

# **CMP3753 Literature Review and Progress Update**

Luke Roberts

University of Lincoln  
25722923@students.lincoln.ac.uk



# UNIVERSITY OF LINCOLN

# 1 Literature Review

As specified in the project proposal (L. Roberts 2023) the project aim is to compare the ability of different convolutional neural network (CNN) architectures to classify the morphology of galaxies. Images and classifications from the NASA/IPAC Extragalactic Database (NED) will be used to train and test the performance of each model. This literature review serves to inform the reader about the background of CNNs and personal considerations in choosing the models that will be tested in this project.

## 1.1 Early Convolutional Neural Networks

CNNs are a specific type of machine learning model that aims to classify images as well as other kinds of unstructured data. While much more commonplace in studies today, CNNs have existed since the late 1980s. (Y. LeCun et. al, 1989) developed a CNN called LeNet which successfully categorised handwritten digits with a 3.4% error rate. The study, undertaken in AT&T Bell Labs, pioneered the utilisation of ‘convolutional feature maps’ for feature abstraction in higher hidden network layers, which are still a core component of CNN architectures. Since CNN architectures derive from artificial neural networks, training LeNet is undertaken through forward propagation and back propagation.

LeCun and his colleagues continued their research throughout the next decade, incrementally improving on the design of LeNet. In the paper (Yann LeCun et. al, 1998), LeNet-5 architecture improved upon the original model by adding more alternating convolution and subsampling layers, and traditional neural network layers using full connections before the output layer. LeNet-5 also used a minimisation procedure for backpropagation called stochastic gradient descent, which minimises parameter vectors around the local minimum at greater speeds than more traditional gradient descent algorithms, improving model training speed greatly. While relatively simple, LeNet-5 has been used in many practical applications across many fields of study. Indeed, a 2022 conference paper (P. S. Radhamani et. al 2022) showed that LeNet-5 was capable of categorising galaxy morphology with 96% accuracy for two classes. However, the statistic found in the conclusion is misleading because the paper intended to categorise nine different classifications, and did so with varying degrees of success. Nevertheless, because of its implementation simplicity and historic relevance the LeNet-5 architecture will be used as a baseline CNN in the project for comparison to more recently designed CNN architectures.

## 1.2 Developments in Convolutional Neural Networks

Although LeNet-5 was a breakthrough for CNN architecture, neural networks had yet to be widely used in research and commercial settings. This was largely due to the limitations the hardware could provide at the time – only high end hardware could run the number of complex matrix multiplication operations needed for LeNet-5, limiting the scope and number of projects using the architecture. As hardware continued to improve through the 2000s and early 2010s, deep CNNs became more accessible, meaning scope of CNN-based projects became larger and more widespread. One pioneering architecture was AlexNet by (A. Krizhevsky et. al 2017), originally trained on the ImageNet dataset for the ImageNet Large Scale Visual Recognition Challenge 2012. Compared to earlier models such as LeNet-5, AlexNet was much more deep, containing five convolution layers and three full layers. This meant that the neural network had the potential to learn patterns of greater abstraction about the input dataset. However, because of the increase in model depth, new techniques were also employed to mitigate the impact of a CNN training issue called the vanishing gradient problem. (S. Basodi et. al 2020) describe the vanishing gradient problem as when ... during backpropagation, network weights are updated proportional to the gradient value (...) after each training iteration (epoch). ..., sometimes the gradient value is too small and gets gradually diminished during backpropagation to the initial layers.' In more understandable terms, as backpropagation trains layers that are closer to the input layer less. To facilitate continued training of AlexNet, a ReLU activation function was used instead of a sigmoid function. AlexNet continues to be used as an image categorisation model because of its lower computational needs and its high performance.

