# Detection of Exoplanets System in Kepler Light Curves using Deep learning

MSc Research Project
MSc in Data Analytics


Nandhavarman Jeevarathinam
Student ID: x18186459

School of Computing
National College of Ireland


Supervisor: Dr. Paul Stynes

**National College of Ireland**
**Project Submission Sheet**
**School of Computing**

| | |
|---|---|
| **Student Name:** | Nandhavarman Jeevarathinam |
| **Student ID:** | x18186459 |
| **Programme:** | MSc in Data Analytics |
| **Year:** | 2020 |
| **Module:** | MSc Research Project |
| **Supervisor:** | Dr. Paul Stynes |
| **Submission Due Date:** | 30/09/2020 |
| **Project Title:** | Detection of Exoplanets System in Kepler Light Curves using Deep learning |
| **Word Count:** | 4042 |
| **Page Count:** | 14 |

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

**ALL** internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

| | |
|---|---|
| **Signature:** | |
| **Date:** | 28th September 2020 |

## PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies). | ☐ |
| **Attach a Moodle submission receipt of the online project submission**, to each project (including multiple copies). | ☐ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | ☐ |

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# Detection of Exoplanets System in Kepler Light Curves using Deep learning

Nandhavarman Jeevarathinam
x18186459

**Abstract**

*With the development made in Optical and Imaging technology, many astronomical observations were made recently. The brightness of a star over a period of a time are called light curves which are scanty, sparse and heteroscedastic. Using these huge time series data, classification task needs to be performed to label as planet candidates and false positive. In this research work, the time series extracted from the time series are transformed using recurrence plot for better pattern recognition using VGG16 Convolution architecture. Introduction of image augmentation with recurrence plot for the light curve data produces good result compared to many other researches and technologies. Main limitation in this domain is the size of the data to be processed. Using this methodology, comparatively less amount of data is used to train a model which can recognize transit planet patterns in a light curve time series data.*

***Keywords***— Exoplanet detection, Recurrence plot, Time series classification

## 1 Introduction

Most of the exo-planetary systems were detected using indirect methods like transit photometry and radial-velocity method. There are trillions of stars in our galaxy. Each one has atleast one planet with solid surface around them. In the habitable zone, around 30 percent of the medium sized stars has habitable planets. There are trillions of earth like habitable planets waiting to be discovered in our milky way galaxy. In this research work, transit data (light curves) is taken for the detection of exoplanets. Using machine learning techniques, proper transit patterns can be recognised by the machines to detect the transit of exoplanet around an alien star.

In astronomical community, Data is used as a Flexible Image Transport System(FITS). To plot the time series data from the FITS file, this research work used Astropy.io package is used in python. In research work [2], to classify light curves, they used Adaptive Neuro-fuzzy system. In Artificial intelligence area, lot of algorithms uses fuzzy logic concept. Neural network has more learning capacity than fuzzy systems. Using fuzzy system, they achieved an accuracy of 81%. According to the research conducted by authors of research paper [3], deep neural network is used for light curve classification. First, they flattened the data into a single dimensional vector which is fed into a artificial neural network to process. This same kind of approach is used in our research work using convolutional neural network.

Comparing results from the paper [2] and [3], deep learning algorithms perform better than parametric algorithms in classification. So, VGG16 architecture which is a deep learning model is chosen for the classification task. This research work is implemented to focus on the below research question.

1. *"The aim of this research work is to investigate to what extent machine learning can identify the perfect planet transit pattern based on their light curves?"* When a planet crosses between a distant star and observer from earth, it forms a unique shape of fall in the time series data of the light curve. This is called as Planet candidate. The proposed thesis work is excepted to achieve the above mentioned research objective.

In section 2, various literature and previous relevant work is discussed. The research methodology which is used to implement the insights derived from section 2 is described in section 3 of the paper. Section 4 conveys about the design architecture created for the implementation. In section 5, the technologies and tools required to implement the model are described in detail. The results obtained by conducting different experiments were discussed in section 6 of the paper. Finally, the motive of the paper is concluded and scope for future researchers is mentioned in the section 8 of this paper.

# 2 Related Work

In this section, various relevant research and literature work are discussed and critically analysed. For the classification of light curves, researchers used parametric modelling such as hypothesis testing prior to the development of machine learning techniques. A clear research and study about parametric models till advanced neural network architecture needs to be conducted for deeper understanding of this research methodology. Plethora of relevant research paper are clubbed under four parts.

1. Papers using Parametric Modelling.
2. Methods to get light curve data from astronomical data.
3. Light curve pre-processing.
4. Various classification techniques used.

