Detecting Dual Nuclei in closely merging Galaxies using Machine Learning Techniques over SDSS Images

Bhavesh Mukheja - 21PH10008

Summer Internship Presentation

Indian Institute of Technology Kharagpur

Department of Physics

November 27th 2024

Contents

Introduction

Introduction

- I had my Summer Internship from May 13th 2024 - July 19th 2024 at Indian Institute of Astrophysics, Bangalore under Dr. Mousumi Das

- My role involved conducting a thorough review of prior research, performing a comprehensive literature review, and contributing to the development of a machine learning model capable of detecting Dual Active Galactic Nuclei (DGNs) within a sample of SDSS (Sloan Digital Sky Survey)images.

Motivation

Motivation

- Although galaxy mergers are common, the detection of dual Active Galactic Nuclei (AGN) is rare.
- Their detection is very important as they help us understand the formation of supermassive black hole (SMBH) binaries, SMBH growth and AGN feedback effects in multiple nuclei systems.
- There is thus a need for an algorithm to do a systematic survey of existing imaging data for the discovery of DANGs.

Previous Research

Previous Research

- In the previous research, the **GOTHIC** (Graph-Boosted iterated Hill Climbing) algorithm was developed to detect DGNs in SDSS DR16 Data.
- The study utilized **1 million galaxies blindset from SDSS DR16**, focusing on the **u, g, r, and i bands**, while excluding the noisy **z band** to ensure reliable image quality
- The galaxies were examined within a **redshift range of 0 to 0.75**, with confirmed DNGs predominantly concentrated around **z ≈ 0.073**.

GOTHIC Algorithm Explanation

GOTHIC Algorithm Explanation

The **GOTHIC** algorithm detects DANGs in galaxy images based on their visual and structural features. Here's an outline of the algorithm's steps.

1. Image Normalization and Smoothing
   - The algorithm starts by normalizing the input images to standardize their brightness.
   - Gaussian Blur is used for smoothing the images to suppress noise and enhance features like bulges.

GOTHIC Algorithm Explanation

- Log normalization of all pixel values is applied in the cutout followed by appropriate scaling of the image within the range of [0, 255]

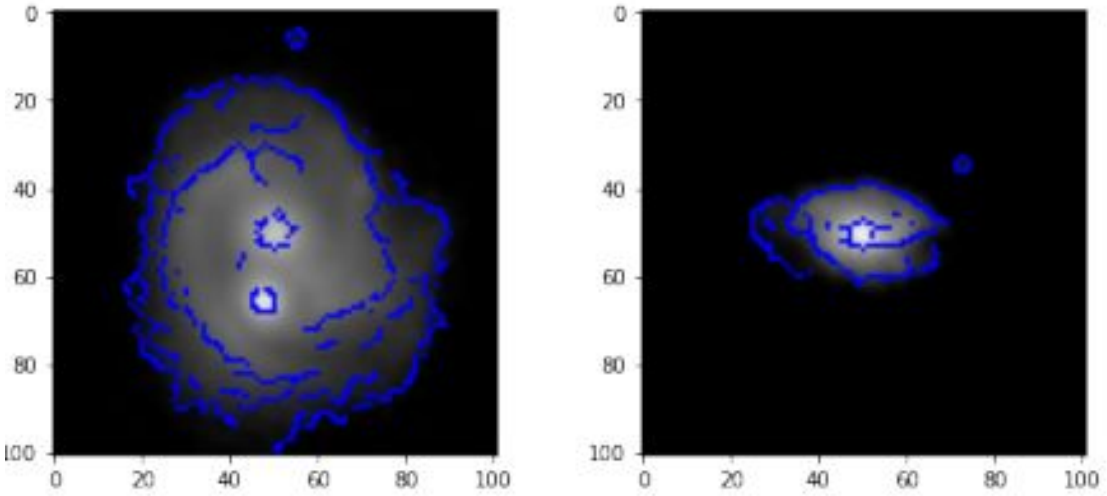
$$I_{\log}(x,y)=\log(I(x,y)+1)$$

$$I_{scaled}(x,y) = \frac{I_{log}(x,y)-min(I_{log})}{max(I_{log})-min(I_{log})} \times 255$$

GOTHIC Algorithm Explanation

2 Edge Detection

- The **Canny edge detection algorithm** is used to identify the boundaries of the galaxy in the 40" × 40" SDSS image cutout.

