Proposal: Superresolution for Strong Gravitational Lensing

Personal Details

Name: Tanmay Ambadkar
Email: tanmay.ambadkar@gmail.com / tanmay.ambadkar@psu.edu
University: The Pennsylvania State University
Year of study: 1st Year
Major: M.S in Computer Science and Engineering
GitHub: http://github.com/TanmayAmbadkar
Linkedin: https://linkedin.com/in/tanmay-ambadkar
Resume: Tanmay_s Resume.pdf
Timezone: EST/EDT

About Me

I am Tanmay Ambadkar. I am pursuing my MS in Computer Science and Engineering at PennState. I am currently in the 2nd semester of my degree. I am interested in Deep reinforcement learning and computer vision using deep learning.

I am working on a few projects that involve inverse image reconstruction problems like colorization. In addition, I am working on Deep RL-based computer vision problems and composition in reinforcement learning. I work with Pytorch and TensorFlow, with PyTorch being my main library because it is easier to define and manipulate networks and their inputs. I have used TensorFlow for internship projects which have been deployed into production.

I have worked with various deep-learning algorithms for image processing and have taken several courses that have helped me understand how they work. I am keen on working with ML4Sci for the DeepLense projects because I want to apply my deep learning knowledge to a new domain.

<u>The time I can dedicate to the project.</u>

I can dedicate 20 hours a week to the entire project.

Preferred communication medium

I am reachable via email, Google Meet, and Slack.

Why ML4Sci and DeepLense?

I am very interested in applying my deep learning knowledge to various domains. Dark matter has always fascinated me, and the proposed projects have intrigued me.
I would like to understand the work being done at the laboratories and how conventional deep-learning methods can support their research. In addition, super-resolution has always interested me. Deep-learning-based super-resolution has to understand the semantics of the low-res image to then upscale it. I have tried to implement super-resolution algorithms to see which performs the best on a small dataset and how well it has generalized. This is why I would like to work on the Super-resolution project at DeepLense.

Experience with Deep learning

I started working with deep learning in 2019, with my first framework being TensorFlow. I implemented basic projects in computer vision using CNNs for classification. Reinforcement learning interested me, which motivated me to learn Pytorch, as it is much easier to define the training loop for reinforcement learning.

I have taken several courses in deep learning that introduced me to the applications of neural nets. I have worked with residual architectures, self-supervised learning, and GANs in my classes. I have implemented all these algorithms in PyTorch.

My research is at the intersection of deep reinforcement learning and computer vision. My publication - "Deep reinforcement learning approach to predict head movement in 360◦ videos," has been implemented in PyTorch. Using a custom Deep RL framework, I created a multi-frame convolutional network to predict head movement in the videos. In addition, I worked on image colorization, creating a simple network in PyTorch that leverages end–to–end training to show that we don't need a multi-stage very-deep network with millions of parameters to learn semantic transfer of color. The work has been submitted to IEEE SSP 2023.

In addition, I have dabbled with Natural language processing by working with sentence transformers for a project titled Paragraph boundary detection. We have created a custom CNN to process BERT embeddings for binary classification.

Courses at PennState have helped me explore more complex topics in deep learning, like image captioning, image-text retrieval, visual question answering, and neurosymbolic AI, which I have implemented in PyTorch. I am currently studying how to implement robust classifiers in PyTorch.

My internship at Siemens required production-ready code; thus, I developed all networks in TensorFlow, which is easier to deploy. Part of my research at Siemens has been published at IDSTA 2022.

Some relevant projects

The following are some of my relevant deep learning projects that are part of Open source repositories:
1) Deep reinforcement learning approach to predict head movement in 360◦ videos - TanmayAmbadkar/DRL-FOV (github.com)
2) ImageColorNet - Perception consistent image colorization - TanmayAmbadkar/ImageColorNet-Residual-Colorization (github.com)
3) Microsoft Generative Image-to-text transformer - TanmayAmbadkar/GenerativeImage2Text: GIT: A Generative Image-to-text Transformer for Vision and Language (github.com)

Past Open Source experience

I have contributed to open source projects during my undergraduate. My major open-source projects are TanmayAmbadkar/CertificateGenerator (github.com) and TanmayAmbadkar/gymkhana (github.com). Apart from this, all of my research projects are open-source for replication of results and improvement.

The proposal

I wish to apply for "**Superresolution for Strong Gravitational Lensing**." (175 hours)

<u>Summary</u>

Super-resolution is a process using which a high-resolution image is obtained from a low-resolution image. The application is vast. Images captured at a lower resolution due to hardware limitations can be upscaled for better quality. Basic super-resolution algorithms are linear, bicubic, and nearest neighbor interpolation. These algorithms introduce blocks in the image and are unable to understand the semantics of the image. Deep learning-based algorithms use millions of low-res and high-res image pairs to capture these semantics and then estimate the textures and details in the image required for super-resolution. They have proven their efficacy in terms of both metrics and visual results. Several websites allow people to upscale their images using deep learning, giving astonishing results.

