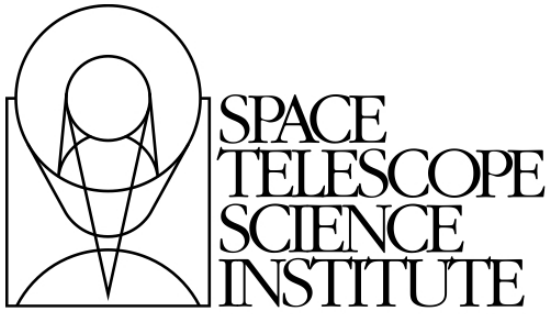
Version 1.3
April 2014

HST Legacy Extragalactic UV Survey

Cluster classification and visualization tool



Space Telescope Science Institute
3700 San Martin Drive
Baltimore, Maryland 21218

Operated by the Association of Universities for Research in Astronomy, Inc., for the National Aeronautics and Space Administration

Revision History

Version	Date	Editor and Contributors
1.3	April 2014	Leonardo UBEDA, Brad WHITMORE, Stacey BRIGHT
1.2	April 2014	Leonardo UBEDA, Angela ADAMO, Brad WHITMORE, Jenna RYON
1.1	March 2014	Leonardo UBEDA, Linda SMITH
1.0	March 2014	Leonardo UBEDA

This is an unofficial document developed for the LEGUS cluster team.

Send comments and questions to:
Space Telescope Science Institute
3700 San Martin Drive
Baltimore, Maryland 21218
e-mail: lubeda@stsci.edu

Contents

1 Installation and dependencies **3**

1.1 Installation 3

1.1.1 What software do I need? 3

1.1.2 Dependencies 4

2 Cluster catalogue format **5**

2.1 Formatting the catalogue 5

3 Cluster classification **7**

3.1 Executing the main code 7

3.1.1 Option [ID number] 8

3.1.2 Option [a = add] 11

3.1.3 Option [q = quit] 11

3.2 The source catalogue 11

4 Troubleshooting **13**

4.1 Solving some common problems 13
4.1.1 ImportError: No module named ds9 13
4.1.2 I installed PyDS9 but I get ImportError: No module named ds9 13
4.1.3 ImportError: No module named pyraf 13
4.1.4 The RGB frame colours are not well balanced 14
4.1.5 Task imexamine is not working 14
4.1.6 WARNING: Moffat fit did not converge 14

1

Installation and dependencies

1.1 Installation

1.1.1 What software do I need?

The supported software is provided in a zipped folder that contains the following items:

FILE	USE
legus_cluster_classification.py	cluster visualization and classification code
legus_cluster_classification.input	input file for the main code
legus_cluster_table_formatter.py	code that allows the user to format a simple input list
ngc6503.xy	example of input list with cluster coordinates (test only)
legus_clusters.pdf	installation guide and tutorial
cheatsheet.pdf	simple cheat sheet

Unpack the folder in the directory of your choice. You will also need the LEGUS drizzled mosaics from the LEGUS website. Please download the NGC 6503 files and place the FITS files in the same working directory.

1.1.2 Dependencies

The Python codes have been tested on Mac desktops and laptops only. You will need your operating system to be version 10.8.5 or later (Mountain Lion or Mavericks).

The provided Python codes depend on the following Python packages that MUST be installed in your system before executing anything: numpy, sys, os, string, time, pyraf, and ds9.

In order to verify that those are already installed in your computer, open a terminal and type:

```
UNIX> python
>>> import numpy
```

and the same for each package. They should all load without any errors. Once you are certain that those packages are on your computer, exit Python with the command

```
>>> exit()
```

In the event that one (or some) of the packages does (do) not exist, you will have to install it (them) yourself in order to proceed. Most likely you will not have pyds9 in your system, so you might get the error

```
ImportError: No module named ds9
```

This means that you need to install the package ds9 in your system. You will need pyds9 version 1.7 or later for the visualization tool to work. Follow the instructions in the official pyds9 website:

http://hea-www.harvard.edu/RD/pyds9/

and see the Troubleshooting chapter of this tutorial.