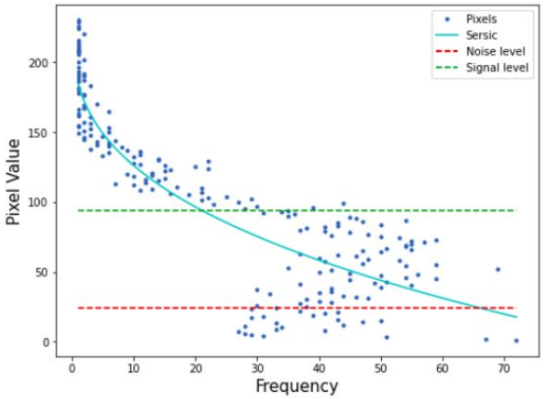


Examples of Edges detected by Canny

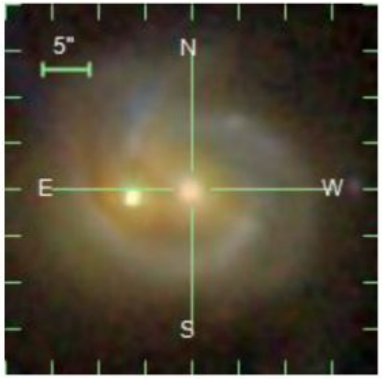
GOTHIC Algorithm Explanation

3. Light Profile Fitting

- The pixel intensity histogram of the galaxy is fitted to a **Sérsic light profile**, which describes how light intensity varies with radius.
- The **Sérsic index** helps distinguish between single and double nuclei systems.



Sersic Fit



1237667734504407133

Example of a Seric fit of a DGN

GOTHIC Algorithm Explanation

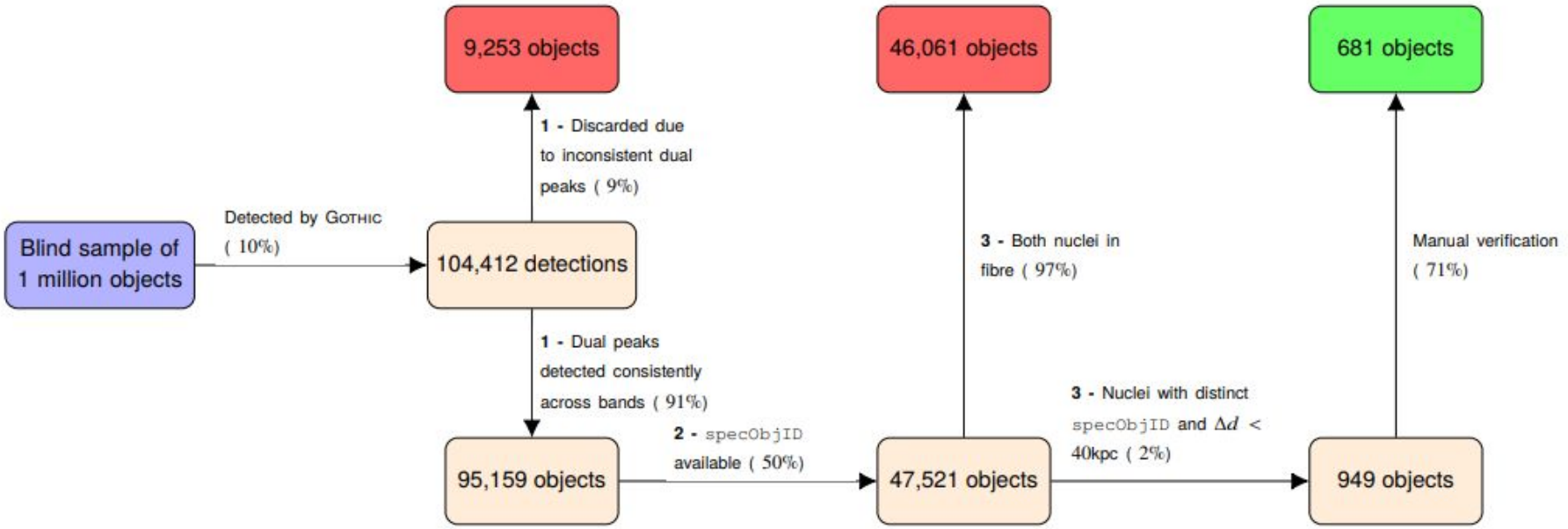
4. Determination of the Search Region
   - Based on the light profile, a pixel intensity cutoff value is defined.
   - Only pixels above this cutoff are considered part of the galaxy's high-intensity regions, narrowing the search for nuclei.
5. Iterated Hill Climbing
   - A peak-finding algorithm is applied to locate regions of maximum brightness within the search region.
   - This step ensures that the detected peaks correspond to potential nuclei.

