutout42 = hdu3010a

#picking the center
one = np.zeros((1, 2))
one[0] = [711.48275, 293.03799]

#saving to drive
filename = "/Users/iman/Documents/Midd_Coursework/Fall_2022/704/amnh/swiftuvot4/cutty2.
cut2 = cutout(one[0][0], one[0][1], data22a, cutout42, filename)
```




```
<class 'astropy.nddata.utils.Cutout2D'>
```

find counts/second of source

In [19]:
```
# repeat all sources you'd like to take a closer look at
```

In [20]:
```
from photutils.aperture import CircularAperture
from photutils.aperture import aperture_photometry
```

In [21]:
```
aperture = CircularAperture((755.82003, 396.32866), r=5.0)
phot_table = aperture_photometry(data11a, aperture)
phot_table['aperture_sum'].info.format = '%.8g'  # for consistent table output
print(phot_table)
```

```
 id  xcenter   ycenter  aperture_sum
       pix       pix                
--- --------- --------- ------------
  1 755.82003 396.32866   0.18538495
```

In [22]:
```
aperture2 = CircularAperture((711.48275, 293.03799), r=5.0)
phot_table = aperture_photometry(data22a, aperture2)
phot_table['aperture_sum'].info.format = '%.8g'  # for consistent table output
print(phot_table)
```

```
 id  xcenter   ycenter  aperture_sum
       pix       pix                
--- --------- --------- ------------
  1 711.48275 293.03799    0.1235919
```