2

Cluster catalogue format

2.1 Formatting the catalogue

In order to classify clusters according to morphology, we first need a list with their approximate position in a suitable reference system. We are using the reference system defined by the LEGUS astrodrizzled mosaics.

This input list is very simple. It is a text file with two columns: the X coordinate and the Y coordinate. As an example, take a look at the provided file called ngc6503.xy.

Once a set of potential clusters is identified, we need to create an input table with a proper format. This is achieved with the python code

legus_cluster_table_formatter.py

Open a terminal and move to the directory that contains the file ngc6503.xy and the table formatter. The input file is ngc6503.xy. The output formatted catalogue should have a personalized name e.g. ngc6503_leonardo.txt

```
UNIX> python legus_cluster_table_formatter.py -in ngc6503.xy -out ngc6503_leonardo.txt
```

The code will create the new file (ngc6503_leonardo.txt) in the working directory. You may open this file with any text editor. The first three lines should look like this:

```
1   |       2604.0   |       1786.0   |       0   |         no comment
2   |       2637.0   |       1821.0   |       0   |         no comment
3   |       2602.0   |       1846.5   |       0   |         no comment
```

The first column is a cluster ID number. The second column is the X position. The third column is the Y position. The fourth column is the class number, and the last column is for optional comments.

3

Cluster classification

3.1 Executing the main code

Once the list of clusters is properly formatted, we may start the cluster classification. The main Python code for doing this is:

```
legus_cluster_classification.py
```

This code needs to know where your files are placed. We do this with a simple input file to keep everything organized. Open the text file

```
legus_cluster_classification.input
```

with the editor of your choice. This file looks like this

```
/your/path/here/                 #  working directory
ngc6503_leonardo.txt             #  input catalog
ngc6503_uvis_f336w_sci.fits      #  name of input image in the B channel
ngc6503_uvis_f555w_sci.fits      #  name of input image in the G channel
ngc6503_uvis_f814w_sci.fits      #  name of input image in the R channel
```

There are five rows, each with a piece of information that is needed by the main code:

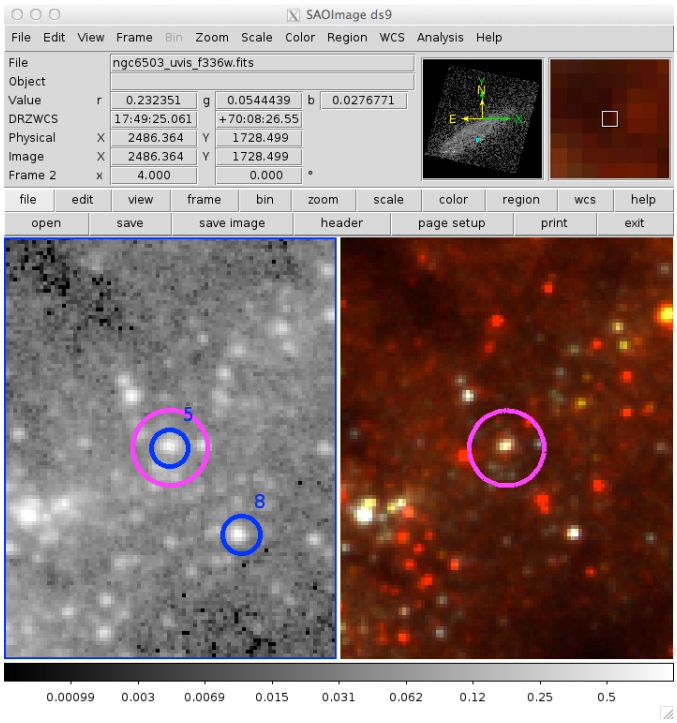


FIGURE 3.1— Screen capture of PyDS9 showing one cluster ready to be classified. The left frame shows a grey-scale F555W image and on the right is the RGB frame.