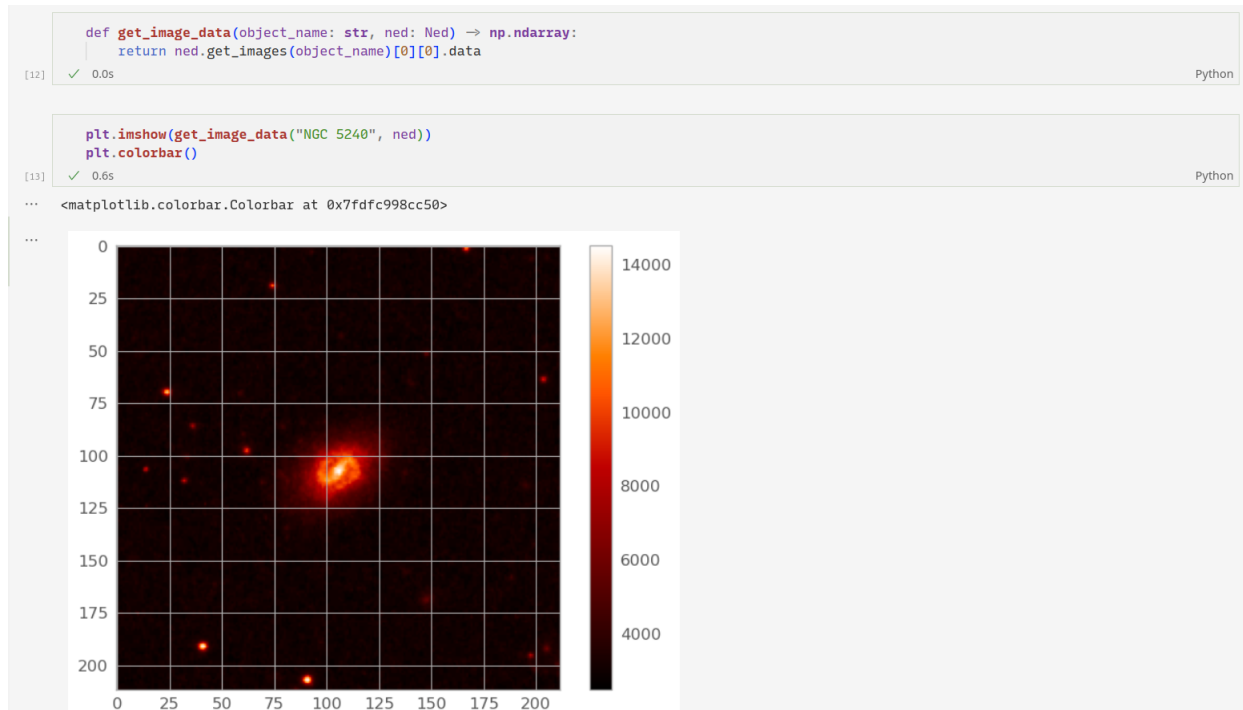
After the success of AlexNet in 2012, there was a renewed interest in deep learning models for use in computer vision and other domains. New breakthroughs in CNN architectures became much more frequent and the depth of new models continued to increase. A problem that was addressed by S. (Basodi et. al 2020) was that the gap of error between networks with 18 and 34 layers converged after a certain number of iterations known as the degradation problem. They discovered that by using residual connections between layers, the 34 layer model performed better throughout all training iterations. The model they developed became known as ResNet, which has been used in many practical applications for image categorisation. Both ResNet and AlexNet are modern CNN architectures which will be tested as part of the project; comparing the abilities of a modern but simpler model to a deeper residual network architecture.

A similar study involving morphology categorisation of galaxies from Galaxy Zoo 2 by (J. Dai, J. Tong. 2018) showed that implemented ResNet-50 and AlexNet models could achieve >90% accuracy. That being said the best performance was achieved by their custom designed model. Nevertheless, ResNet and AlexNet have been selected for this project alongside the LeNet-5 model to test how different architectures affect model performance.

