A

Project Report

On

AI Photo Editor

In partial fulfillment of requirements for

System Design Practice

B.Tech, Sem VII

Submitted By:

1. **Arshit Vaghasiya (IT - 133)**
2. **Darshan Vesatiya (IT - 137)**

Under the Guidance of

Prof. Deepak C. Vegda

DEPARTMENT OF INFORMATION TECHNOLOGY

FACULTY OF TECHNOLOGY, DHARMSINH DESAI UNIVERSITY COLLEGE ROAD, NADIAD- 387001

CANDIDATE'S DECLARATION

We declare that pre-final semester report entitled **"AI Photo Editor"** is our own work conducted under the supervision of the guide **Prof. Deepak C. Vegda.**

We further declare that to the best of our knowledge the report for B.Tech. VII semester does not contain part of the work which has been submitted either in this or any other university without proper citation.

Arshit Vaghasiya
16ITUES044

Darshan Vesatiya
16ITUSF069

Submitted To:

Prof. Deepak C.Vegda,
Department of Information Technology, Faculty of Technology,
Dharmsinh Desai University, Nadiad,
Gujarat.

DHARMSINH DESAI UNIVERSITY
NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project carried out in the subject of Software Design Project, entitled "AI Photo Editor" and recorded in this report is a bonafide report of work of

1) ARSHIT VAGHASIYA	**Roll No. IT - 133**	**ID No: 16ITUES044**
2) DARSHAN VESATIYA	**Roll No. IT - 137**	**ID No: 16ITUSF069**

of Department of Information Technology, semester VII. They were involved in Project work during academic year 2019 -2020.

Prof. Deepak C. Vegda,
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

Prof. (Dr.) V. K. Dabhi,
Head , Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad.
Date:

ACKNOWLEDGEMENT

On completion of project we would like to express our sincere thanks to all those who have guided, advised, inspired and supported during our project work.

Every work that one completes successfully stands on the constant encouragement, good will, and support of the people around. We, hereby, avail this opportunity to express our heartfelt gratitude to a number of people who extended their valuable time, full support and cooperation in developing this project.

We are heartily thankful to the qualified staff of the university and especially to our project guide Prof. Deepak C. Vegda. We believe that his field expertise, problem solving abilities and valuable guidance have made it possible to present such a nice project.

Arshit Vaghasiya
Darshan Vesatiya

TABLE OF CONTENTS

ABSTRACT

Machine Learning has opened plethora of initiatives which has created a digital ecosystem with user as its cynosure. With the huge amount of data available, someone needs to collect it and extract valuble information from it so it can be used to improve the way we're using current technologies.

From the same concept, we've used different Machine Learning models of GAN - Generative Adversarial Network, and algorithms to extract information from the images and created two modules of it summed up as "AI Photo Editor". One of it is capable of removing different mini objects from the image and other one is able to generate the 'Map View' from the given input satellite image. Which can be very useful in the field of Image processing.

Further more it can be very useful by providing dataset/database for the other Machine Learning and Neural Network models.

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 PROJECT OVERVIEW

"AI Photo Editor" is a Machine Learning project, aims to remove the mini objects from the image as well as able to generate the map from the given satellite image. User can upload custom images of their own and can remove the objects from the image in which model will generate background automatically. Also user can generate map view from the given satellite images which can be very useful for developing purpose.

1.2 PURPOSE

Main purpose of this project is to provide the handy tool to users which can edit the photo without any indepth knowledge of the editing and how it works. Also it can be very useful to generate the different datasets for specific models.

1.3 SCOPE

If any user want to remove the entire portion of the image then this machine learning model needs to be trained from the scratch and also if one want to create entire map of the city then he/she needs to pass one by one portion of image to this model – not the whole image.

1.4 OBJECTIVE

This project has the main objective of serving the users, essential tool to edit his/her images, generate maps for personal use as well as commercial use. So the user do not need to worry about how it developed and how it works.

1.5 TECHNOLOGY AND LITERATURE REVIEW

- **Front-end:** HTML, CSS, Bootstrap, Materialize, JavaScript
- **Back-end:** Python, Flask
- **Python**

Python is an interpreted high-level programming language for general purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scale. Rather than having all of its functionality built into

its core, Python was designed to be highly extensible. This compact modularity has made it particularly popular as a means of adding programmable interfaces to existing applications.

- **Development Tools:** Google Colab, Visual Studio Code, Python IDLE
- **Libraries Used:**
    - **TensorFlow:** TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is used for machine learning applications such as neural networks.
    - **Keras:** Keras is an open-source neural-network library written in Python. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, Theano, or PlaidML. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible.
    - **Flask:** Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It is used as backend server to navigate image from GUI to trained model.
    - **Numpy:** NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
    - **OpenCV:** OpenCV is a library of programming functions mainly aimed at real-time computer vision.

2. PROJECT MANAGEMENT

2.1 FEASIBILITY STUDY

2.1.1 Technical Feasibility

Technical analysis evaluates technical merits of the system at the same time collects additional information about performance, reliability, maintainability and productivity. The technical feasibility means that the project can be done with the current equipment, existing software technology and the current knowledge. The present system is technically feasible as it has been developed on Google Colab for professional developers and frontend technologies like HTML, CSS etc. for GUI. We've used Image Inpainting Dataset for object removal and Sat2Map Vector Image Dataset for generting map image from the satellite image.

2.1.2 Time Schedule Feasibility

The Project has simple working and the basic requirement can be satisfied within allotted time period so the time development feasibility is satisfied.

2.1.3 Operational Feasibility

There are mainly three operations:

1) User uploads the image.
2) Image is transferred to python trained model via Flask and processed.
3) Generated image is shown in GUI using Flask.

2.1.4 Implementation Feasibility

The system can be easily implemented as it has been developed via python plus machine learning model trainer – Google Colab. It can be easily implemented by installing python environment and libraries stated above. Implementation feasibility is concerned with specifying external resources and software that will successfully satisfy the requirements.

2.2 PROJECT PLANNING

2.2.1 Project Development Approach and Justification

For Project Development Iterative Waterfall Model is used.The Iterative water fall model approach overcomes the problems associated with the waterfall model approach. If any difficulty or problem encounter in any phase may require going back to the previous phase and performing the required modifications and proceeds sequentially. This backtracking allows modifying any corrections or modifications required in the previous phase.