First row: this is where you tell the code where your files are located. Enter the full path.
Second row: this is the name of the formatted cluster catalog.
Third-fifth rows: these are the names of the astrodrizzled mosaics that DS9 will use to generate an RGB image on-the-fly. You may choose any combination, as long as there are three images.

Do not erase the "#" sign from any of the rows. Keep this file in the same directory as the main code. Now we are ready to execute the main code. This is done from the Terminal by simply writing

```
UNIX>  python legus_cluster_classification.py
```

The code will first output some information that was read from the input file. Make sure that this is correct. Then it provides three options from the command line. We will explain each option in the following sections.

3.1.1 Option [ID number]

From your formatted catalogue, you may choose which potential candidate to classify first. Just type in the ID number which is the first column in the catalogue and type return. This should open DS9 and display two images: a grey-scale version of the image in the green band (image in filter F555W), and a three colour (RGB) image in another frame (Figure 3.1).

The images will be synchronized and zoomed into the selected object. The first time you start the code, you will see that the object is marked with a blue circle with a radius of 4 pixels and surrounded by a

magenta circle. The magenta circle indicates the current object. Note that these circles are DS9 regions. You may select the circles and move them around for your convenience. The colour coding is explained in the table below.

You should also notice the blinking pointer, indicating that you are within PyRAF and running the task imexamine. This means that you have all the powerful tools provided by this task to analyze the cluster and surrounding area. For example, focus your pointer on the grey-scale image and place your pointer on top of the cluster and press the letter "E" in your keyboard. You should get a contour plot in a new window. Now press on the letter "R" and you will get a radial profile. Finally, press on the letter "S" for a surface plot. For a complete description, please search the help pages of the task imexamine. A simple imexamine output is also printed to the Terminal screen. The information provided contains several useful parameters such as the FWHM that will help you in your classification.

The potential clusters should be classified according to the following options:

CLASS	COLOUR	MEANING
class 0	blue	The source is not classified yet.
class 1	red	Centrally concentrated objects, extended PSF profile, isolated (the photometry is not affected by nearby bright objects).
class 2	yellow	Asymmetric objects (massive OB associations), multiple peaks with underlying cluster (diffuse light).
class 3	green	Faint objects, multiple peaks that are likely two nearby single stars. This class of objects could contain low mass clusters, small associations.
class 4	cyan	False detections, i.e. bright foreground stars, background galaxies, clusters at the edge of the chip, or bad pixels. These objects will be removed from the final catalogue.

For your guidance, refer to Figure 3.2 to see examples of the cluster classes as defined by the LEGUS team. For each class, you have four figures: the F555W image of the cluster as displayed in DS9 with a linear scale, the contour, surface, and radial plots produced using the task imexamine.

Classification Checklist

☐ Stretch image contrast with DS9 to find several "stars" (things that come and go at the same rate). You can achieve this by right-click and drag your mouse through the DS9 frame. Use the task imexamine and type the letter "R" to see the stellar PSF, then do the same on top of the source you want to classify. Are the FWHM similar? Is the PSF of the source more extended? Two different FWHM measurements are shown at the bottom of the window. The one before the last is a MOFFAT fit, the last one is a DIRECT fit. Please, use the former when you compare FWHM between stars and the source to classify.