## 2 Progress Report

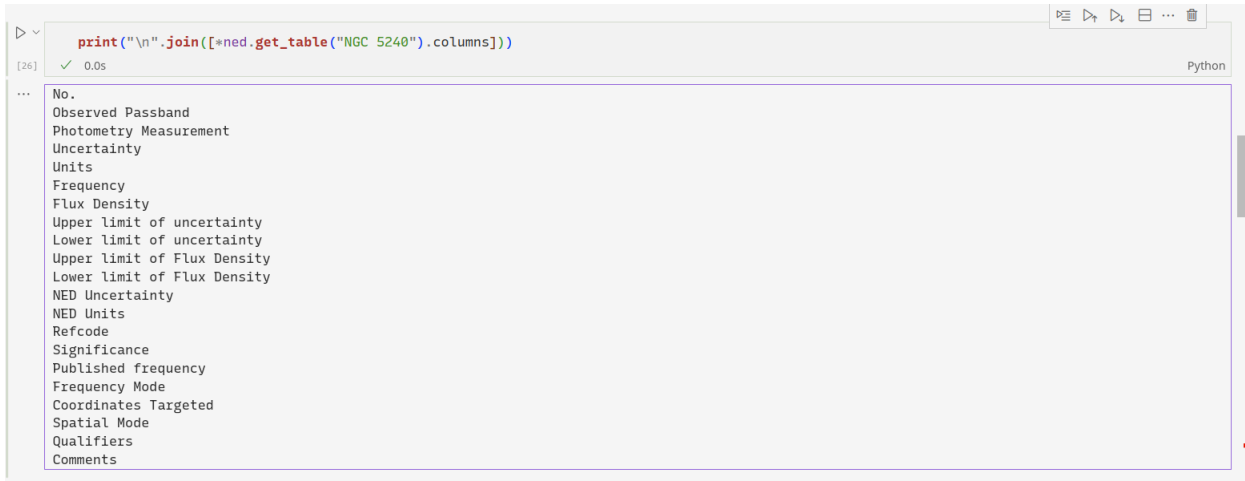
As planned in the objectives of the project proposal (L. Roberts 2023) the first step of the project was to collect a dataset of images and classifications using the python astronomy database querying tool AstroQuery. AstroQuery allows the user to interface with many online astronomy databases including the European Space Agency Hubble Space Telescope (ESA HST) Archive and the NASA/IPAC Extragalactic Database (NED). The NED archive was chosen because it compiled different records of galaxies from separate observatories, having 2,500,000 images (NED Current Holdings n.d.) as of its most recent release.

My initial AstroQuery experimentation was done on a Jupyter Notebook (Jupyter 2019) because of its ability to interface with scientific computing libraries. Firstly image downloading was tested using the `Ned.get_images` function. Using the python scientific diagram library Matplotlib (Matplotlib 2012), the images downloaded from the archive could be displayed. The specific file format that the NED database contains are Flexible Image Transport System (FITS) images.



**Fig. 1.** Output image from NED using astroquery

Once image acquisition was tested, the next step was to query for a table of galaxy names and each galaxy's associated classification using `Ned.get_table`. Unfortunately the NED API implemented for AstroQuery did not return a column for galaxy classification when tested.



```
print('\n'.join([*ned.get_table("NGC 5240").columns]))
```

```
[26]: ✓ 0.0s
```

```
... No.
Observed Passband
Photometry Measurement
Uncertainty
Units
Frequency
Flux Density
Upper limit of uncertainty
Lower limit of uncertainty
Upper limit of Flux Density
Lower limit of Flux Density
NED Uncertainty
NED Units
Refcode
Significance
Published frequency
Frequency Mode
Coordinates Targeted
Spatial Mode
Qualifiers
Comments
```