As illustrated in Fig 2.1, this model divides the cycle into the phases mentioned below:

1. Feasibility Study
2. Requirements analysis and specification
3. Design
4. Coding and Testing Unit
5. Integration and System Testing
6. Maintenance

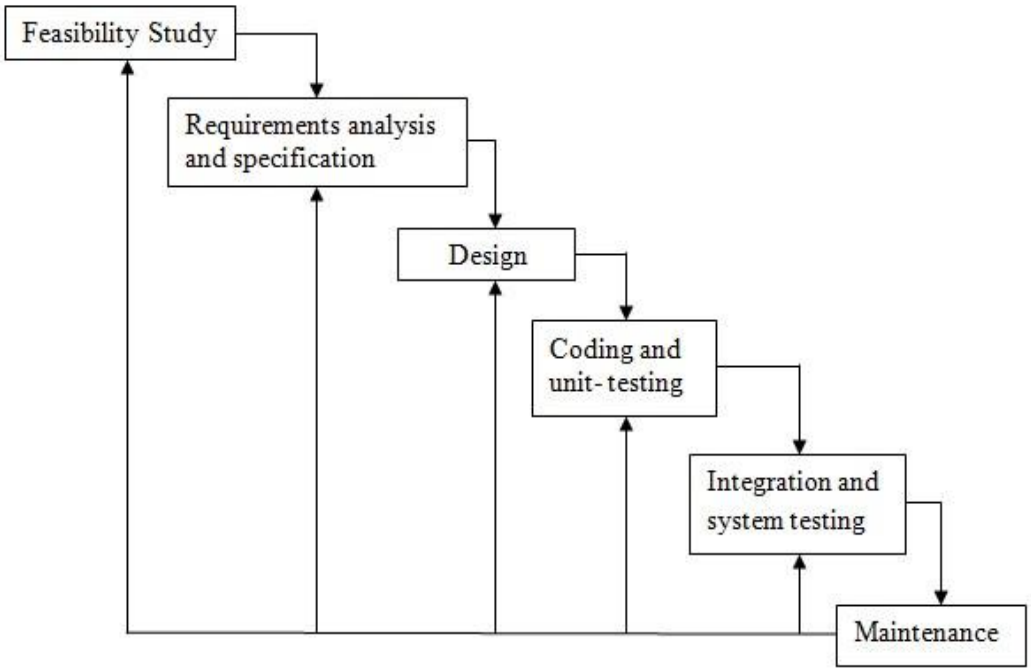

Fig 2.1 Iterative Waterfall Model

Advantages of using Iterative Waterfall Model:

- You are provided the chance to see the potential outcomes of every stage and make changes to areas of concern if necessary. This is one of the reasons that make the iterative model useful.
- Iterative development is more adjustable to changes as it considers each stage like a vital portion of the development cycle.
- The time spent on each successive interval may be lessened depending on how the last stage went and what knowledge was gained from past stages. The system therefore grows through adding new functionalities in the development part of all iterations.

Disadvantages of using Iterative Waterfall Model:

- When using the iterative model people working on the project can get stuck in a loop. Always finding problems than having to go back and design a fix, implement it, then test the system again and finding another problem can mean that the project can run over time and budget.
- Informal requests for improvement after each phase may lead to confusion and may also create scope creep, since user feedback following each phase may lead to increased customer demands. As users see the system develop, they may realize the potential of other system capabilities which would enhance their work, this can be an advantage as much as it can be a disadvantage.

Justification:

- After feasibility study as the functional requirements were almost clear. Here we have decomposed the system into modules. That is why we decided to use iterative waterfall model which is most suitable model here i.e. if we find any difficulty in coding and testing a modification in design can be done easily.

2.2.2 Project Plan

After feasibility study as the functional requirements were almost clear which were decided by our project lead. After analysing and thoroughly understanding the requirements of the application we planned the project.

The project had been planned in mainly two phase: Designing and Implementation. Designing phase consists of the road map of the system. It has been determined which frameworks, libraries, language etc will give us optimal solution for the system and then in Implementation, it has been used to successfully accomplish the project.

2.2.3 Milestones and Deliverables

Timely directions are always required to run a project successfully. Milestones tell the developers how far he has reached and also tell him what things are still left and how to fulfil them. Milestones may be the short report of achievement in project activity that are used by the project manager to check project progress but which are not delivered to the Clients. The deliverables are the project results that are provided to the customer. It is usually delivered at the end of some major project phases.

MILESTONES	DELIVERABLES	PURPOSE
Software Installation and Understanding of Technology.	Had complete knowledge of Google Colab and used libraries.	To be familiar with Google Colab and used libraries.
System feasibility study, Requirement and Analysis.	Functional Specifications. Non Functional Specifications.	It gives exact understanding of the User's requirements.
System Design	Sequence Diagram Use Case Diagram Activity Diagram	It gives the logical Structure that describes the system.
Coding and Unit Testing and corrections if any.	Individually Tested and Functional Modules. Individual Modules for Comment Classification	It gives the required Module.
Integration and System Testing.	The output obtained for the required functionality after implementing and doing various types of testing.	Integrated System is Ready.

Table 2.1 Milestones and Deliverables

2.2.4 Roles and Responsibilities

As only two members were involved in the whole team each of them had to perform all the tasks as the project proceeded through its different phases. This helped each one to develop skills in all the phases.

Name	Role				
	Analysis	Designing	Coding	Testing	Documentation
Arshit Vaghasiya	✓	✓	✓	✓	✓
Darshan Vesatiya	✓	✓	✓	✓	✓

Fig. 2.2.4 Roles and Responsibilities

3. SYSTEM REQUIREMENT

3.1 STUDY OF CURRENT SYSTEM

The current available system does not provide any GUI interface to use this kind of machine learning system which normal user can easily access. So this is very handy tool for the novice users.

3.2 USER CHARACTERISTICS

The targeted users can be any student, researcher or any individual who don't want to code or who don't know how to code for machine learning algorithms but need some handy AI image editor, then using GUI provided in system - they can easily use this system.

3.3 HARDWARE AND SOFTWARE REQUIREMENTS

3.3.1 Software Requirements

- Python 3.0 or above
- TensorFlow Library
- Keras Library
- Flask Library
- Numpy Library
- OpenCV Library

3.3.2 Hardware Requirements

- Inter i3 or above
- Minimum 2GB of RAM

3.4 CONSTRAINTS

3.4.1 Hardware Limitations

Minimum Requirement of Hardware is Intel core i3 processor and 2GB of RAM. If system with hardware less then the stated requirement is used then result generation would take much time and application might get non responding.

3.4.2 Interface To Other Applications

There are no other systems that use this application as an interface.

3.4.3 Reliability Requirements

The application does demand much reliability and it is fully assured that the particular information about the user should be secured and flow is maintained and accessed according to the rights.

3.4.4 Criticality of the Application

It is desktop application and there is no use of personal details, also it don’t require internet connection. So No need to worry about criticality.

3.4.5 Security and Safety considerations

The system desktop application and it doesn’t require any personal details and internet connection so it is secured.

4. SYSTEM ANALYSIS

4.1 REQUIREMENTS OF NEW SYSTEM

4.1.1 User Requirements

Description:

Providing GUI to the user so that user do not need to worry about how and why is process take place. It must generate proper output for any specific input image.

4.1.2 System Requirements

R1. Select Module:

Input: Select either ‘Object Removal’ or ‘Image to Map’.

Output: Related page opens up.

Description: User selects any of the given option and related page opens up.

R2. Upload Image:

Input: User uploads image.

Output: Generated image shown.

