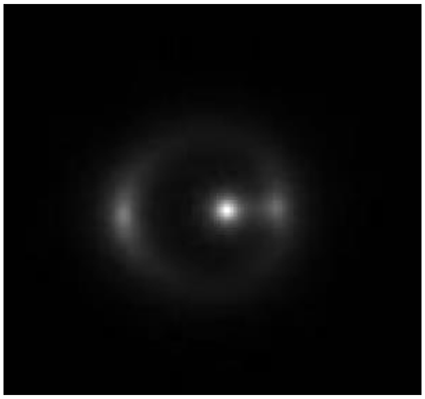
Project Proposal for Google Summer of Code (GSOC) 2021

Equivariant Neural Networks for Decoding Dark Matter



SUBMITTED BY: **Asad Imtiaz Malik**

Date of Submission: 13/04/2021

PERSONAL DETAILS

PERSONAL DETAILS	
Name	Asad Imtiaz Malik
Timezone	Pakistan UTC+05:00
Program	BSc Computer Science
University	National University of Science and Technology (NUST)
Email	aimalik.bscs19seecs@seecs.edu.pk
Github	github.com/asadimtiazmalik
Time Availability	7 - 8 hours per day

WHY ML4SCI?

I have always been fascinated by the vastness of our universe and the nature of the complex yet beautiful structures that exist in it. This is one of the reasons why I was so intrigued when I read about the ML4SCI organization and the projects they were offering. I want to work on this project with ML4SCI because it will help me grow as a Machine Learning Engineer and a researcher. The DeepLense Pipeline uses strong gravitational lensing for dark matter searches. This project can help scientists get a better understanding of the distribution and population statistics of the substructure of dark matter and this will unravel new horizons. Contributing to this ground breaking work in any way would be nothing less than an honor and that one of the reasons why I chose ML4SCI. I believe considering the use of Equivariant Neural Networks in the study of dark matter promises great results. Taking this project with ML4SCI meant more to me due to the presence of great mentorship that can help me push this project in the right direction.

SYNOPSIS

One of the most pressing questions in physics today, is the true identity of dark matter. To date this study has been elusive. Although there are many potential candidates that may qualify as dark matter, there has been no concrete information as to what is the true identity of the dark matter. The only chance of knowing the true identity of dark matter is by observing the gravitational interactions of the substructure of the dark matter halos. The substructure of the superfluid dark matter appears in the form of a sub halo substructure (spherical) or a vortex substructure.

Strong gravitational lensing has proven to be a powerful probe of substructure of dark matter for analyzing its nature. The substructures of superfluid dark matter i.e. vortex and sub halo substructures of the dark matter can be simulated using strong gravitational lensing. This information can be further processed by passing it through a machine learning pipeline.

DeepLense works by utilizing the information provided by the strong gravitational lensing of the substructure of superfluid dark matter. It passes it through a deep learning pipeline that estimates the population level quantities of the dark matter substructures. It does this by using Convolutional Neural Networks for classification among dark matter sub halos with no substructure, vortex substructure and sub halo substructure.

In addition to that, there has also been work related to Auto encoders to reconstruct galaxy-galaxy strong lensing images to observe the distribution of images with no substructure. This is performed as an unsupervised anomaly detection task. The pipeline also has functionality for using auto encoder to reconstruct images with vortex and sub halo substructures.

Although a considerable amount of work has been done to implement unsupervised and supervised machine learning algorithms, the potential of using an equivariant neural network for classification is yet to be explored. The primary aim of this project is to create a python based framework that implements equivariant neural networks for performing a multi class classification of galaxy-galaxy strong lensing images. The project further aims to introduce a Group Equivariant Neural Network to the DeepLense pipeline which uses filters that group the features based on equivariant transformations of the image. Traditional convolutional neural networks are equivariant to translations of their input but are invariant to other isometric transformations such as rotation. This project aims to explore this problem by creating a Convolutional Network that is equivariant to rotational transformations. Consequently, exploring how equivariance affects the classification of strong lensing images based on their morphology.