## 2.1 Papers using Parametric Modelling

According to research work [4], a model is presented for the detection of exoplanets. The time series data with same mean and variance were arranged as an autoregressive function. A hike in the parameters like variance and mean, is triggered by a planet transition in front of a alien star. Neyman-Pearson detector is used if the change parameter is known. Otherwise, generalized likelihood ratio detector is used. Lot of photon noise mix up with the signals received. Two hypothesis were decided in the AC detection problem. No hike or jump in the mean or variance would become the null hypothesis and sudden jump would the alternate hypothesis. The test statistics called Neyman-Pearson detector was computed and if it was lower than the critical value calculated, null hypothesis would not be accepted. Failing to accept the null hypothesis would mean there was a hike in the statistics which was caused due the transit of an alien planet.

In the research paper by I.Nikiforov and M.Basseville, they improved the detection algorithm. But the algorithm was limited to off-line jump exposure algorithm [5]. Similarly in [6], off-line jump is designed to train the model. The model proposed in this paper [6], can control the case such as hike in the mean and the power of Autoregressive(AR) process.

## 2.2 Methods to get light curve data from astronomical data

In this thesis work, critical step is to extract the time series data from the astronomical data. Some prominent packages and methods are listed as follows. Hannu Parviainen in the research work [7], using PYTRANSIT package presented a effective way to access light curves. The aim of the package PYTRANSIT is to reduce the complexity of the study of time series data in the light curves. In research paper [1] and [3], the library Astropy is used to get the time series data from the FITS file. Astropy is stable and has many improvements over PYTRANSIT package. With getdata() and getheader() commands the header and data are easily accessed respectively. As compared to other packages, astropy is faster and easy to use.

## 2.3 Light curves pre-processing

The time series or the light curves needs to be processed properly to achieve good results while applying a machine learning model. This part of the previous work studies the some of the pre processing techniques used in this research work.

A median filter was enforced to the time series which was extracted in the research work [1]. It makes the time series more clear and normalize it to rise the computational speed. Missing points in data were removed to design the model without any errors. Time series is represented as a collective or kernal features. A huge part of false positive cases were removed to reduce the data imbalance issue in the paper [2]. Similarly like research work [1], the time series data was normalised first. To reduce the dimensions used, they used Dynamic time warp, statistical measures and PCA were used.

In the research project [10], Large Synoptic Survey Telescope(LSST) data was used. They used 7894 light curves data. SNCosmo was the package used for pre-processing the time series data in the research work [10]. SNCosmo is used to extract properties from time series data from the raw data. They normalized the data as KNN was used where Euclidean distance will be reactive to huge values. The research paper [11] used DMDT mapping method. For every pair of points in the time series data, the difference in time(dt) and magnitude(dm) was calculated. Skipping data points were also used in some of the research work like [12]. As the original data was very dense, every tenth point were taken into account to make the data more sparse. Using the above approach, the algorithm's learning speed can be increased. Outliers were also removed using normal standard deviation method. To remove the missing data problem, filling zeros and linear interpolation was used in the papers [13] and [14].

According to the authors of research paper [19], they proposed a new MA detection that is done using recurrence image processing. In this paper, they have converted a PPG data into a 2D image by recurrence plot algorithm. By using recurrence plot algorithm, they achieved an accuracy of 97.8 percent. Similarly in the work [20], the authors have used recurrence plot for time series classification. The proposed method in [20] outperforms Dynamic time warp and Euclidean distance.

## 2.4 Various classification techniques used

This part of the paper reviews and critically analyses different type of machine learning algorithms used in the previous work. According the research paper [10], authors used different parametric models like KNN,SVM, Naive bayes and Decision tree for the classification task. Since the training data were imbalanced, the above mentioned models produced very less results. Models produced good results after re-sampling.

Many research work uses deep learning for the classification of light curves. In research paper [11], authors used convolutional neural network (CNN) as it works well with image data. Convolutional network was used for classification of galaxies in research paper [15]. VGG16 architecture was used for detecting bright explosion of stars called supernovae in [16]. Light

Table 1: Models used in research work [10]

| Classifier | Technique name | Accuracy | SD |
|---|---|---|---|
| KNN | k = 1 | 83.3 | 1.9 |
| KNN | k = 3 | 78.3 | 2.3 |
| KNN | k = 5 | 77.5 | 3.1 |
| Decision Tree | Random Forest | 78.2 | 1.9 |
| SVM | | 71.6 | 1.1 |
| Naive bayes | | 47.2 | 3.8 |

curves were converted to dmdt images and given as Convolution neural network which got an accuracy of 83 percent. In the research paper [13], authors had taken KNN, Logistic regression, Random forest and SVM for the classification task. They analysed these models using different pre-processing techniques. The results they achieved are displayed in the table below.

Table 2: Accuracy achieved in research work [13]

| Test Set | KNN | Logistics regression | SVM | Random Forest |
|---|---|---|---|---|
| Fourier + PCA | 0.679 | 0.493 | 0.486 | 0.713 |
| OwnFATS | 0.666 | 0.583 | 0.575 | 0.658 |
| OwnFATS + Stellar metadata | 0.844 | 0.864 | 0.876 | 0.883 |

The authors of the research paper [12], used LSTM RNN and results were not good. So similarly like [13], they used parametric models with SMOTE pre-processing technique. The results of this research work were discussed in the evaluation section of this paper.

