0

A PROJECT REPORT ON

Astronomical Image colorization and super-resolution using GANs

SUBMITTED TOWARDS THE
PARTIAL FULFILMENT OF THE REQUIREMENTS OF

BACHELOR OF ENGINEERING (Computer Engineering)

BY

Shreyas Kalvankar	Exam No:
Hrushikesh Pandit	Exam No:
Pranav Parwate	Exam No:
Atharva Patil	Exam No:

Under The Guidance of

Prof. Dr. S.M. Kamalapur

Department of Computer Engineering
K. K. Wagh Institute of Engineering Education & Research
Hirabai Haridas Vidyanagari, Amrutdham, Panchavati, Nashik-422003
Savitribai Phule Pune University

A. Y. 2020-21 Sem I

K. K. Wagh Institute of Engineering Education and Research
Department of Computer Engineering

CERTIFICATE

This is to certify that the Project Titled

Astronomical Image colorization and super-resolution using GANs

Submitted by

Shreyas Kalvankar	Exam No:
Hrushikesh Pandit	Exam No:
Pranav Parwate	Exam No:
Atharva Patil	Exam No:

is a bonafide work carried out by Students under the supervision of Prof. Dr. S.M. Kamalapur and it is submitted towards the partial fulfilment of the requirement of Bachelor of Engineering (Computer Engineering) Project during academic year 2020-21.

Prof. Dr. S.M. Kamalapur
Internal Guide
Department of Computer Engineering

Prof. Dr. S. S. Sane
Head
Department of Computer Engineering

Abstract

Automated colorization of black and white images has been subject to much research within the computer vision and machine learning communities. Beyond simply being fascinating from an aesthetic and artificial intelligence perspective, such capability has broad practical applications. It is an area of research that possesses great potentials in applications: from black and white photo reconstruction, image augmentation, video restoration to image enhancement for improved interpretability.

Image downscaling is an innately lossy process. The principal objective of super resolution imaging is to reconstruct a low resolution image into a high resolution one based on a set of low-resolution images to rectify the limitations that existed while the procurement of the original low-resolution images. This is to insure better visualization and recognition for either scientific or non-scientific purposes. No matter how good an upscaling algorithm is, there will always be some amount of high frequency data lost from a downscale-upscale function performed on the image. Ultimately, even the best upscaling algorithms cannot effectively reconstruct data that does not exist. Traditional methods for image upsampling rely on low-information, smooth interpolation between known pixels. Such methods can be treated as a convolution with a kernel encoding no information about the original image. A solution to the problem is by using Generative Adversarial Networks (GANs) to hallucinate high-frequency data in a super-resolved image that does not exist in the smaller image. Although they increase the resolution of an image, they fail to produce the clarity desired in the super-resolution task. By using the above mentioned method, not a perfect reconstruction can be obtained albeit instead a rather plausible guess can be made at what the lost data might be, constrained to reality by a loss function penalizing deviations from the ground truth image. A huge number of raw images lie unprocessed and unseen in the Hubble Legacy Archives. These raw images are typically low-resolution, black and white and unfit to be shared with the world. It takes huge amounts of hours to process them. This processing is necessary because astronomers often struggle to distinguish objects from the raw images. Random and synthetic noise from the sensors in the telescope, changing optical characteristics in

the system and noise from other bodies in the universe all make the processing further necessary. Furthermore, colorization is needed to help highlight small features that ordinarily wouldn't be able to be picked out against noise of the image. The processing of the images is so time consuming that the images are rarely seen by human eyes. The problem is only likely to get worse. Not only is new data being continuously produced by Hubble Telescope, but new telescopes are soon to come online. A simplification of image processing by using artificial image colorization and super-resolution can be done in an automated fashion to make it easier for astronomers to visually identify and analyze objects in Hubble dataset.

Acknowledgments

please enter text here.

Student Name1
Student Name2
Student Name3
Student Name4
(B.E. Computer Engg.)