PROJECT GOALS

Objectives

1. Pre-emptive Research
2. Simulating the strong lensing images with no substructure, sub halo and vortex substructure
3. Designing and training the Equivariant Neural Network
4. Measuring its performance relative to other models
5. Write a detailed documentation about all my code and summarize my work

Tasks

1. **Pre-emptive Research**
    a. Go through the PyAutoLens documentation thoroughly reading how to simulate
    b. Go through the e2cnn library to completely understand all the functionalities for designing an effective equivariant neural network
    c. Go through the equivariant convolutional neural networks such as R2Conv, etc.
    d. Go through the groupy library for creating equivariant neural network
    e. Choose one of the two libraries for developing your network

2. **Simulating the strong lensing images with no substructure, sub halo and vortex substructure using PyAutoLens**
    a. Simulating the strong lensing images with no substructure, sub halo and vortex substructures
        - Determine the right set of methods for simulating the images of the substructures
        - Vortex substructures are considered a string of uniform mass density, ensure that while simulation this structure is taken into consideration. This can be done by assuming each sub halo of the substructure as a point mass. When studied together as a part of the substructure they appear to be a string of mass.
        - Simulate the sub halo sub structure based on the parameters specified in the "Decoding Dark Matter without Supervision" Paper.

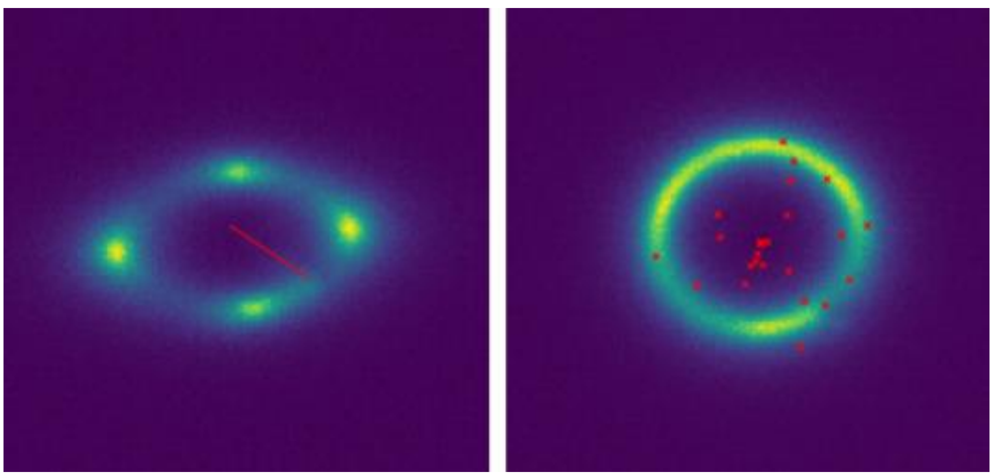
    b. Simulate two different models (Model A and B) for simulating the images. Each model should differ to the other in SNR (signal to noise ratio)
        - Design a method for creating two models with differing difficulty levels with one being the easier to train(with low value of SNR)

    c. Assign labels to the collected data dividing them between all the three classes
    d. Detailed documentation after the task is completed

3. **Designing and training the Equivariant Neural Network using the e2cnn or groupy library**

   a. The need for exploring an Equivariant Neural Network

      The equivariant design can greatly help while working with the substructures of dark matter. The substructures of the dark matter are shown in the figure below.



      It can be seen from the image how mass is distributed across the substructures. This can affect the lensed image in varying degrees.