Applying trained state-of-art algorithms to Strong Gravitational Lensing is not feasible as the models have never seen this data and could not estimate the patterns in these images. We must adapt these algorithms or create new ones for strong gravitational lensing datasets. In addition, these models can serve as self-supervised models for other tasks like lens-finding and properties of dark matter particle candidates, where multiple labeled examples might not be present.

<u>Milestones</u>

I have created short-term goals that I can follow

1) Get familiarised with the data - Understand the dataset I am working with, what kind of images are present, if any additional variables can be used to improve the superresolution results or if any kind of augmentation can help

2) Apply state-of-art algorithms - Apply algorithms that have been published at top conferences on the dataset and analyze the results.

3) Develop a novel algorithm - Try to develop a novel algorithm that does not need many parameters but can achieve good super-resolution and be at par with the present state-of-art.

4) Extending codebase - Create wrappers for dataset loading, model training, and reporting metrics. This way, anyone in the future would have to develop a PyTorch-based model and change a JSON file to train the model and report the metrics.

5) Self-supervised learning - Apply the trained models to other tasks like lens finding and deep regression with very few labeled pairs to see how they perform.

6) Publication - Publish the results in a peer-reviewed journal (Work can extend beyond the contributor timeline).

Current implementation

The Specific Test VI. Image Super-resolution" provided a dataset of 2000 strong lensing images. I implemented the SRGAN algorithm, an efficient and popular algorithm in the deep learning community. The implementation has two parts, 1st being the generator training on the MSE loss and the GAN training on the discriminator loss.

The results showed that training the generator alone was much more effective than training the GAN. The results are part of this repository - deeplense-tests/Test 6 at main · TanmayAmbadkar/deeplense-tests (github.com). Training for just 30 epochs yielded excellent and usable results, with no overfitting on the training set. This means we can use very few image pairs in the future to train this algorithm.

There are much better implementations than SRGAN which I would like to try and come up with one of my own that does not use a lot of parameters.

Deliverables

1) Model suite - A variety of models, trained weights and their results that can be used for analysis and deployment.
2) Wrappers - Data loading, preprocessing, training and inference wrappers that can be used to extend the model suite
3) Self-supervised learning - Apply the super-resolution models to self-supervised learning tasks in the same domain.

Timeline

The goals in the timeline for the coding period is part of the milestones

Task	Duration
Proposal submission	April 4
• Exploring ML4Sci and DeepLense • Learn in-depth about Strong lensing imaging • Learn about dark matter properties	April 5 - May 4
Community Bonding [1] • Learning about the community • Get to know mentors • Learn about other student projects • Get familiar with the review process	May 4 - May 28
Coding Period starts	May 29 - August 28
Week 1: Understand dataset and its properties and augment the dataset by collection or transformation	May 29 - June 4
Week 2: Creating wrappers for data ingestion, processing, training and inference	June 5 - June 11
Week 3: Look at few papers that are feasible and note down their contributions and applicability for domain	June 12 - June 18
Week 4, 5 and 6: Implement chosen algorithms and report findings [2]	June 18 - July 9
Week 7: Evaluations	July 10 - July 14
Week 7, 8: Implement an architecture that is shallow but suitable for super-resolution of strong-lensing images [3]	July 10 - July 22
Week 9: Commit all developed models and weights and clean up repository by using linters.	July 23 - July 29
Week 10, 11: Apply models to other tasks like lens finding and deep regression and report findings	July 30 - August 12

Week 12: Create a wrapper to import trained models for self-supervised learning tasks	August 13 - August 21
Week 13: Final submission and cleaning up work[4]	August 21 - August 28
Submissions and Final Evaluations	August 28 onwards

[1] I have an exam on May 5, so I will need May 4 and 5 to prepare for the exam.
[2] I have chosen 3 weeks for model implementation because I will be implementing at least 4 algorithms throughout the period and will need the time for model training and testing.
[3] I will be starting to document my findings as a report along with developing the shallow architecture, which can be used for a potential publication
[4] This week and onwards will be used for writing the manuscript and looking for a publication avenue
[5] Since there are 4 weeks of community bonding, I can utilize the time to start researching the datasets and their properties and learn more about the past work that has been done to solve this task

If I cannot complete the target due to unforeseen circumstances (health, family problems), I will put in extra work to ensure the future targets are met. The timeline has been set so that I have 2-3 extra days for every milestone, considering I work 15 hours a week. I do not have any other work/research commitments, so I can dedicate more time if there are delays.

Target

My goal with this project is to understand the domain of strong image lensing and dark matter, as I have not explored this as part of my deep learning research. I want to apply my knowledge to scientific domains that have not used deep learning. Even if I do not get selected, I would like to be in touch with the mentors and contribute towards the development of such algorithms, enhance my knowledge, and be a part of this contribution to science and open source.

DeepLense test solutions - [TanmayAmbadkar/deeplense-tests (github.com)]()