GOTHIC Algorithm Explanation

6. Final Classification
   - **DNG**: If two distinct, well-separated peaks are identified.
   - **Single Nucleus Galaxy**: If only one peak is found.
   - **Three Peaks**: Very few images contain three peaks

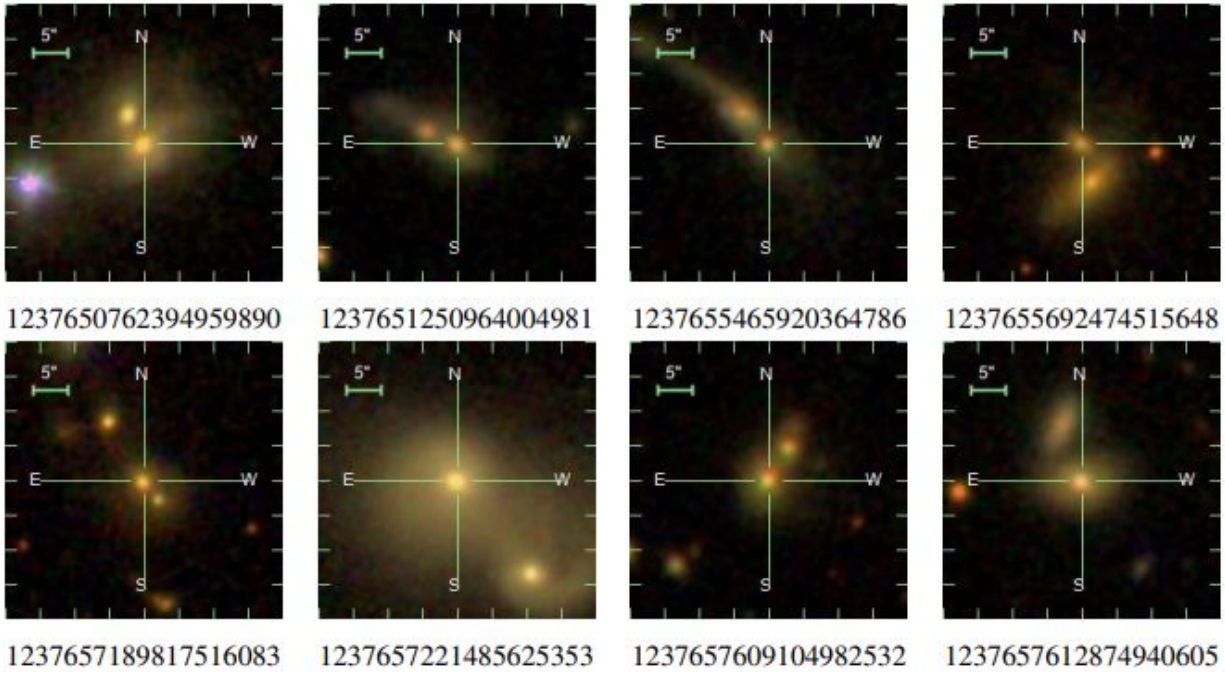
Results and Conclusion

Results and Conclusion



Flowchart demonstrating the filtration of the blind sample down to the visually confirmed sample. The bold numbers refer to the filtration steps in Section 4 and the numbers in parenthesis indicate the percentage reduction in the sample size caused by each filtration step

Results and Conclusion



Panel of a subset of 6 out of the 681 discovered DGNs (with SDSS objIDs listed)

Reason to develop a Machine Learning Algorithm

Reason to develop a Machine Learning Algorithm

- GOTHIC Algorithm is a pure image processing algorithm and does detect dual peaks in the 46,061 sample but is unsure if the there is single galaxy or two because of the bulge like shape.

- Our main aim of the project is to study these 46,061 samples of galaxies declared as probable candidates for DGNs by GOTHIC and develop a Machine Learning Algorithm that can predict with high accuracy if there are Dual Nuclei in the fibre or not.