      The images can be lensed in different orientation and might be transformations of one another. Inherently the images will still belong to the same class of substructure after the transformations. The substructures of dark matter have a great deal of natural symmetry to them and the models can benefit greatly from equivariance in the network architecture. Since Convolutional Neural Networks are only inherently equivariant to translational transformations. Some information might not be extracted due to invariance to symmetrical rotations. Therefore it is important to consider the equivariance in the lensed images of substructure of the dark matter.

b. Data Augmentation
    - Apply the basic transforms on the data
    - According to the paper "Group Equivariant Neural Networks", "G-CNNs benefit from data augmentation in the same way as convolutional networks"
    - Split the data into train, test and validation sets

c. Creating the Network Architecture using e2cnn PyTorch library or groupy
    - Design a suitable architecture for the equivariant neural network
        1. Either create a network over a pre-trained Resnet (groupy)
        2. Or create a network entirely using the functionality provided in the library (e2cnn)
    - Design a network based on different groups of feature map transformations
        1. Group where all the images are randomly rotated
            a. `Rot2dOnR2 (e2cnn.gspaces)`
            b. `P4CNN (groupy)`
        2. Group where all the images are randomly rotated and reflected
            a. `FlipRot2dOnR2 (e2cnn.gspaces)`
            b. `P4MCNN (groupy)`
    - Determine a loss function for training the network
    - Determine the number of layers for each group

d. Hyper tuning and Evaluation
    - Performing hyper tuning on the obtained model by trying different kernel sizes and different steering angles for the kernels.
    - Choose an appropriate evaluation metric for evaluating equivariant neural network.

e. Write a detailed documentation for the code written during this task

4. **Measuring the performance relative to other models**
    a. Measuring the performance of the model amongst different groups of network architecture
        - Choose a suitable metric
        - For each model obtained compare the results using the metric score selected
    b. Measuring the performance of each model (Model A and Model B) which are differently simulated using varying SNR
        - Compute ROC Curves and calculate AUC Score for each of the model
    c. Writing a detailed documentation after the task is completed of the code written for this task

Insights after completing the evaluation exercise

The evaluation exercise had two options,

- Creating an equivariant neural network for binary classification of the dark matter halos with substructure and no substructure
- Creating an unsupervised machine learning algorithm for anomaly detection using the images with no substructure

Although I worked on the first option, due to lack of time I couldn't work on the equivariant aspect with respect to rotational transformations. I worked with a Resnet for performing the binary classification between the two classes of lensed images. The idea behind using a Resnet for this task was to take into consideration the translational equivariant nature of the convolutional Networks. I applied some invariant transforms such as Random Affine and Random Rotation, on the data provided.

This gave me an insight on how to approach the problem with an equivariant neural network. I started researching about equivariant neural networks and the related libraries in PyTorch that can help me write my code.

Timeline

Application Review Period (April 13, 2021 - May 17, 2021)

- Begin Pre-emptive research i.e. the first task
- Learn more about the PyAutoLens , e2cnn and groupy libraries
- Go through previous work done on the DeepLense pipeline

Community Bonding (May 17, 2021 - June 7, 2021)

- Get involved with ML4SCI community
- Reach out to mentors, inquire about their time zones and preferred medium of communication
- Learn more about the other on going ML4SCI projects
- Get acquired with different tools used at ML4SCI that can help set up the Python framework for the project

Week 1 (June 7, 2021 – June 11, 2021)

- Begin Task 2: Simulating the strong lensing images with no substructure, sub halo and vortex substructure using PyAutoLens
- Set up required repositories on Github
- Install any dependent libraries to set up the architecture or to get the results
- Finish Task 2 and add the lensing code to the repository created
- Compile the dataset for further processing next week
- Compile two different datasets namely Model A and Model B, each with different SNR with one dataset being harder to train when compared to the other

Week 2 (June 14, 2021 – June 18, 2021)

- Begin Task 3: Designing and training the equivariant neural network
- Use the dataset acquired after completing task 2 for further processing

Week 3 (June 21, 2021 – June 25, 2021)

- Start in-depth research about equivariant neural networks and its architecture for a classification task
- Install any dependencies for running the library that will be used during development
- Build a robust equivariant neural network using the functionalities of the chosen library
    - Build classes for processing the loss and logging the results during training
    - Build classes to pass the image vector to the model
- Remove any bugs while setting up the network

Week 4 (June 28, 2021 – July 2, 2021)

- Start the training of the neural network for Model A of the dataset
- Build a base line classifier for each network group architecture
- Save the model weights for tuning at a later stage