INDEX

tentsfinish

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

1.1 PROJECT IDEA

- Project Idea

1.2 MOTIVATION OF THE PROJECT

- Motivation of the Project

1.3 LITERATURE SURVEY

- Review of the papers, Description , Mathematical Terms

CHAPTER 2

PROBLEM DEFINITION AND SCOPE

2.1 PROBLEM STATEMENT

Description of Problem

2.1.1 Goals and objectives

Goal and Objectives:

- Overall goals and objectives of software, input and output description with necessary syntax, format etc are described

2.1.2 Statement of scope

- A description of the software with Size of input, bounds on input, input validation, input dependency, i/o state diagram, Major inputs, and outputs are described without regard to implementation detail.
- The scope identifies what the product is and is not, what it will and won?t do, what it will and wont contain.

2.2 MAJOR CONSTRAINTS

- Any constraints that will impact the manner in which the software is to be specified, designed, implemented or tested are noted here.

2.3 METHODOLOGIES OF PROBLEM SOLVING AND EFFICIENCY ISSUES

- The single problem can be solved by different solutions. This considers the performance parameters for each approach. Thus considers the efficiency issues.

2.4 SCENARIO IN WHICH MULTI-CORE, EMBEDDED AND DISTRIBUTED COMPUTING USED

Explain the scenario in which multi-core, embedded and distributed computing methodology can be applied.

2.5 OUTCOME

- Outcome of the project

2.6 APPLICATIONS

- Applications of Project Currently, given the sheer number of the raw and unprocessed images in Hubble Legacy Archives, much of the images are not workable for scientific evaluation. The main application of building a GAN and automating the upscaling and colorization of these images is to help in visual classification for astronomers. Through a high resolution and colourized image, objects which would've been imperceptible to the human eye could be now visible for visual inspection. While upscaling is expected to address the poor quality of the original images, colourization will help distinguish astronomical objects and activites from the noise generated by various factors.

2.7 HARDWARE RESOURCES REQUIRED

The project is based Machine Learning, and the use of Tensorflow-GPU brought forward the need for a very high end hardware. Google Colab (or Colaboratory) is a free Jupyter notebook environment offered by Google which runs notebooks from Python kernels and uses Google Drive for storage.

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	Intel(R) Xeon(R) CPU @ 2.20GHz	Training the model on CPU
2	GPU	Nvidia K80-12GB-0.82GHz	To train model on GPU
3	RAM	12GB	To load the Dataset

Table 2.1: Hardware Requirements

2.8 SOFTWARE RESOURCES REQUIRED

Platform :

1. Operating System: Windows 10

2. Platform: Google Colab

3. Programming Language: Python 3

CHAPTER 3

PROJECT PLAN

3.1 PROJECT ESTIMATES

Use Waterfall model and associated streams derived from assignments 1,2, 3, 4 and 5(Annex A and B) for estimation.

3.1.1 Reconciled Estimates

3.1.1.1 Cost Estimate

3.1.1.2 Time Estimates

3.1.2 Project Resources

Project resources [People, Hardware, Software, Tools and other resources] based on Memory Sharing, IPC, and Concurrency derived using appendices to be referred.

3.2 RISK MANAGEMENT

This section discusses Project risks and the approach to managing them.

3.2.1 Risk Identification

For risks identification, review of scope document, requirements specifications and schedule is done. Answers to questionnaire revealed some risks. Each risk is categorized as per the categories mentioned in [1]. Please refer table 3.1 for all the risks. You can refereed following risk identification questionnaire.

1. Have top software and customer managers formally committed to support the project?
2. Are end-users enthusiastically committed to the project and the system/product to be built?
3. Are requirements fully understood by the software engineering team and its customers?
4. Have customers been involved fully in the definition of requirements?
5. Do end-users have realistic expectations?

6. Does the software engineering team have the right mix of skills?
7. Are project requirements stable?
8. Is the number of people on the project team adequate to do the job?
9. Do all customer/user constituencies agree on the importance of the project and on the requirements for the system/product to be built?