1237648720146333848


1237651250975670525


1237655107837362315


1237653613720502346

Some Examples showing probable DGNs candidates detected by GOTHIC

Contribution to the project during internship period

Contribution to the project during internship

- Data Analysis
  - Rerunning the GOTHIC Algorithm and obtaining the data of interest i.e. 46,081 probable DGNs candidates
  - Developing the python code to get images with dual peaks across u,g,r & i bands.
  - Filtered out the objects which are present in all the 4 bands (u,g,r,&i) **~2.5k** and in 2 bands (r&i)

Contribution to the project during internship

```python
# Define the directory containing the CSV files
csv_dir = '/content/Double in 4 bands'

# List all CSV files in the directory
csv_files = [f for f in os.listdir(csv_dir) if f.endswith('.csv')]

# Initialize an empty list to hold DataFrames
dataframes = []

# Iterate over the CSV files and read them into DataFrames
for file in csv_files:
    file_path = os.path.join(csv_dir, file)
    df = pd.read_csv(file_path)
    dataframes.append(df)

# Concatenate all DataFrames into a single DataFrame
merged_df = pd.concat(dataframes, ignore_index=True)

# Save the merged DataFrame to a new CSV file
output_file = os.path.join(csv_dir, '47kSDSSSample.csv')
merged_df.to_csv(output_file, index=False)

print(f"All CSV files have been merged and saved to {output_file}")
```

```
All CSV files have been merged and saved to /content/Double in 4 bands/doubleIn4bandsMain.csv
```

Code snippet for finding "DOUBLE" Peaks in u,g,r,i bands in the raw_doubles data

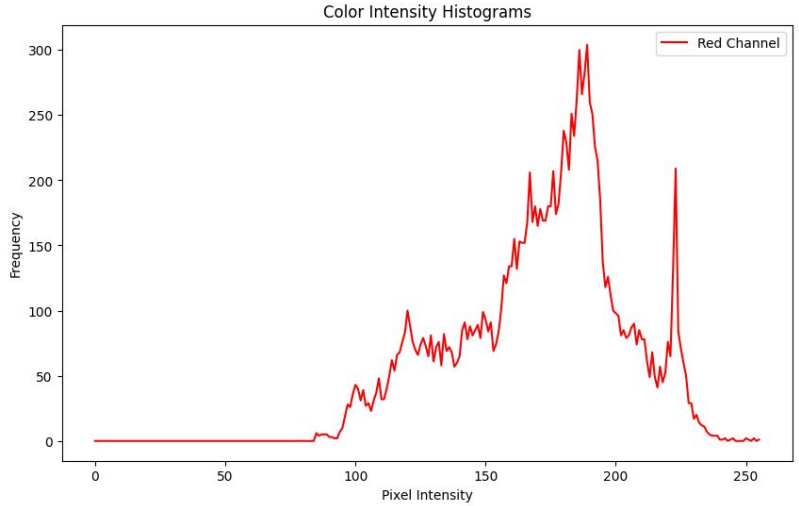
Contribution to the project during internship