Week 5 (July 5, 2021 – July 9, 2021)

- Start the training of the neural network for Model B of the dataset
- Build a base line classifier for each network group architecture
- Save the model weights for tuning at a later stage

Evaluation (July 12, 2021 – July 16, 2021)

- Provide a detailed report on the work done up to this point explaining the strategy used
- Provide a detailed documentation for the code written during the last 5 weeks
- Add a README.md file for each model trained explaining the implementations

Week 7 (July 19, 2021 – July 23, 2021)

- Choose an appropriate evaluation metric for testing the model for Model A of the dataset
- Make inferences on the test data
- Tune the model parameters for each model obtained for a group architecture
- Save the model weights and the hyper parameter tuning data to dedicated github repositories

Week 8 (July 26, 2021 – July 30, 2021)

- Choose an appropriate evaluation metric for testing the model for Model B of the dataset
- Make inferences on the test data
- Tune the model parameters for each model obtained for each group architecture
- Save the model weights and the hyper parameter tuning data to dedicated github repositories

Week 9 (August 2, 2021 – August 6, 2021)

- Finish task 3
- Begin task4: Measuring performance relative to other models obtained during training and hyper parameter tuning
- Evaluate the performance of models obtained for each group network architecture in dataset Model A and repeat this for models obtained for in Model B
- Compute the performance in the form of an ROC Curve for the performance of each Model A and B of the dataset
- Add these results to the dedicated github repositories

Week 10 (August 9, 2021 – August 13, 2021

- Compare the performance of the equivariant neural network to a traditional Convolutional Network for the same data
- Use a pre trained Resnet or VGG16 model for this task
- Save the best parameters and models to the github repositories
- Finish task 4

Week 11 (August 16, 2021 – August 23, 2021)

- Begin Task 5: Write a detailed documentation about all my code and summarizing my work
- Document all your findings in form of a report
- Check for any unexplained code in the github repositories
- Align your repositories and add the necessary information for anyone to use the pre trained models and perform inference
- Complete task 5
- Submit the final evaluation with github repositories and detailed documentation of the code written

About Me

I am currently doing a BS degree in Computer Science from the National University of Science and Technology (NUST) and I am in my second year of the program. My motivation to pursue this degree is a career in Artificial Intelligence and Machine Learning. None other than the well-known Katie Bouman introduced me to the field of Artificial Intelligence and Machine Learning. At the time when she created the algorithm, which led to the first-ever picture of a supermassive black hole, I was enrolled in a bachelor's program in Mechanical Engineering at a university in Turkey. I was so inspired by the work that Katie Bouman did that I started independent research about this field of study and soon after I realized my passion for Computer Science. Despite the societal pressures and academic stigma of wasting my years, I left the field of Mechanical Engineering to pursue something that I wanted from all my heart.

I am currently working as a research intern at the National Center for Artificial Intelligence that is a deep learning lab at my university. My area of research concerns Generative Adversarial Networks. As one of the projects for the Lab I implemented Pix2Pix GAN for generating maps from satellite images. I am currently working on sketch completion using Cascaded GAN Architecture.

I am also working as a Machine Learning Engineer at a startup that is working on an agriculture based application for farmers. I am currently working on a model for python based framework that can detect plant disease and produce inference on Node.js.

I recently took part in a Hackathon called Digital Defense Hack where my team landed among the top 30 teams. During this competition, we created an analytics dashboard for detecting fraudulent transactions across accounts. We were successful at locating networks of people transferring high volumes of money in a short period of time using the K means clustering algorithm.

Other commitments during summer

I have my end semester exams from June 7th to June 11th during which I'll be a little busy, other than that I do not have any commitments for summer.

Preferred medium for communication

I am perfectly fine with communicating through Zoom, MS Teams, Skype or any other communication medium

References

Decoding Dark Matter without supervision

Deep Learning the Morphology of Dark Matter Substructure

Group Convolutional Equivariant Networks

Equivariant Steerable CNNs

Groupy Library

E2CNN Library

PyAutoLens