3.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Description 1	Low	Low	High	High
2	Description 2	Low	Low	High	High

Table 3.1: Risk Table

Probability	Value	Description
High	Probability of occurrence is	$> 75\%$
Medium	Probability of occurrence is	$26 - 75\%$
Low	Probability of occurrence is	$< 25\%$

Table 3.2: Risk Probability definitions [1]

Impact	Value	Description
Very high	$> 10\%$	Schedule impact or Unacceptable quality
High	$5 - 10\%$	Schedule impact or Some parts of the project have low quality
Medium	$< 5\%$	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 3.3: Risk Impact definitions [1]

3.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Risk ID	1
Risk Description	Description 1
Category	Development Environment.
Source	Software requirement Specification document.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Strategy
Risk Status	Occurred

Risk ID	2
Risk Description	Description 2
Category	Requirements
Source	Software Design Specification documentation review.
Probability	Low
Impact	High
Response	Mitigate
Strategy	Better testing will resolve this issue.
Risk Status	Identified

3.3 PROJECT SCHEDULE

3.3.1 Project task set

Major Tasks in the Project stages are:

- Task 1:
- Task 2:
- Task 3:

Risk ID	3
Risk Description	Description 3
Category	Technology
Source	This was identified during early development and testing.
Probability	Low
Impact	Very High
Response	Accept
Strategy	Example Running Service Registry behind proxy balancer
Risk Status	Identified

- Task 4:
- Task 5:

3.3.2 Task network

Project tasks and their dependencies are noted in this diagrammatic form.

3.3.3 Timeline Chart

A project timeline chart is presented. This may include a time line for the entire project. Above points should be covered in Project Planner as Annex C and you can mention here Please refer Annex C for the planner

3.4 TEAM ORGANIZATION

- Team of 4 members.
- 1 Project Guide

3.4.1 Team structure

- 1 Team Leader
- Total 4 Team members

3.4.2 Management reporting and communication

- Weekly meetings with Project Guide about the work done.

- Project Idea Presentation

CHAPTER 4

SOFTWARE REQUIREMENT SPECIFICATION

4.1 INTRODUCTION

4.1.1 Purpose and Scope of Document

The purpose of SRS and what it covers is to be stated

4.1.2 Overview of responsibilities of Developer

What all activities carried out by developer?

4.2 USAGE SCENARIO

This section provides various usage scenarios for the system to be developed.

4.2.1 User profiles

The profiles of all user categories are described here.(Actors and their Description)

4.2.2 Use-cases

All use-cases for the software are presented. Description of all main Use cases using use case template is to be provided.

Sr No.	Use Case	Description	Actors	Assumptions
1	Use Case 1	Description	Actors	Assumption