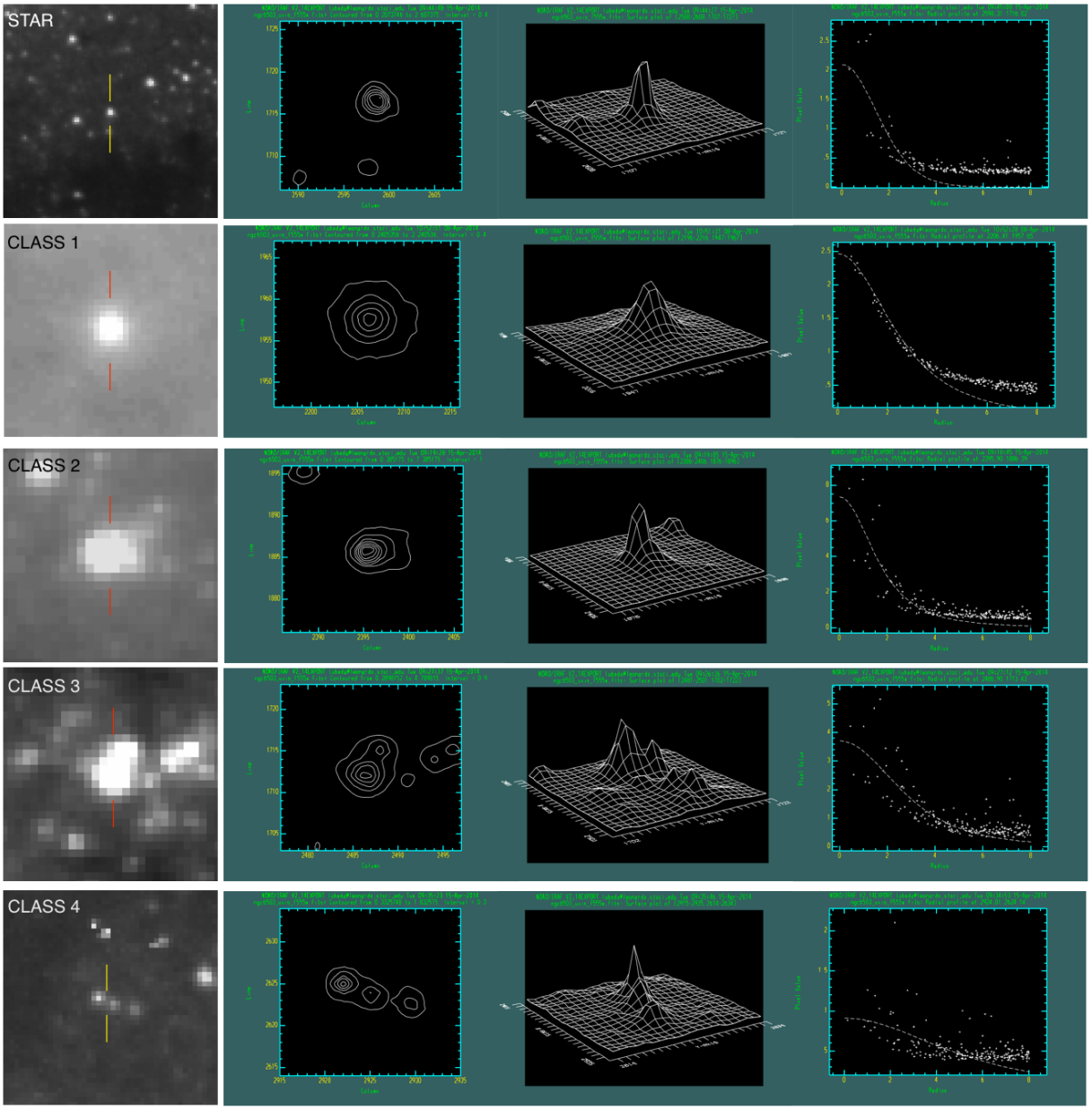


FIGURE 3.2— LEGUS classification system for clusters. The figures show examples of classes 1, 2, 3, and 4. For comparison, a typical star is also shown. The contour, surface, and radial plots for each case are the direct output from the task imexamine.

☐ Stretch image contrast to see if cluster candidate is "softer" (shows up later and ends up being bigger than stars that show up at the same time) and more extended.