## 2.5 Insights from the previous works

From the above research works and literature, many methodologies and ideas are critically analysed and some are chosen for our research methodology. Out of the many ready made packages available to access data within FITS file, Astropy package is chosen for this research project. Another insight got from the previous research papers is that good results cannot be achieved without properly pre-processing the light curves. Plethora of pre-processing techniques were studied and recurrence plot with image augmentation shows promising improvements in the previously analysed work.

# 3 Methodology

Extracting time series from the downloaded FITS format file has various stages before giving it as a input and arriving at a conclusion from the raw data. To track the different processes, a standard procedures or methodology needs to be followed. The suitable methodology recommended for this astronomical data would be Knowledge Discovery in Databases (KDD).

This methodology gives importance to the technical sections of the research. The proposed methodology (KDD) works well with neural networks as mentioned in the research paper [17].

### 3.0.1 Exoplanet detection research methodology

Exoplanet detection methodology is used to derive knowledge and information from the large collection of astronomical data FITS data. This below used methodology is better for pattern recognition in time series data. There are six important phases or steps followed in this methodology. It will classified under these two categories:

- **Planet Candidate:** Confirmed exoplanet.
- **Astrophysical False Positive:** Not a planet.

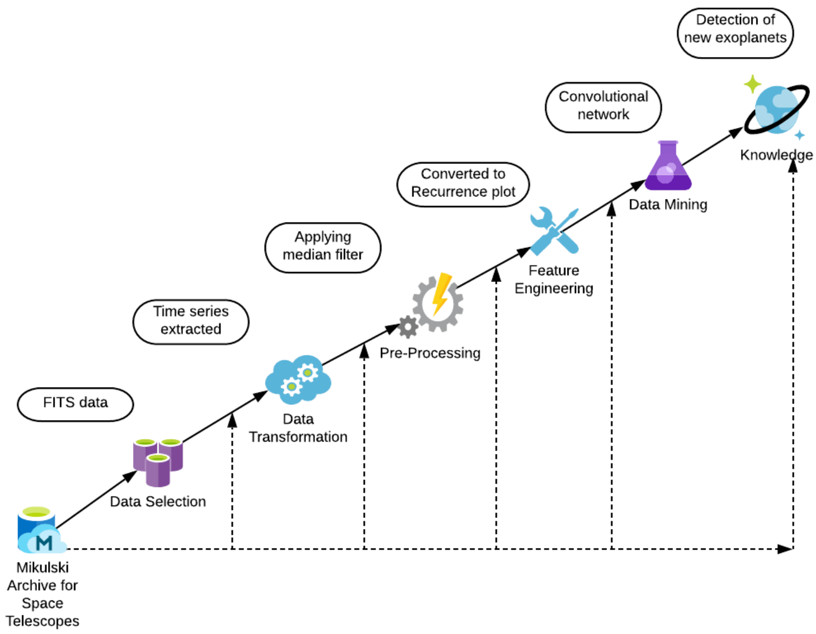



Figure 1: Exoplanet detection research methodology

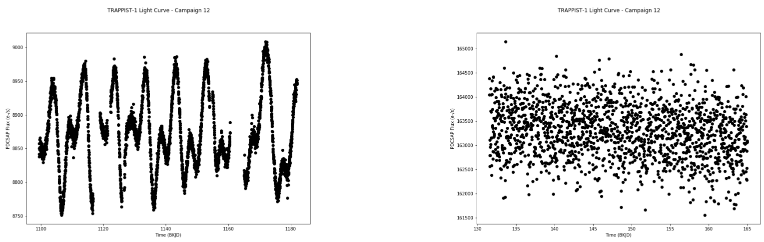
### 3.0.2 Data Collection

Data for this research work are collected from Mikulski Archive for Space Telescope (MAST). Using the Kepler Data search and Retrieval feature in the MAST portal[1], specific kepler objects or specific targets can be downloaded to train the model without any data imbalance. Using this portal, the links of each light curves can be downloaded as a text file.

### 3.0.3 Data Transformation

This part involves the process of converting the FITS data into time series data for pattern recognition. Then using Astropy.io package in python, each light curve data can be accessed one by one where their 'TIME', 'SAP FLUX' and 'PDCSAP FLUX' can be stored in a variable. Before plotting the time series, the data are stored in a data frame where all the missing values are dropped to make the time series continuous without any missing data points. Using matplotlib.pyplot package in python, the time series is plotted.