Table 4.1: Use Cases

4.2.3 Use Case View

Use Case Diagram. Example is given below

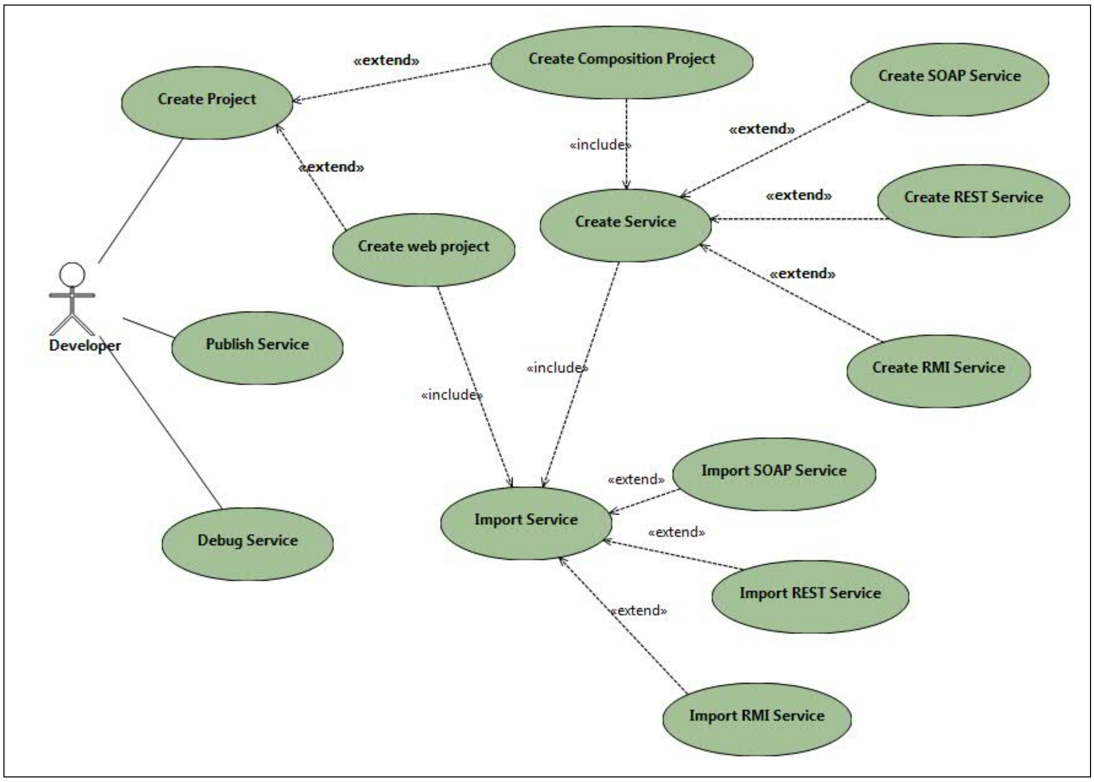

Figure 4.1: Use case diagram

4.3 DATA MODEL AND DESCRIPTION

4.3.1 Data Description

Data objects that will be managed/manipulated by the software are described in this section. The database entities or files or data structures required to be described. For data objects details can be given as below

4.3.2 Data objects and Relationships

Data objects and their major attributes and relationships among data objects are described using an ERD- like form.

4.4 FUNCTIONAL MODEL AND DESCRIPTION

A description of each major software function, along with data flow (structured analysis) or class hierarchy (Analysis Class diagram with class description for object oriented system) is presented.

4.4.1 Data Flow Diagram

4.4.1.1 Level 0 Data Flow Diagram

4.4.1.2 Level 1 Data Flow Diagram

4.4.2 Description of functions

A description of each software function is presented. A processing narrative for function n is presented.(Steps)/ Activity Diagrams. For Example Refer 4.2

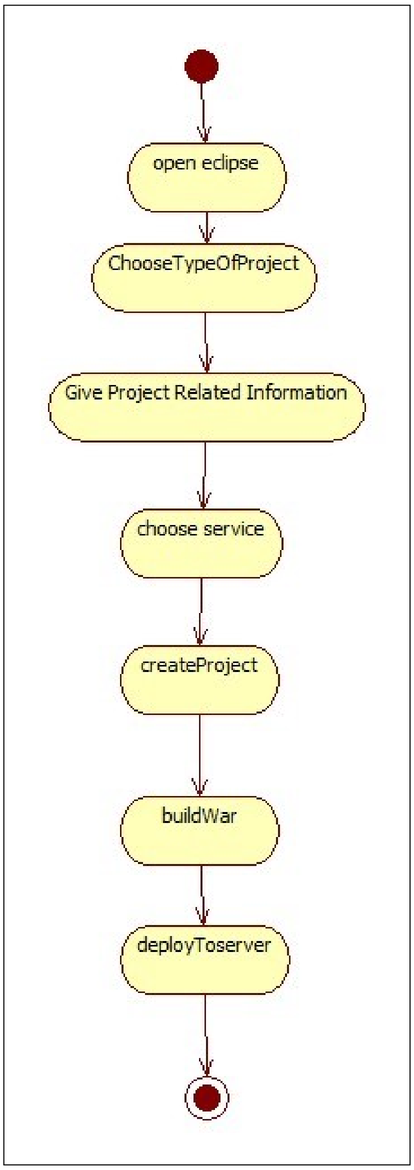

Figure 4.2: Activity diagram

4.4.3 Activity Diagram:

- The Activity diagram represents the steps taken.

4.4.4 Non Functional Requirements:

- Interface Requirements
- Performance Requirements
- Software quality attributes such as availability [related to Reliability], modifiability [includes portability, reusability, scalability] , performance, security, testability and usability[includes self adaptability and user adaptability]

4.4.5 State Diagram:

State Transition Diagram
Fig.4.3 example shows the state transition diagram of Cloud SDK. The states are represented in ovals and state of system gets changed when certain events occur. The transitions from one state to the other are represented by arrows. The Figure shows important states and events that occur while creating new project.