☐ Hit the the letters "S" (surface plot) and "E" (contour plot) and look for evidence of pairs or asymmetries. Stars have very concentrated contours, clusters are more extended. See Figure 3.2 .

□ Look at the color image to see if there is evidence for a pair (e.g. color gradient).

To quit imexamine and continue with the classification process, click on the cluster and type the letter "Q". This will return your pointer to the Terminal and you will see the available options. Enter your class number and then you will be prompted to enter an optional comment. Keep this comment brief, maximum of 50 charachters long. If you have no comment, just press enter. This will return you to the original prompt that will allow you to classify another source. In the case that you are satisfied with the current classification, just type the equal sign and press enter to move on to the next source.

3.1.2 Option [a = add]

In the possible case in which you identify a cluster that is not on the list, you will be able to easily add it to the bottom of the list with the option "[a = add]". Before doing this you should note the coordinates of the source from DS9. The code will request the X coordinate and the Y coordinate from you. Simply type those numbers in and press return. The code will add a line automatically to the bottom of the list, increasing the ID number by one. You may then proceed to classify this new source, following the instructions above.

3.1.3 Option [q = quit]

If you want to quit the code seamlessly, this is the only opportunity you have. Just type in the letter "q" and the code will terminate.

3.2 The source catalogue

The source catalogue will be modified and updated by the main code each time the user makes a change. The current working directory contains the latest version of this catalogue. Once the user is finished classifying all the sources in the catalogue, the "class" column should be one of the following numbers: 1, 2, 3, or 4.

If you are an experienced IRAF and DS9 user, then you know that those codes may be unstable sometimes and they are prone to crash frequently. For peace of mind of the user, a backup version of the source catalogue is saved in the backup/ folder, everytime an update is made. The catalogue is saved with the following name:

```
catalogue_YYYYMMDDHHMMSS.txt
```

If, for any reason, the working catalogue gets corrupted, you can always use the last backuped version and start over. In the worst case you will only lose one classification.

4

Troubleshooting

4.1 Solving some common problems

4.1.1 ImportError: No module named ds9

This means that you need to install the package ds9 in your system. Follow the instructions in the official pyds9 website:

http://hea-www.harvard.edu/RD/pyds9/

4.1.2 I installed PyDS9 but I get ImportError: No module named ds9

You need to make sure that your Python distribution sees the newly added package. This should be solved by setting the variable environment PYTHONPATH

```
UNIX> setenv PYTHONPATH  /your/pyds9/folder/
```

4.1.3 ImportError: No module named pyraf

If you are at STScI you may need to activate PyRAF issuing the command

```
UNIX> irafx
```

If you are outside STScI, you may need to set your environment variables for PyRAF. Try placing your login.cl file in the current working directory.

4.1.4 The RGB frame colours are not well balanced

When PyDS9 generates an RGB frame, it will display one image per colour channel. The scale range selected for those are hard-coded in the Python script, but they can be easily modified using the small RGB window that pops up. This window shows two coloumns: Current and View. The Current column allows you to select which is the current active channel. If the tick is on Red, it means that you you may modify the Red channel at this time. Use your mouse to drag and change the stetch as you would in a single channel frame. The View column simply states which channel(s) is(are) displayed. I recommend classifying a couple of clusters before making these changes. These changes will remain current throughtout your session and will be lost when you quit the Python code.

4.1.5 Task imexamine is not working

The Terminal-Python-DS9 interaction is VERY finicky. If you focus your mouse on the DS9 window and the cursor is not blinking, then go back to the Terminal and you should see the prompt

```
display frame (1:) (1):
```

Accept the command by simply pressing the key "return". DO NOT enter the number 1.

4.1.6 WARNING: Moffat fit did not converge

Sometimes, when you are examining a source with the task imexamine and you try to make a radial plot you might get a warning that the Moffat fit did not converge. This has to do with the distribution of flux in the region that you are trying to fit. It is just a warning.