[1] `http://archive.stsci.edu/kepler/data_search/search.php`



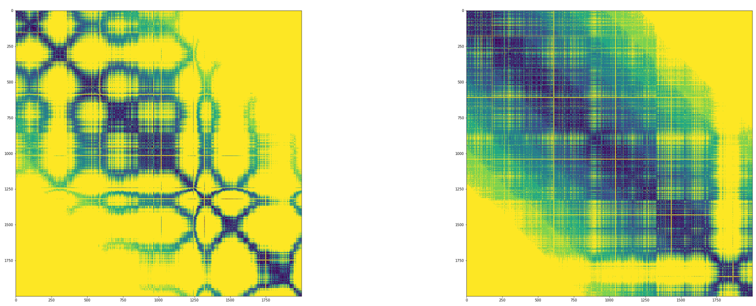

(a) Positive case (b) Negative case

Figure 2: Light curves

The above time series images is given as a input to the convolutional network in Experiment 2 which is explained later in the research paper.

### 3.0.4 Data Pre-processing

Before using these raw time series plot as inputs for convolutional network, certain pre-processing steps needs to be carried for better pattern detection of transit planets. After the time series data is extracted from the FITS file, to smoothen the time series data, a median filter is implemented. Unwanted time series data is also removed from the time series. Main advantages of using a median filter would be (1) Median filter does not remove the sharp edges. (2) Spiky noises are cleared [18]. Then, Recurrence plot is used to make the convolutional network recognize patterns easily. Recurrence plot is an progressive technique of non linear data analysis. Recurrence plot is a graph of a square matrix which is created using the time series data, in which for those specific time the matrix elements corresponds to dynamic states. Recurrence plot is nothing but a visualization tool to compute the time consistency. A recurrence of a case at time i at a diverse time j is pointed within a 2D squared matrix with 0s and 1s where both poles describe time.



(a) Positive case (b) Negative case

Figure 3: Recurrence plot from light curves

### 3.0.5 Data Mining

For the classification task to be performed, convolutional neural network is chosen in this research work. CNN is very efficient for the image data classification. Novelty of this research work would be using recurrence plot with CNN. Different architecture of CNN are trailed and applied in different experiments to find the right architecture to classify the time series. The experiments are explained in detail in the implementation section.

### 3.0.6 Evaluation metrics

For any classification task or prediction model, various evaluation metrics are available to measure the goodness of the model. Using the confusion matrix created from the model, many evaluations metrics are computed for the model. The metrics used in this research project are Accuracy, valuation accuracy, ROC curve, precision and recall plot.

# 4 Design Specification

For this work, three layers of architecture is designed i.e. Database layer, Application layer and Presentation layer. This architecture will imply the whole structure of this project work including mandatory requirements to be used. Figure 4 is made using [2].

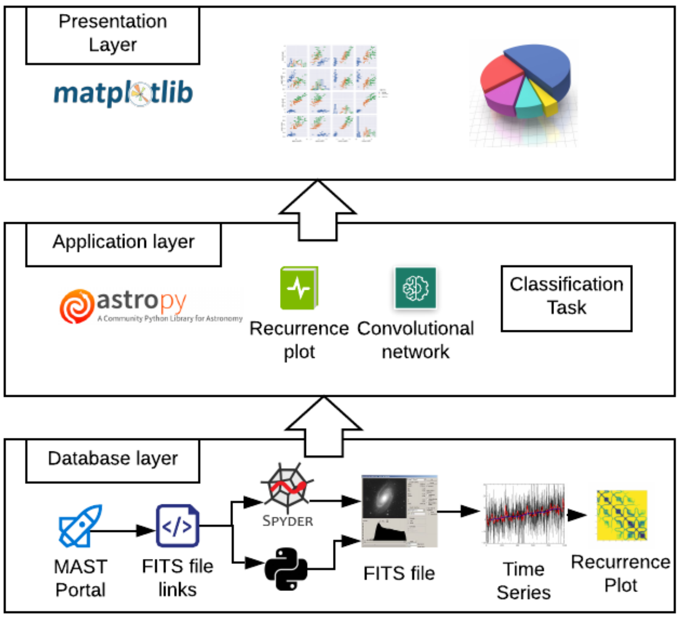



Figure 4: Design Architecture

In the database layer, data is collected from the data source and processed to be given as a input for the convolutional network for classification task. In here, Links of the FITS file are downloaded from the MAST portal. Then each link is accessed and FITS file is downloaded in a automated way using python 3.7. Then using the astropy.io package, the time series data is derived from the FITS file and plotted as time series using matplotlib package. With help of a custom made function, the time series plots are converted to recurrence plot. These recurrence plot of each time series data is stored in the local machine for the classification task. The recurrence plot images which was stored in the previous layer, are given as input to the convolutional network. The configurations of the convolutional network are explained in detail the implementation section of this research paper. After the model have been trained, plethora of evaluation metrics are used to gauge how well the model predicted. Presentation

[2] `https://www.lucidchart.com`

layer contains dashboards and exploratory visualizations of the evaluation results of the deployed VGG16 convolutional architecture.