4.4.6 Design Constraints

Any design constraints that will impact the subsystem are noted.

4.4.7 Software Interface Description

The software interface(s)to the outside world is(are) described. The requirements for interfaces to other devices/systems/networks/human are stated.

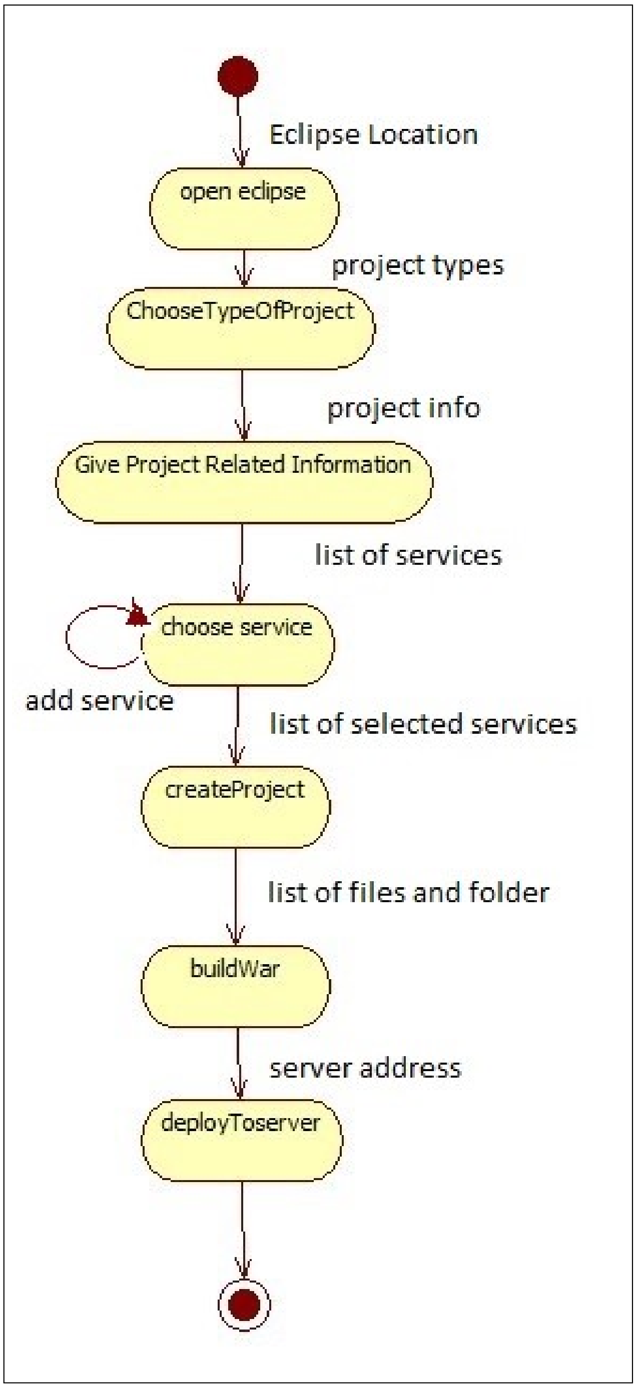

Figure 4.3: State transition diagram

CHAPTER 5

DETAILED DESIGN DOCUMENT

5.1 INTRODUCTION

This document specifies the design that is used to solve the problem of Product.

5.2 ARCHITECTURAL DESIGN

A description of the program architecture is presented. Subsystem design or Block diagram,Package Diagram,Deployment diagram with description is to be presented.