	A	B	C	D	E	F	G	H	I	J	K	L
1	objid	ra	dec	u-type	u-peaks	g-type	g-peaks	r-type	r-peaks	i-type	i-peaks	
2	1.24E+18	244.3774	46.0917	DOUBLE	[(50, 50)(6	DOUBLE	[(50, 50)(6	DOUBLE	[(50, 50)(3	DOUBLE	[(50, 50)(36, 40)]	
3	1.24E+18	116.6311	42.89139	DOUBLE	[(51, 36)(5	DOUBLE	[(51, 37)(5	DOUBLE	[(51, 37)(5	DOUBLE	[(51, 37)(50, 50)]	
4	1.24E+18	117.5319	43.6445	DOUBLE	[(64, 69)(6	DOUBLE	[(65, 67)(6	DOUBLE	[(62, 67)(6	DOUBLE	[(65, 68)(66, 78)]	
5	1.24E+18	197.5538	65.47992	DOUBLE	[(51, 62)(5	DOUBLE	[(50, 50)(5	DOUBLE	[(50, 51)(5	DOUBLE	[(51, 51)(51, 62)]	
6	1.24E+18	217.7719	-0.91944	DOUBLE	[(9, 52)(16,	DOUBLE	[(8, 52)(50,	DOUBLE	[(8, 53)(50,	DOUBLE	[(9, 53)(50, 50)]	
7	1.24E+18	217.4418	-0.1524	DOUBLE	[(35, 55)(4	DOUBLE	[(35, 55)(4	DOUBLE	[(50, 50)(3	DOUBLE	[(52, 50)(36, 55)]	
8	1.24E+18	14.07227	-0.36902	DOUBLE	[(55, 44)(5	DOUBLE	[(56, 44)(5	DOUBLE	[(50, 50)(5	DOUBLE	[(50, 50)(55, 44)]	
9	1.24E+18	156.3485	64.9987	DOUBLE	[(50, 50)(3	DOUBLE	[(50, 50)(4	DOUBLE	[(50, 50)(5	DOUBLE	[(50, 50)(41, 42)]	
10	1.24E+18	179.0024	-2.72006	DOUBLE	[(68, 42)(5	DOUBLE	[(68, 43)(5	DOUBLE	[(69, 43)(5	DOUBLE	[(69, 42)(51, 50)]	
11	1.24E+18	179.3016	-2.68647	DOUBLE	[(50, 50)(2	DOUBLE	[(50, 50)(2	DOUBLE	[(50, 50)(2	DOUBLE	[(29, 45)(49, 50)]	
12	1.24E+18	144.2059	57.9921	DOUBLE	[(49, 50)(4	DOUBLE	[(50, 50)(4	DOUBLE	[(50, 50)(5	DOUBLE	[(50, 50)(51, 36)]	
13	1.24E+18	144.1331	59.40518	DOUBLE	[(17, 15)(1	DOUBLE	[(13, 1)(17,	DOUBLE	[(13, 1)(17,	DOUBLE	[(12, 1)(17, 15)]	
14	1.24E+18	146.5368	58.53046	DOUBLE	[(51, 50)(5	DOUBLE	[(50, 49)(5	DOUBLE	[(50, 50)(5	DOUBLE	[(50, 49)(57, 55)]	
15	1.24E+18	194.5769	51.13008	DOUBLE	[(50, 49)(5	DOUBLE	[(50, 50)(6	DOUBLE	[(50, 50)(6	DOUBLE	[(60, 60)(50, 50)]	
16	1.24E+18	133.4081	40.73002	DOUBLE	[(50, 50)(6	DOUBLE	[(50, 50)(6	DOUBLE	[(50, 50)(6	DOUBLE	[(50, 50)(67, 34)]	
17	1.24E+18	122.4675	35.19424	DOUBLE	[(51, 57)(5	DOUBLE	[(51, 57)(5	DOUBLE	[(50, 50)(5	DOUBLE	[(50, 50)(51, 58)]	
18	1.24E+18	122.1046	34.9047	DOUBLE	[(47, 44)(4	DOUBLE	[(47, 44)(5	DOUBLE	[(50, 50)(4	DOUBLE	[(50, 50)(47, 44)]	
19	1.24E+18	252.8739	40.29336	DOUBLE	[(51, 41)(5	DOUBLE	[(51, 41)(5	DOUBLE	[(52, 40)(5	DOUBLE	[(52, 41)(51, 50)]	
20	1.24E+18	154.4533	4.615416	DOUBLE	[(48, 73)(4	DOUBLE	[(48, 70)(4	DOUBLE	[(48, 70)(4	DOUBLE	[(48, 70)(48, 83)]	
21	1.24E+18	155.0912	4.888983	DOUBLE	[(58, 61)(5	DOUBLE	[(58, 61)(5	DOUBLE	[(50, 50)(5	DOUBLE	[(50, 50)(58, 60)]	
22	1.24E+18	221.9567	1.051977	DOUBLE	[(36, 42)(4	DOUBLE	[(33, 46)(4	DOUBLE	[(33, 46)(5	DOUBLE	[(33, 46)(49, 50)]	
23	1.24E+18	156.6218	20.23355	DOUBLE	[(26, 23)(4	DOUBLE	[(5, 7)(26,	DOUBLE	[(6, 8)(13,	DOUBLE	[(7, 7)(13, 3)]	
24	1.24E+18	130.7751	26.80619	DOUBLE	[(67, 36)(5	DOUBLE	[(66, 36)(5	DOUBLE	[(67, 36)(5	DOUBLE	[(66, 37)(50, 50)]	
25	1.24E+18	211.3107	9.275822	DOUBLE	[(45, 29)(4	DOUBLE	[(44, 29)(4	DOUBLE	[(50, 50)(4	DOUBLE	[(50, 49)(45, 29)]	
26	1.24E+18	241.3946	7.457648	DOUBLE	[(41, 55)(5	DOUBLE	[(51, 49)(4	DOUBLE	[(50, 51)(4	DOUBLE	[(50, 50)(41, 55)]	
27	1.24E+18	198.8336	33.00119	DOUBLE	[(45, 44)(5	DOUBLE	[(50, 50)(4	DOUBLE	[(50, 50)(4	DOUBLE	[(50, 49)(45, 43)]	