Description: If ‘Object Removal’ is selected then user can erase the objetc he/she want. And if ‘Image to Map’ is selected then Map view of satellite is generated.

R3. Output Image:

Input: Generated image from the trained model.

Output: Image is shown in GUI.

Description: After processing the image uploaded from user side – generated image from the trained model is transferred to GUI via Flask.

4.2 FEATURES OF NEW SYSTEM

The new system consists the features of user interactive functionality along with existing features and more accurate object removal and generation of Map from satellite images.

4.3 Use Case Diagram

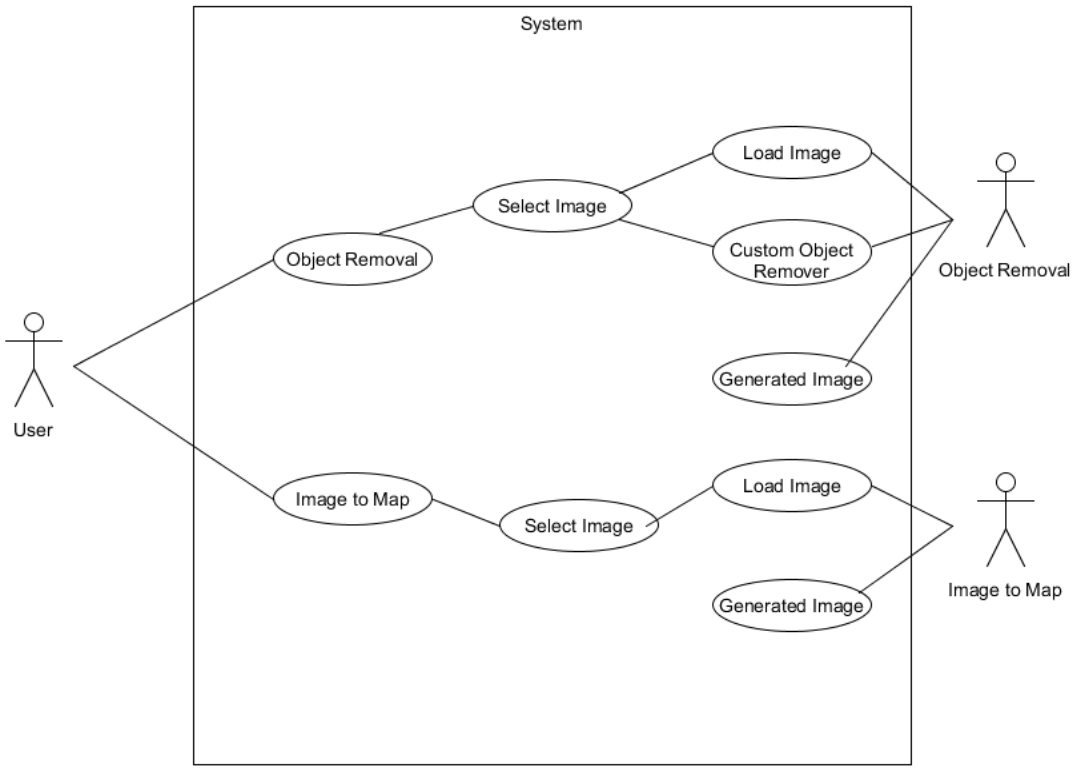

Fig 4.1 Use Case Diagram

5. SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE DESIGN

5.1.1 Activity Diagram

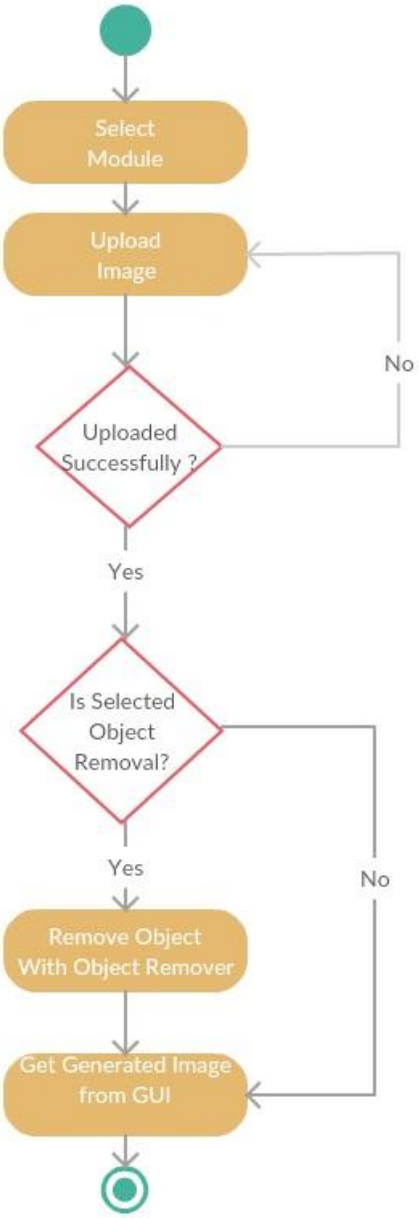

Fig 5.1 Activity Diagram

5.1.2 Sequence Diagram

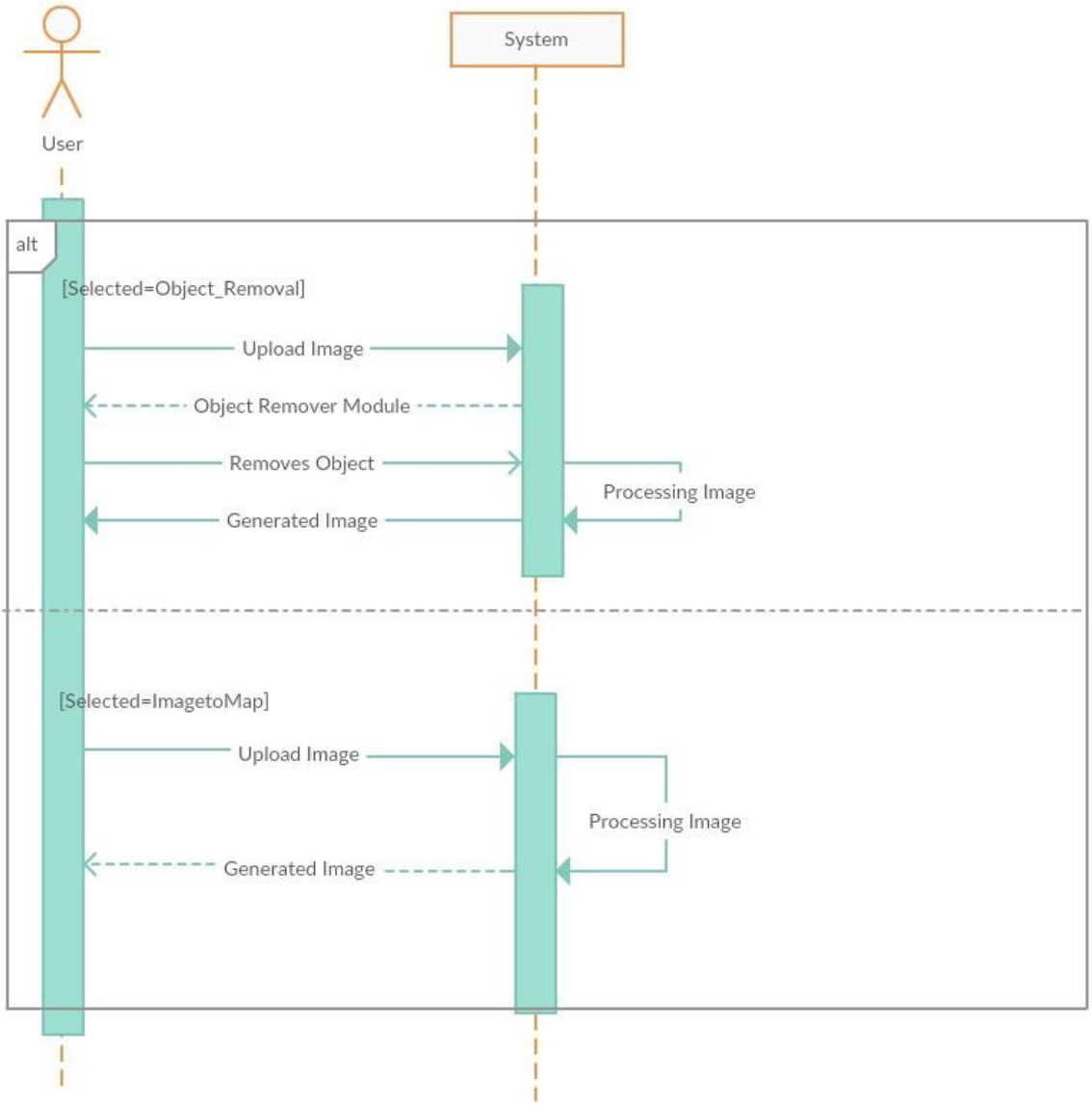

Fig. 5.2 Sequence Diagram

5.1.3 Dataset and Library Design

Datasets contains large numbers of images which is used while training the models for the 'Object Removal' and 'Image to Map'.

For Object Removal: Image Inpainting Dataset

For Image to Map: Sat2Map Vector Image Dataset

- **Libraries Used:**
    - **TensorFlow:** TensorFlow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is used for machine learning applications such as neural networks.
    - **Keras:** Keras is an open-source neural-network library written in Python. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, Theano, or PlaidML. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible.