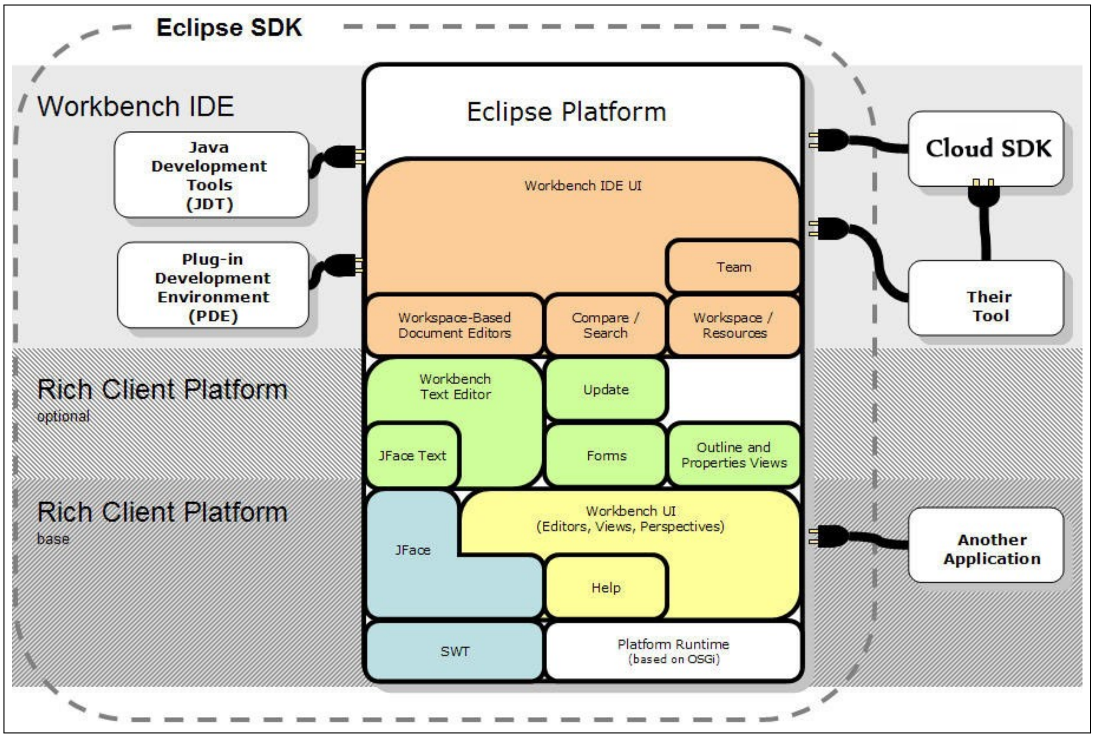

Figure 5.1: Architecture diagram

5.3 DATA DESIGN

A description of all data structures including internal, global, and temporary data structures, database design (tables), file formats.

5.3.1 Internal software data structure

Data structures that are passed among components the software are described.

5.3.2 Global data structure

Data structured that are available to major portions of the architecture are described.

5.3.3 Temporary data structure

Files created for interim use are described.

5.3.4 Database description

Database(s) / Files created/used as part of the application is(are) described.

CHAPTER 6

DATASET AND EXPERIMENTAL SETUP

CHAPTER 7

SUMMARY AND CONCLUSION

7.1 SUMMARY

Low Resolution and Black and White Images in the Hubble Legacy Archive are needed to be upscaled and colourized for better visual discernment of the astromers. Images are scraped and a dataset is formed which is cleaned and is then fed to a two Generative Adversarial Networks. The images of size 64 X 64 X 3 are initially colorized and then upscaled to a size of (64.n) X (64.n) X 3. The output images are visually superior and of greater use to astromers.

7.2 CONCLUSION

The images are upscaled and colourized using a completely automated algorithm which uses Deep Learning. Generative Adversarial Networks are thus successfully used for the implementation and the images obtained are in public forum to be used for research, it is anticipated that this will aid the astronomers vastly in their efforts.

REFERENCES

[1] R. S. Pressman, *Software Engineering (3rd Ed.): A Practitioner's Approach*. New York, NY, USA: McGraw-Hill, Inc., 1992.

ANNEXURE A

MATHEMATICAL MODEL

ANNEXURE B

PLAGIARISM REPORT

ANNEXURE C

PAPER PUBLISHED (IF ANY)

ANNEXURE D

SPONSORSHIP DETAIL (IF ANY)