Snippet of the filtered objects that have “DOUBLE” peaks in all four bands

Contribution to the project during internship

- Removing the false positives from the sample
  - Developed the python code to detect the false positive images by using threshold filtering technique



1237651252046921791

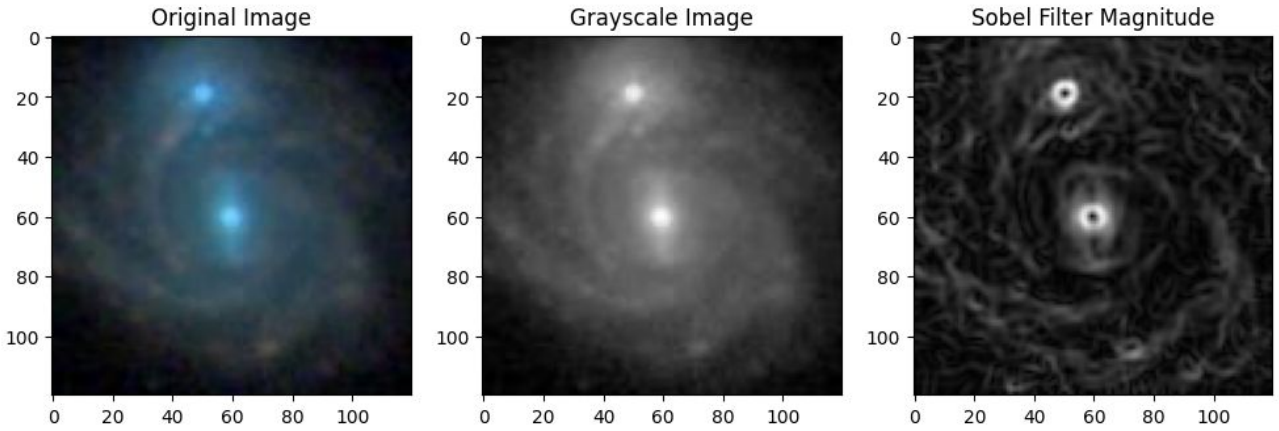
Example of a false positive in the sample. On left, is the Color intensity profile of the red channel which is used for threshold filtering. On right, is the image from the SDSS server of object with objID(1237651252046921791)

Contribution to the project during internship

- Proposed the Machine Learning Model
  - After extensive study proposed YOLOv8 [**Y**ou **O**nly **L**ook **O**nce] object detection Machine Learning Model.
  - YOLOv8 was tentatively selected as the model because of its computational efficiency, high accuracy, and good performance on small objects
- Categorized the Training, Testing, and Validating Dataset
  - Training Dataset:- 681 confirmed DGNs
  - Testing and Validation Dataset:- 45,380 probable DGNs candidates

Contribution to the project during internship

- Applying various Image Processing Techniques over training dataset for better accuracy
  - Programed the python code to try various Image Processing Techniques over images for better results
  - Sobel filter over grayscale image proved to be better for edge detection purpose



Scope of error

Scope of error

- Quality of Training Dataset may be compromised because of less number of confirmed DGNs to train the model upon
- One major error is that the model either detects some unnecessary objects in the background hence resulting in poor accuracy
- It does not detect anything at all because of too much noise in the image or low intensity of the sources

The Way Ahead

The Way Ahead

- Augmenting the images to increase the size of the training data set and hence increasing the model accuracy
- Exploring better and more advanced image processing techniques for improving the data set
- Thorough review of the YOLOv8 model shall be done to know if it is right model for this task.

Acknowledgement

Acknowledgement

I would like to express my sincere gratitude to my supervisor, Dr. Mousumi Das, for accepting my application and placing her trust in me. I am deeply thankful for the opportunity to work under her exceptional guidance for two months. Her mentorship not only paved the way for me to explore the vast field of astrophysics but also provided me with the invaluable chance to apply my knowledge of machine learning to her research project.

Bibliography

Bibliography

https://arxiv.org/pdf/2011.12177

https://www.geeksforgeeks.org/data-analysis-with-python/

https://www.geeksforgeeks.org/image-processing-in-python/

https://docs.ultralytics.com/

https://www.youtube.com/watch?v=vT1JzLTH4G4&list=PL3FW7Lu3i5JvHM8ljYj-zLfQRF3EO8sYv