    - **Flask:** Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It is used as backend server to navigate image from GUI to trained model.
    - **Numpy:** NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
    - **OpenCV:** OpenCV is a library of programming functions mainly aimed at real-time computer vision.

6. Implementation Planning

6.1 IMPLEMENTATION ENVIRONMENT

The application is a multiuser system with GUI. For the implementation of the project, the environment required is Windows 10 or above. The language used is Python and libraries used are TensorFlow, Keras, Flask, Numpy, OpenCV.

6.2 PROGRAMS / MODULES SPECIFICATION

- **GAN(Generative Adversarial Network):**

  The main focus for GAN (Generative Adversarial Networks) is to generate data from scratch, mostly images but other domains including music have been done. But the scope of application is far bigger than this. GAN composes of two deep networks, the generator, and the discriminator.

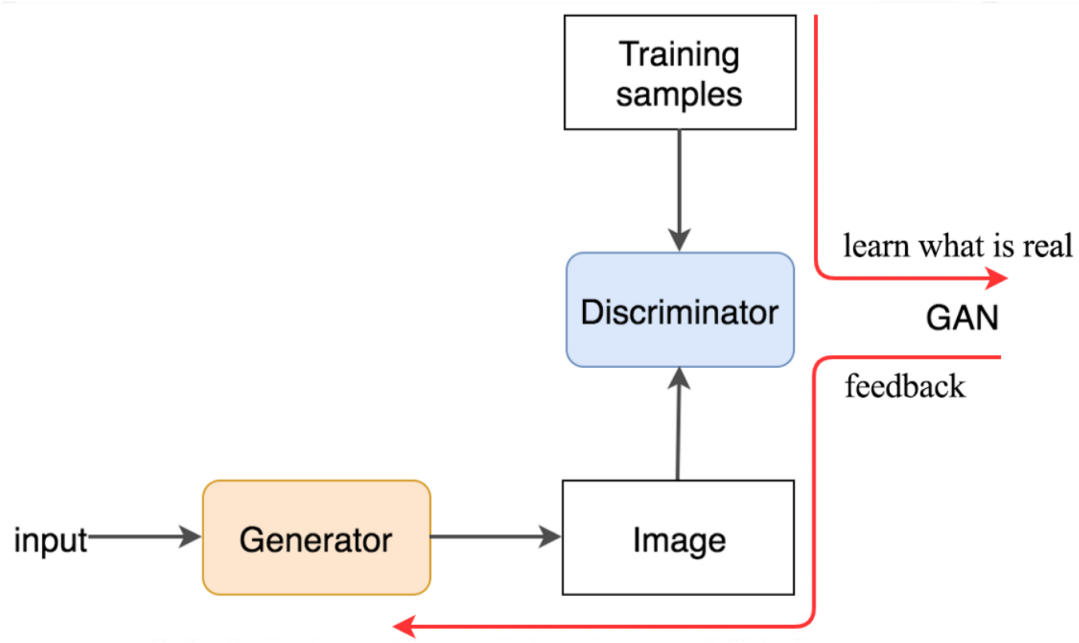

Fig. 6.2.1 GAN Architecture

- **DCGAN (Deep Convolutional Generative Adversarial Networks):**

  DCGAN is one of the popular and successful network design for GAN. It mainly composes of convolution layers without max pooling or fully connected layers. It uses convolutional stride and transposed convolution for the downsampling and the upsampling. The figure below is the network design for the generator.

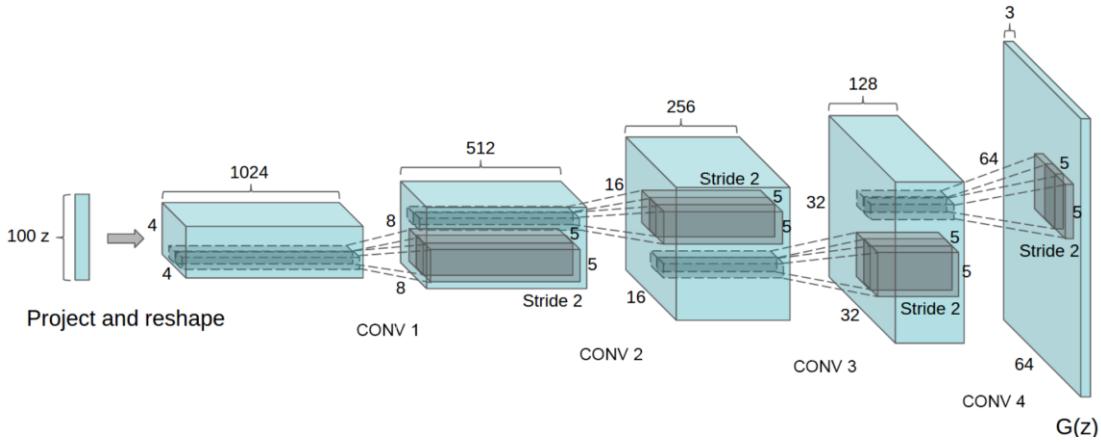

Fig. 6.2.2 DCGAN Architecture

6.2.1 Object Removal

- After the given custom imgae input, Firstly image will be masked with the white colour layer on the object that should be removed.