**Fig. 2.** Output showing no classification column

The API reference was consulted to determine a way to configure the NED database search, however, nothing that was tried gathered the classifications. A risk that was discussed in my project proposal was that 'an external database [was] unavailable' which is similar to the issue that was currently being faced; not having part of the database available. Since it was unlikely that the API would be updated, the contingency was to use a different database meaning that a table containing both object names and classifications had to be collected from a different source. It was preferable to keep using NED as there would be a set of both names and images so more research was directed to find a source using NED. Luckily, there was an old online search tool that allowed specific classifications and object names to be searched (NED Search by Classifications) however, it only outputted a HTML table on many separate pages. The issue was resolved by writing a web scraping script to get-request every page given certain search criteria. The webpage had to be inspected in the browser so that each web page input that was needed was included as an option. The text from the requested page was then parsed into a dataframe, joined to other queries and exported as 'NED\_list.csv'. To maximise efficiency, a thread pool executor was used to parallelise the process of searching for pages.

```

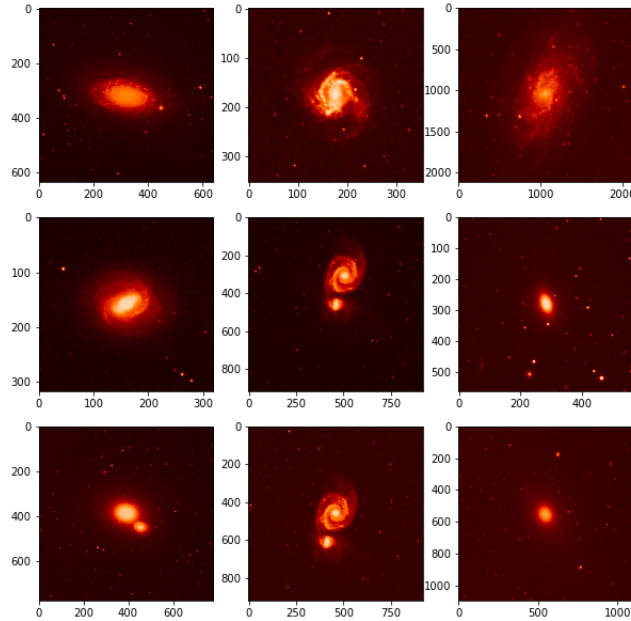
36  ZMASS J23164775+1534598,0.039000,S0/a,S0
37  ZMASS J23164775+1534598,0.039000,S0/a,S0/a
38  ZMASS J23573806-2943460,0.028330,E,E
39  ZMASX J00160121+1601331,0.027893,S0^-,S0
40  ZMASX J00162902-0036265,0.068192,S0,S0
41  ZMASX J00202768+0000160,0.046050,S0/a,S0
42  ZMASX J00202768+0000160,0.046050,S0/a,S0/a
43  ZMASX J00243207-4007300,0.069575,D/cD,E
44  ZMASX J00260179-0931577,0.053729,E0,E
45  ZMASX J00264283-4848061,0.071000,S0,S0
46  ZMASX J00372753-3907467,0.063427,E,E
47  ZMASX J00403677-3652360,0.034564,E,E
48  ZMASX J00431088-0940543,0.054956,E,S0/a
49  ZMASX J00431088-0940543,0.054956,E,E
50  ZMASX J00431088-0940543,0.054956,E,S0
51  ZMASX J00450258-0948288,0.019401,S0,S0
52  ZMASX J00524743-1012003,0.021870,S0/a,S0
53  ZMASX J00524743-1012003,0.021870,S0/a,S0/a
54  ZMASX J00550553-0114013,0.044364,Sb(f),S0
55  ZMASX J00560405-0955209,0.056790,S0,S0
56  ZMASX J00561436-0108397,0.044955,S0,S0
57  ZMASX J00562351-0059129,0.044030,E/S0,E
58  ZMASX J00563045-0132029,0.048167,S02,S0
59  ZMASX J00563838-0107339,0.045283,S0,S0
60  ZMASX J00565706-0123196,0.043497,S0,S0
61  ZMASX J00591457-2848305,0.035700,Sb,E
62  ZMASX J01010118-1018085,0.036003,S0/a,S0
63  ZMASX J01010118-1018085,0.036003,S0/a,S0/a
64  ZMASX J01043012+1430140,0.040923,E0,E
65  ZMASX J01072322+0040191,0.066144,S0^-,S0
66  ZMASX J01083363-1537167,0.099365,S0,E
67  ZMASX J01084837-1528218,0.055251,S0,S0