# 5 Implementation

This part of the research paper demonstrates the hardware and software essentials such as algorithms, packages and IDEs used for implementing the model. Understanding the architecture used helps us in improving the model further.

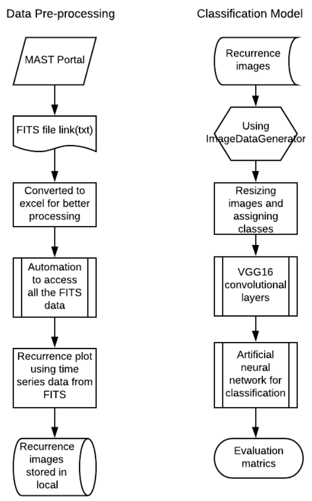



Figure 5: Implementation Workflow

## 5.1 Astropy package

By studying and analysing different research work, astropy.io package is selected for accessing the FITS data. With 'open' command, the data stored in the FITS file can be accessed to plot the time series data which will be the input for the convolutional network.

## 5.2 Recurrence plot

For creating a recurrence plot, sklearn and numpy package from python are needed. Recurrence plots are created from the time series data extracted from the FITS file. It is used in this research work as it provides better pattern recognition for the convolutional neural network. A custom function is made for creating the recurrence plot using the time series data. The recurrence plot custom function is added in the code artifact. The pseudo code for recurrence plot is shown below.

```
def recurrence_plot(time_series_data, eps, steps)
    if eps is not assigned then eps = 5
    if steps is not assigned then steps = 5
    distance = pairwise distance(time_series_data)
    distance normalized by eps value
    return distance
```

Figure 6: Recurrence plot algorithm

## 5.3 Image Augmentation

The main motive of using image augmentation is to modify the original image by resizing, rotating images, zooming etc to design more new images. With this concept, the classification model will have more images or features to capture than before which will increase the exposure to unseen data. Addition of image augmentation is done with TensorFlow image data generator. When different type of augmentation are done, the original image data will be unaffected by this functionality. The above configuration are done for image augmentation in this research

```
train_datagen = ImageDataGenerator(
   rotation_range=40,
   width_shift_range=0.2,
   height_shift_range=0.2,
   rescale=1/255,
   shear_range=0.2,
   zoom_range=0.2,
   horizontal_flip=True,
   fill_mode='nearest',
   validation_split=0.2
   )
```

Figure 7: Image data generator

paper. Using flow from directory function from Image data generator package, the train and test images are resized and assigned to the corresponding category.

## 5.4 Implementation of Exoplanet detection model

For implementing the classification model, VGG16 convolutional architecture is considered. **Why Convolutional architecture?** If the data is spatially aligned and scanty such as audio or image, Convolutional neural network outperforms all other neural network in many cases. We use keras package for implementing the architecture. Keras package has pre-trained five convolutional architecture for image classification.

1. Xception
2. ResNet50
3. Inception V3
4. VGG16
5. VGG19

Out these architecture mentioned above, we have chosen VGG16 as it outperforms many architecture in time series plot classification.

### 5.4.1 VGG16 Architecture

VGG16 is one type of convolutional neural network architecture. VGG16 is best known for efficient vision model architecture. Most of the architecture focuses on the hyper parameters used. But VGG16 targets on the convolutional layer of 3x3 size with same padding and with a max pool layers of 2x2 size. Figure 8 shows the architecture of VGG16. The input to first convolutional layer is a image of 512 x 512 size. Then it is followed by series of convolutional and max pool layers continuously through the full architecture. At the end we can modify the number of input nodes and hidden layer according to our data. The number 16 in VGG16 refers to the 16 weighted layer. After the VGG16 architecture layers, a flatten layer is used to

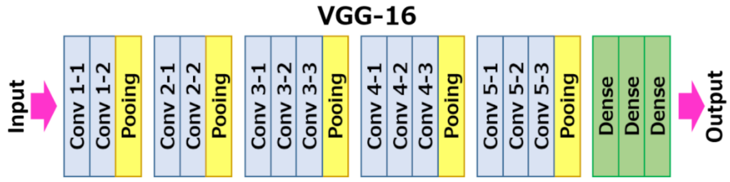



Figure 8: VGG-16 Architecture

```
Layer (type)                 Output Shape              Param #
=================================================================
vgg16 (Model)                (None, 16, 16, 512)       14714688
_________________________________________________________________
flatten_2 (Flatten)          (None, 131072)            0
_________________________________________________________________
Dense_Intermediate (Dense)   (None, 256)               33554688
_________________________________________________________________
Dense_Intermediate1 (Dense)  (None, 128)               32896
_________________________________________________________________
Dense_Intermediate2 (Dense)  (None, 64)                8256
_________________________________________________________________
Dropout_Regularization (Drop (None, 64)                0
_________________________________________________________________
Output (Dense)               (None, 1)                 65
=================================================================
Total params: 48,310,593
Trainable params: 33,595,905
Non-trainable params: 14,714,688
_________________________________________________________________
```

Figure 9: Model Summary

straighten the output from the VGG16 architecture. Then three hidden layers are used with 256, 128 and 64 nodes respectively with ReLU activation function. Finally one output node with sigmoid activation function is used for classification.