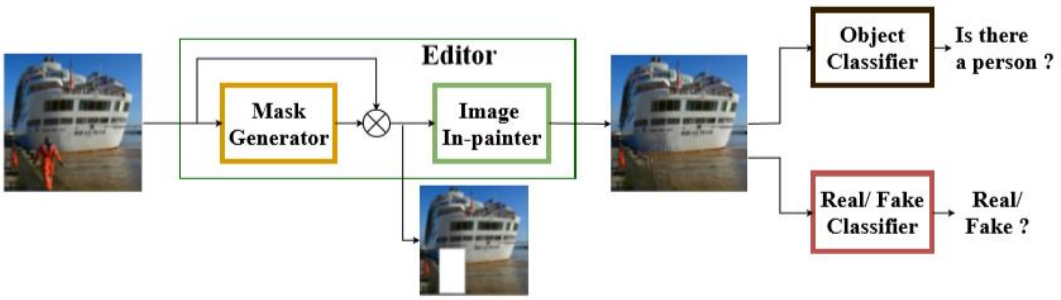

Fig. 6.2.3 Object Removal Process

- Now model will try to generate the background for the removed portion which it can recognize by white spacings.
- Model is trained on ‘Image Inpainting Dataset’ which has collection of images with it’s respected object removed output images.
- While trying to remove the whole mega portion from the image – global or local loss may occur which causes imperfection in generating of the background for the removed object.
- For ex,

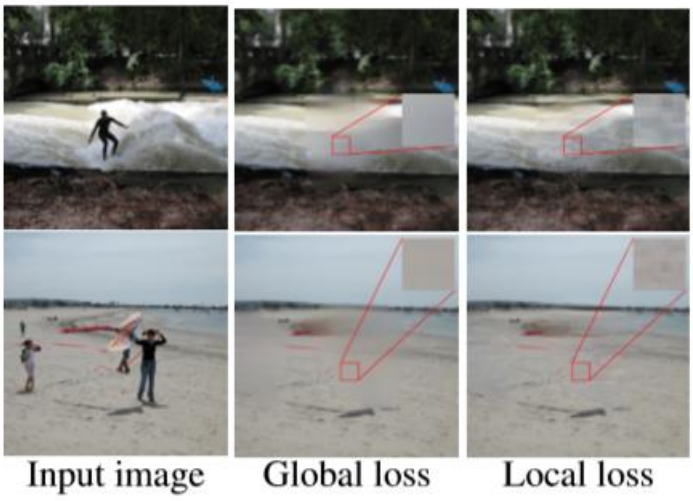

Fig. 6.2.4 Loss in Object Removal

- Dataset used: Image Inpainting Dataset
- Approx training time: 6 hour
- The number of epochs: 50

6.2.2 Image to Map

- Satellite image is given as input via Graphical User Interface, model will firstly create the vector with three channel of given image.
- Now all the vector will be of image size and with value of pixels between 0 to 255.
- Generator is trained on satellite view pixels and it will locate the appropriate pixel on output image.

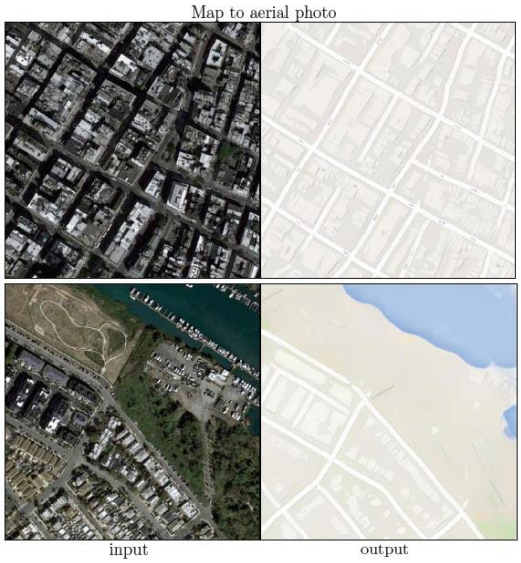

Fig. 6.2.5 Sat2Map Vector Image Dataset

- Descriminator will be used in Generator to decide whether generated image is real pair or fake pair.

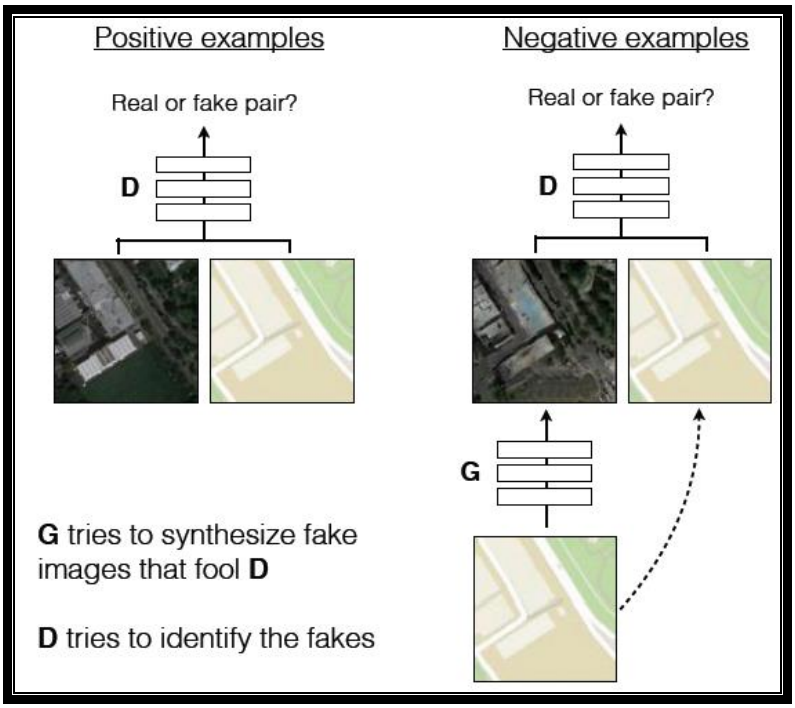

Fig. 6.2.6 Image to Map Process

- Filter in the Generator will pass through the image and will perform metrix multiplication on given image to convert the Satellite image to Map image.
- Dataset used: Sat2Map Vector Image Dataset
- Approx training time: 10 hour
- The number of epochs: 70

7.Testing

7.1 TESTING PLAN

The testing technique that is going to be used in the project is black box testing. In black box testing the expected inputs to the system are applied and only the outputs are checked.