```

**Fig. 3.** Output table showing object and classification

After creating the table of galaxies and morphologies a tool was then developed to gather images from the NED archive. This tool used `Ned.get_image_list` to find if there were valid files available and took the name of the highest quality image file (always indexed at one from testing). `FileContainer.get_fits` was used to download the files individually and then `HDUList.writeto` wrote the data to a file. I then wrote another script that displayed each image for illustrative purposes.

The next project steps will be preprocessing images of the dataset and building the machine learning models for testing. In the project objectives, the next stage after dataset gathering was to analyse the images and dataset to determine how best to perform preprocessing. One issue with many of the images is that most of the image is empty space, which a machine learning model will find redundant when trying to categorise the image. Similar to the preprocessing performed in figure 6 of (J. Dai, J. Tong. 2018), each image will be cropped to a specific scale. Another issue present in the image dataset is that images range greatly in quality. To be able to input each image for training and testing, each image will be downsampled to a specific size and images with quality below a certain threshold will be removed, as they are unlikely to be useful in training. In the database, there are many galaxy objects that appear in two or more astronomical catalogues (MESSIER, NGC, ESO, etc.), so to prevent any biases that could occur only one catalogue will be used to train the models that will be implemented.



**Fig. 4.** Images downloaded from NED archives

Overall progress has been slower due to other university work, commitments and the setbacks discussed in this section. However, more focus can now be drawn to CNN model creation and preprocessing for the project, now that data gathering is completed.

## References

1. L. Roberts (2023) "CMP3753M Project Proposal". University of Lincoln, School of Computer Science. Unpublished essay.
2. Y. Lecun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel (1989) "Backpropagation applied to handwritten zip code recognition". IEEE, Available at: <https://ieeexplore.ieee.org/document/6795724>
3. Y. Lecun, L. Bottou, Y. Bengio, P. Haffner (1998) "Gradient-based learning applied to document recognition". IEEE. Available at: <https://ieeexplore.ieee.org/document/726791>
4. P. Radhamani, M. Sharif and W. Elmedany (2022) "An Effective Galaxy Classification Using Fractal Analysis and Neural Network" International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT). Available at: <https://ieeexplore.ieee.org/document/9990776>

5. A. Krizhevsky, I. Sutskever, G. Hinton (2017) “ImageNet Classification with Deep Convolutional Neural Networks”. ACM Digital Library. Available at: <https://dl.acm.org/doi/10.1145/3065386#sec-ref>
6. S. Basodi, C. Ji, H. Zhang and Y. Pan, (2020) “Gradient amplification: An efficient way to train deep neural networks,” in Big Data Mining and Analytics. Available at: <https://ieeexplore.ieee.org/document/9142152>
7. K. He, X. Zhang, S. Ren and J. Sun, (2016) “Deep Residual Learning for Image Recognition”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Available at: <https://ieeexplore.ieee.org/document/7780459>
8. J. Dai, J. Tong. (2018) “Galaxy Morphology Classification with Deep Convolutional Neural Networks”, Tables 6-7, arXiv (preprint). Available at: <https://arxiv.org/pdf/1807.10406.pdf>
9. NASA/IPAC Extragalactic Database. (n.d.) “Database holdings for release 33.3.1”. [online] Available at: <http://ned.ipac.caltech.edu/CurrentHoldings>
10. Jupyter (2019). Project Jupyter. [online] Jupyter.org. Available at: <https://jupyter.org/>
11. NASA/IPAC Extragalactic Database. (n.d.) “Search by Objects” <https://ned.ipac.caltech.edu/uri/NED::Classifications/#M>
12. Matplotlib (2012). Matplotlib: Python plotting — Matplotlib 3.1.1 documentation. [online] Matplotlib.org. Available at: <https://matplotlib.org/>