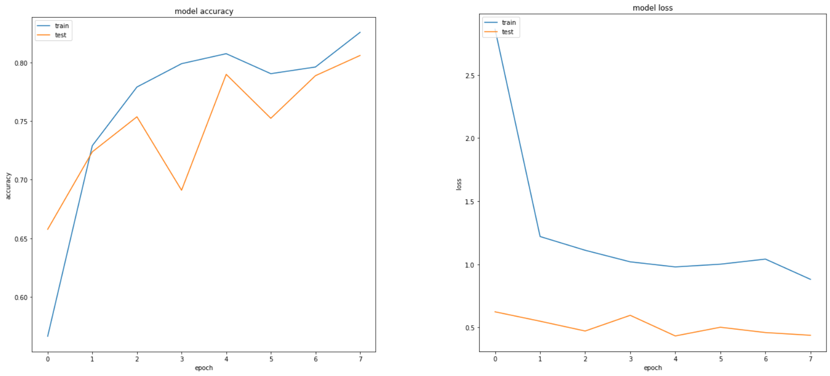
# 6 Evaluation

In this research project, a reference research paper is considered for the data and the methodology used. The Reference research paper is based on the research work by Trisha and Kevin in 2017. In this research, training set is extracted from Mikulski Archive for Space Telescope (MAST) which is provided by NASA Exoplanet Archive. Since the authors from this clearly stated that LSTM RNN gave poor results due to sparseness and noisiness of the light curve data, different parametric classification models like KNN, D-Tree and Random Forest was applied [12]. They achieved the highest accuracy of 85 percent using Decision tree. The main motive is to improve this model.

## 6.1 Experiment 1: Time series plot with VGG16 architecture.

In Experiment 1, same data from research paper [12] is taken. From the light curves, time and flux is plotted as a time series. Instead of plotting the whole data of a star like the reference research paper, plots are separated by quarters in this experiment. The created time series graphs are directly fed into a VGG16 convolutional architecture for the classification task. It gives a relatively low accuracy of 82 percent and test accuracy of 80 percent.

From the model accuracy graph 10(a), at 7 epochs, 82 percent is achieved. Test accuracy is lower than the train accuracy like an ideal model. According to the model loss graph 10(b), train loss is above 1 which denotes that the model's inconsistency. Compared to the research work [12], it is still lower as the authors have achieved 85 percent using Random forest.

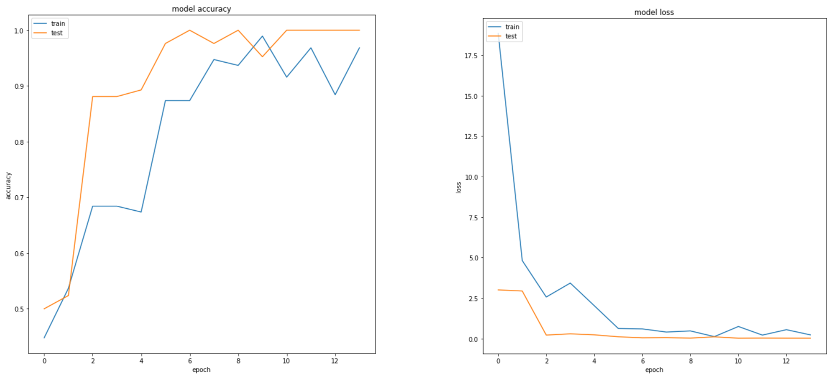



(a) Model accuracy (b) Model loss

Figure 10: Experiment 1 evaluation

## 6.2 Experiment 2: VGG16 architecture with recurrence plot

Repeating experiment 1 but changing the time series plot to recurrence plot for Convolutional network to recognize the pattern better according to the classification of time series using CNN research work published in 2013 [19]. Recurrence plot improves the performance better which gives an accuracy of 94.7 percent with a test accuracy of 96 percent.