7.2 TESTING STRATEGY

The development process repeats this testing sub-process a number of times for the following phases.

a) Unit Testing.

b) Integration Testing

Unit Testing tests a unit of code (module or program) after coding of that unit is completed.

Integration Testing tests whether the various programs that make up a system, interface with each other as desired, fit together and whether the interfaces between the programs are correct.

Testing is carried out in such a hierarchical manner to ensure that each component is correct and the assembly/combination of components is correct. Merely testing a whole system at the end would most likely throw up errors in components that would be very costly to trace and fix.

7.3 TESTING METHODS

Black Box and White Box Testing:

In black-box testing a software item is viewed as a black box, without knowledge of its internal structure or behaviour. Possible input conditions, based on the specifications (and possible sequences of input conditions), are presented as test cases.

In white-box testing knowledge of internal structure and logic is exploited. Test cases are presented such that possible paths of control flow through the software item are traced. Hence more defects than black-box testing are likely to be found.

The disadvantages are that exhaustive path testing is infeasible and the logic might not conform to specification. Instrumentation techniques can be used to determine the structural system coverage in white box testing. For this purpose, tools or compilers that can insert test probes into the programs can be used.

7.4 TEST CASES

Test Case Id	Description	Prerequisite	Expected Output	Result Output	Pass/Fail
1.1	Do not select any module	System should not be in execution mode.	Output should not be generated.	Output not generated	Pass
1.2	Do not select image file type	System should be in execution mode	Invalid type should be printed	Invalid type printed	Pass
1.3	Select any module	System should be in execution mode	Web page should be returned	Web Page loaded	Pass
1.4	Close browser in between	System should be in execution mode	Process should be terminated	Process terminated	Pass

Fig. 7.4 Test Cases

8. User Manual

A user guide or user's guide, also commonly known as a manual, is a technical communication document intended to give assistance to people using a particular system.