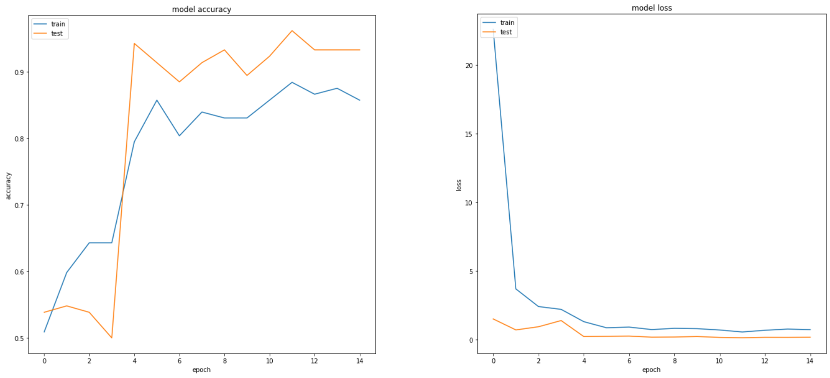

(a) Model accuracy (b) Model loss

Figure 11: Experiment 2 evaluation

From the model accuracy 11(a) graph, it is evident that the model is performing better than the VGG16 model from experiment 1 as recurrence plot patterns are recognised better than the time series plots. But the test accuracy curve is more than the train accuracy curve which denotes that there is a problem of over-fitting in the model. In the model's loss graph 11(b), loss is very much less than the experiment 1 model which denotes the consistency of the model in experiment is better.

## 6.3 Experiment 3: Checking the consistency of the model

In Experiment 3, the same approach is used with different dataset from the MAST portal. Here, we achieve a train accuracy of 87.5 percent still better than the base research work's decision tree model but lesser than the accuracy from the experiment 2. Similarly like experiment 2, the test accuracy is more than the train accuracy denoting there is a problem of over-fitting. The model's accuracy and loss graph is shown below.




(a) Model accuracy (b) Model loss

Figure 12: Experiment 3 evaluation

# 7 Discussion

The direct feeding of raw time series plots was not producing good accuracy which led us to investigate different pre-processing techniques like application of median filter, image augmentation technique and recurrence plot. While Image augmentation exposed the VGG16 convolutional network to variety of unseen data presented to it, recurrence plot provided very promising results for classification. The accuracy of the model still did not reach above 95 percent. Sparseness and Noisiness of the light curves affect the performance of the model to a greater extent.

Moreover most of the research work related to exoplanet detection would use a huge amount of data to achieve good accuracy. But, using this research methodology, with comparatively very less amount of sample data, the model can be trained effectively and achieve good accuracy. The usage of less data denotes less storage space, less memory cache and most importantly very less training time. This is one of the important contributing factor to the exoplanet detection field.

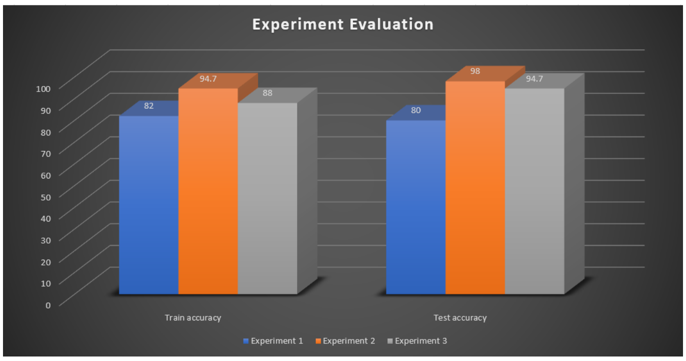



Figure 13: Experiment comparison

From the figure 13, it is evident that experiment 2 and 3 which uses the recurrence plot clearly outperforms the experiment 1 model both in model accuracy and model loss.

# 8 Conclusion and Future Work

With the prominent growth in the size of astronomical data, new approaches, technologies and methods needs to be developed to decrease the processing time, development of more accurate models and deriving new insights from the data itself. The novelty of this research project was using recurrence plot with VGG16 architecture for Light curve classification. The research question stated in this introduction section was also answered as a model was created to recognize pattern in the light curves with 94 percent accuracy. Upon experimenting, it is discovered that transit planet patterns were not recognized properly by giving time series data directly as a input to the VGG16 architecture. It is due to the noisiness and sparseness of the time series data used. To recognise the pattern better among the noise, we considered recurrence plot and image augmentation. It increased the accuracy of the model upto 94 percent. Only limitation faced in this research work would be the overfitting problem. For Future work, researchers can consider removing the overfitting problem faced in this research work by experimenting with different kind of light curve data. This work will motivate researchers and astronomers to find many exoplanets with high accuracy, which can be the second earth to our humanity in far future.

# Acknowledgement

Firstly, my sincere gratitude to Dr. Paul Stynes and Dr.Pramod Pathak for the their expert opinion and guidance throughout the tenure of this research work. Thanks to my parents and friends for the unconditional love and support.

# References

[1] Woodward, D., Stevens, E., & Linstead, E. (2019). Generating Transit Light Curves with Variational Autoencoders. Proceedings - 2019 IEEE International Conference on Space Mission Challenges for Information Technology, SMC-IT 2019, 24–32. https://doi.org/10.1109/SMC-IT.2019.00008

[2] R.M.Asif Aminetal et al,"Detection of Exoplanet Systems In Kepler Light Curves Using Adaptive Neuro-Fuzzy System," 2018 International Conference on Intelligent Systems (IS),Funchal-Madeira,Portugal,2018,pp. 66-72.