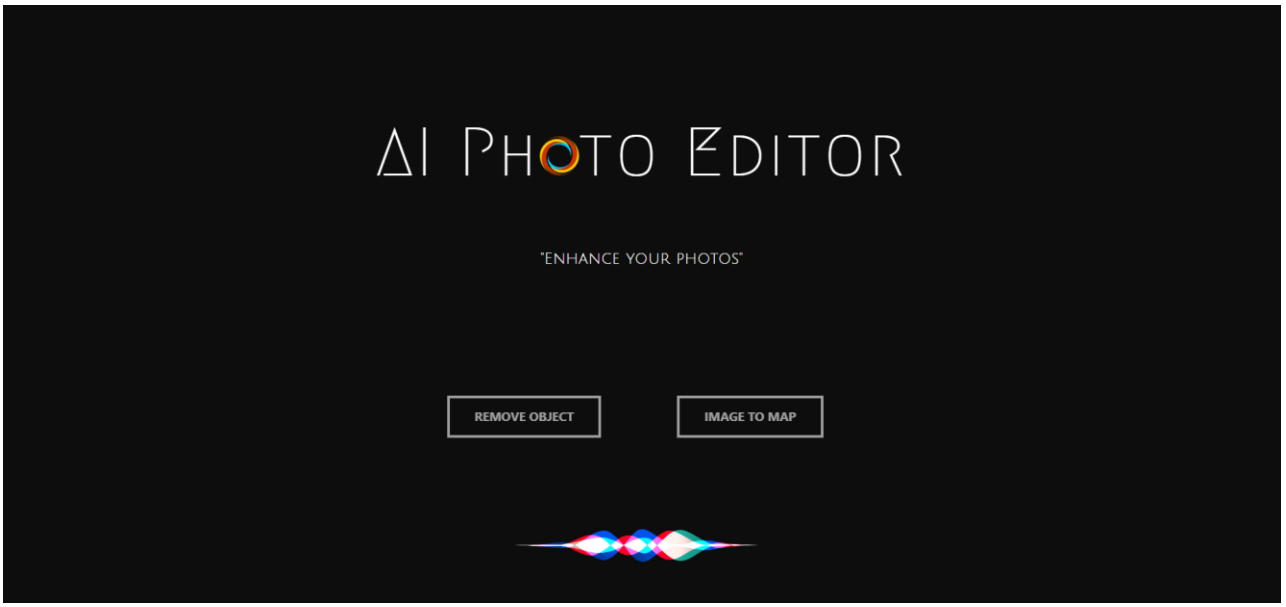

Fig. 8.1 Select Module

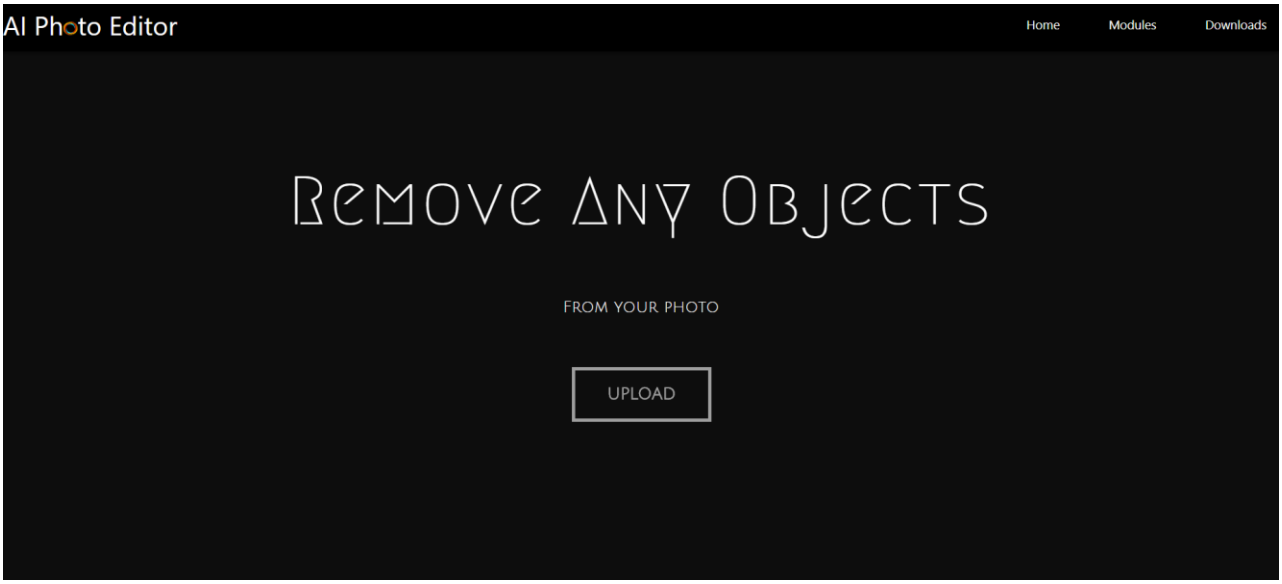

Fig. 8.2.1 Upload Image to remove objects

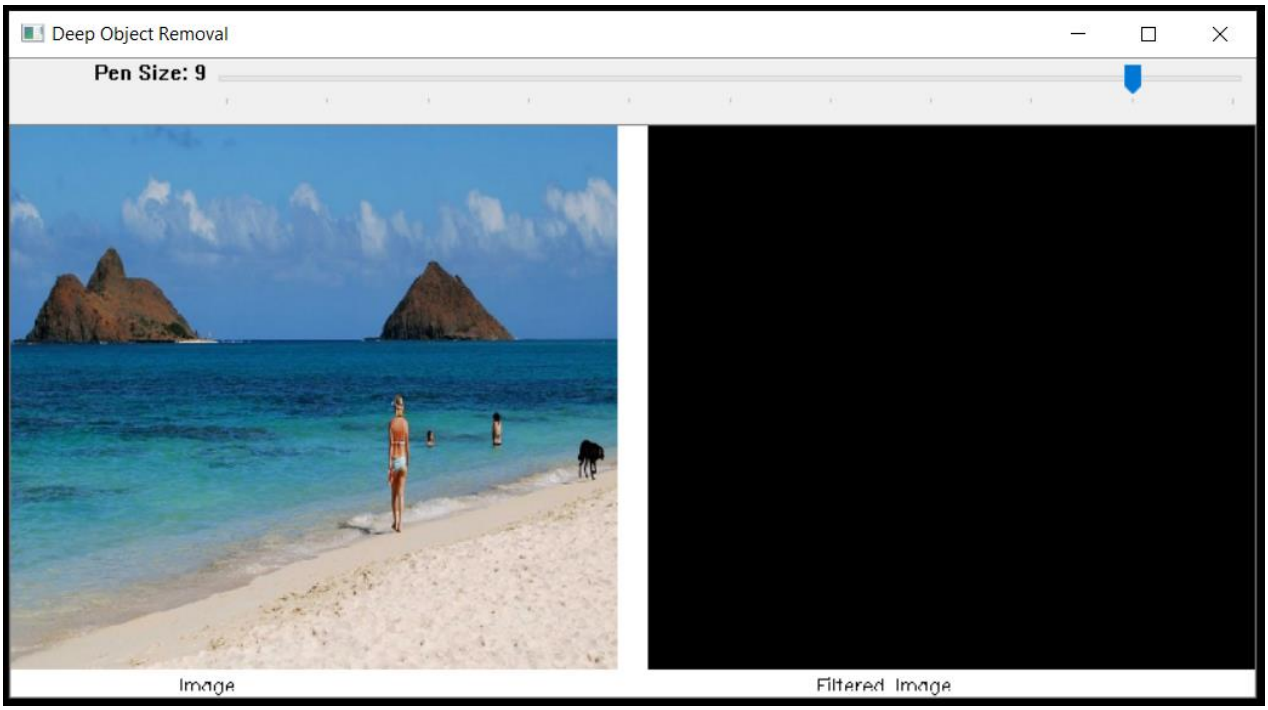

Fig. 8.2.2 Select objects to remove

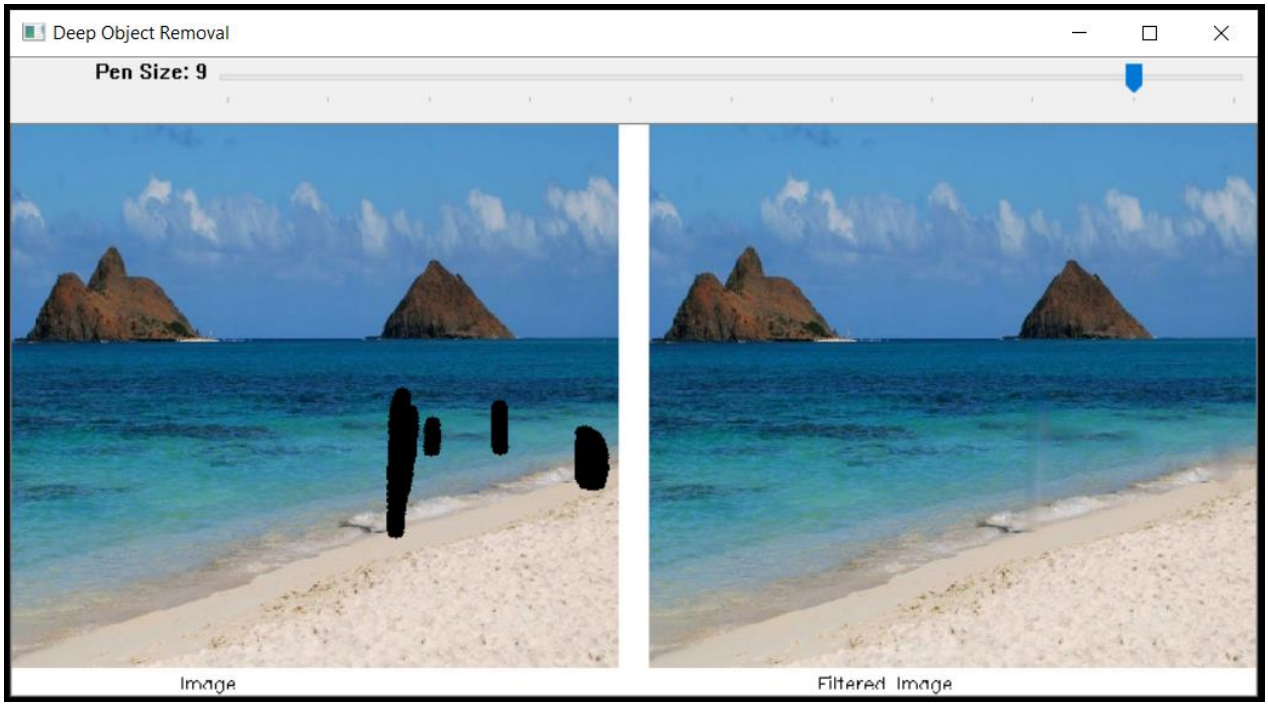

Fig. 8.2.3 Press ‘F’ to generate image, press ‘Esc’ to quit

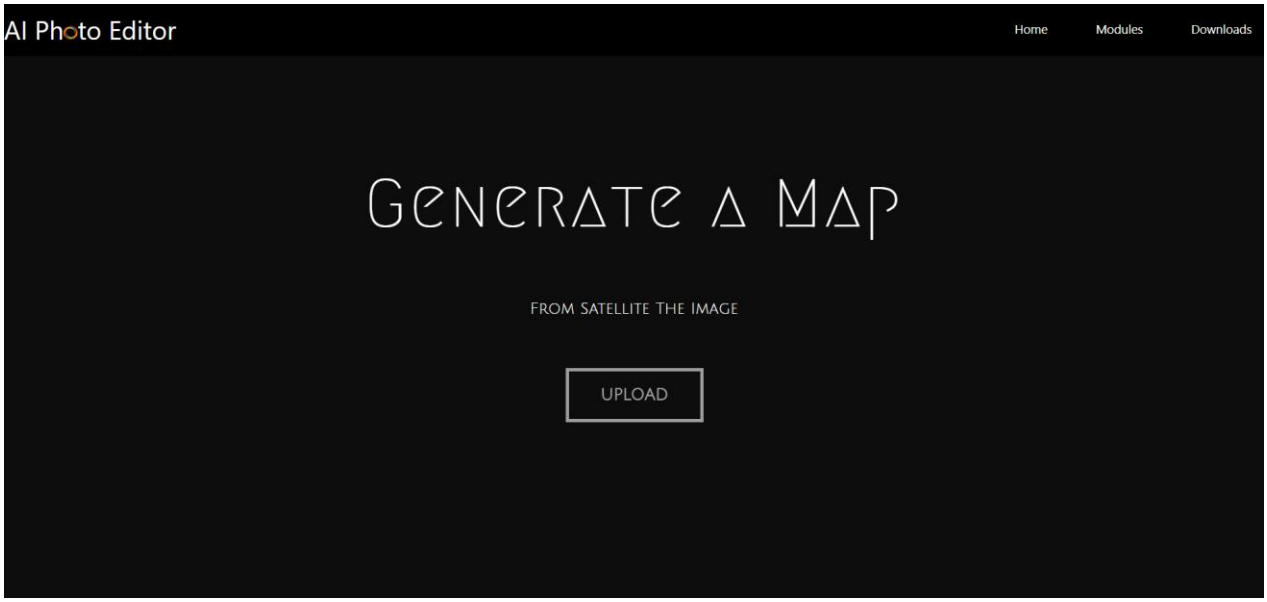

Fig. 8.3.1 Upload Satellite View Image

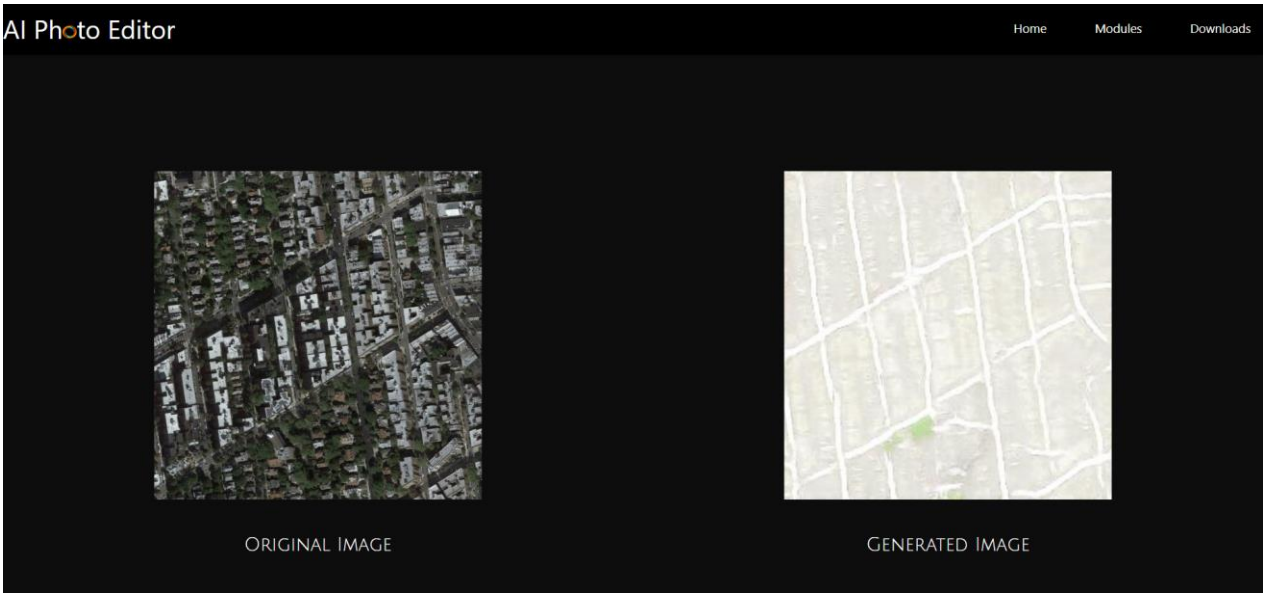

Fig. 8.3.2 Generated Map will be shown

9. LIMITATION AND FUTURE ENHANCEMENT

9.1 LIMITATIONS

Limitation of this project is that sometimes accuracy is not according to our expectation. Sometimes application goes into non responding state, if the system with high configuration is used then this problem is eliminated forever. If anyone don't know regarding the algorithm used in system and want to know how system works then it would be difficult to understand system for them. Espicially while removing some bigger portion of the image in 'Object Removal' module then it may happen that it'll generate distorted image.

9.2 FUTURE ENHANCEMENT

This application can be used to remove entire area from the image to generate appropriate backfround for it. Furthermore, accuracy can be increased by using more enchanced algorithm and hardware for 'Object Removal' as well as 'Image to Map' module.

10. LIMITATION CONCLUSION AND DISCUSSION

10.1 CONCLUSION

Every project work, doesn't matter software project or any other project, could not be the result of sole effort. We tried to give maximum accuracy to remove objects from the images as well as to generate map image from the satellite image but then also its not up-to-date and can be future improved in future as the field of Machine learning is vast and lot more to innovate in it.