[3] Shallue, C. J., & Vanderburg, A. (2018). Identifying Exoplanets with Deep Learning: A Five-planet Resonant Chain around Kepler-80 and an Eighth Planet around Kepler-90. The Astronomical Journal, 155(2), 94. https://doi.org/10.3847/1538-3881/aa9e09

[4] Ferrari, A., Tourneret, J. Y., & Schmider, F. X. (1999). Detection of extra solar planets using parametric modeling. ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 3(7), 1369–1372. https://doi.org/10.1109/icassp.1999.756235

[5] Peach, N., Basseville, M., & Nikiforov, I. V. (1995). Detection of Abrupt Changes: Theory and Applications. Journal of the Royal Statistical Society. Series A (Statistics in Society), 158(1), 185. https://doi.org/10.2307/2983416

[6] J.Y. Toumeret. Detection and Estimation of Abrupt Changes Contaminated by Multiplicative Gaussian Noise. EURASIP Signal Processing, 68(3):259-270, 1998.

[7] Parviainen, H. (2015). PYTRANSIT: Fast and easy exoplanet transit modelling in PYTHON. Monthly Notices of the Royal Astronomical Society, 450(3), 3233–3238. https://doi.org/10.1093/mnras/stv894

[8] S. Srikanthan, A. Kumar and R. Gupta, "Implementing the dynamic time warping algorithm in multithreaded environments for real time and unsupervised pattern discovery," 2011 2nd

International Conference on Computer and Communication Technology (ICCCT-2011), Allahabad, 2011, pp. 394-398.

[9] Kate, R.J. Using dynamic time warping distances as features for improved time series classification. Data Min Knowl Disc 30, 283–312 (2016). https://doi.org/10.1007/s10618-015-0418-x

[10] Sangjan, T., Boongoen, T., Iam-On, N., & Mullaney, J. (2019). Classification of Astronomical Objects Using Light Curve Profile. 2019 IEEE Eurasia Conference on IOT, Communication and Engineering, ECICE 2019, 494–497. https://doi.org/10.1109/ECICE47484.2019.8942673

[11] Mahabal, A., Sheth, K., Gieseke, F., Pai, A., Djorgovski, S. G., Drake, A. J., & Graham, M. J. (2018). Deep-learnt classification of light curves. 2017 IEEE Symposium Series on Computational Intelligence, SSCI 2017 - Proceedings, 2018-January, 1–8. https://doi.org/10.1109/SSCI.2017.8280984

[12] Hinners, T. A., Tat, K., & Thorp, R. (2018). Machine Learning Techniques for Stellar Light Curve Classification. The Astronomical Journal, 156(1), 7. https://doi.org/10.3847/1538-3881/aac16d

[13] Bugueno, M., Mena, F., & Araya, M. (2018). Refining exoplanet detection using supervised learning and feature engineering. Proceedings - 2018 44th Latin American Computing Conference, CLEI 2018, 278–287. https://doi.org/10.1109/CLEI.2018.00041

[14] M. Falk, F. Marohn, R. Michel, D. Hofmann, M. Macke, C. Spachmann, and S. Englert, "A first course on time series analysis: Examples with sas [version 2012. august. 01]," 2012.

[15] S. Dieleman, K. W. Willett, and J. Dambre, "Rotation-invariant convolutional neural networks for galaxy morphology prediction," MNRAS, vol. 450, pp. 1441–1459, Jun. 2015.

[16] G. Cabrera-Vives, I. Reyes, F. Forster, P. A. Estevez, and J. Maureira, "Supernovae detection by using convolutional neural networks," in 2016 International Joint Conference on Neural Networks, IJCNN 2016, Vancouver, BC, Canada, July 24-29, 2016, 2016, pp. 251–258.

[17] Bing, H 2017, „Statistical Machine Translation Algorithm Based on Improved Neural Network, 2017 International Conference on Robots Intelligent System (ICRIS), pp. 294– 297. Available from: http://ieeexplore.ieee.org/document/8101404/.

[18] Justusson, B. I. (1981). Median filtering: statistical properties. In Two-dimensional digital signal processing II. https://doi.org/10.1007/bfb0057597

[19] Silva, Diego Alves de Souza, Vinícius Batista, Gustavo. (2013). Time Series Classification Using Compression Distance of Recurrence Plots. Proceedings - IEEE International Conference on Data Mining, ICDM. 687-696. 10.1109/ICDM.2013.128.

[20] Salehizadeh, Sma Chon, Kaye. (2018). A Novel Motion Artifact Detection Algorithm Based on Recurrence Plot Processing of Photoplethysmogram Signal. 620-624. 10.1109/ICTC.2018.8539413.