10.2 DISCUSSION

10.2.1 Self-Analysis of Project Viabilities

According to us, this project is absolutely a good start for learning the machine learning and gaining hands-on experience on project. It is useful if it is managed according to the goal for which it is made.

10.2.2 Problems Encountered and Possible Solutions

Following are problems encountered during this project,

- Not generated appropriate output.
- Many time Flask server was going to non-responding state.
- Taking more time for predicting output for huge dataset.
- Accuracy was not to the expected level.

10.2.3 Summary of Project Work

It has been successfully completed the project. The prior knowledge of software engineering has helped immensely in overcoming the various roadblocks. We have done work with pre-planned scheduling related with time constraints and result oriented progress in project development.

REFERENCES

- CodePen
  - https://codepen.io
- Flask Docs
  - http://flask.palletsprojects.com/en/1.1.x/
- Kaggle
  - https://www.kaggle.com/datasets
- MaterializeCSS
  - https://materializecss.com/
- Medium – GAN & DCGAN
  - https://medium.com/@jonathan_hui/gan-whats-generative-adversarial-networks-and-its-application-f39ed278ef09
  - https://medium.com/@jonathan_hui/gan-dcgan-deep-convolutional-generative-adversarial-networks-df855c438f
- NumPy
  - https://www.geeksforgeeks.org/python-numpy/
- OpenCV
  - https://docs.opencv.org/
- StackOverflow
  - https://stackoverflow.com
- TensorFlow
  - https://www.tensorflow.org/tutorials