Satellite Image Data Analysis

- Sayali Lokhande
200020041

Topics covered

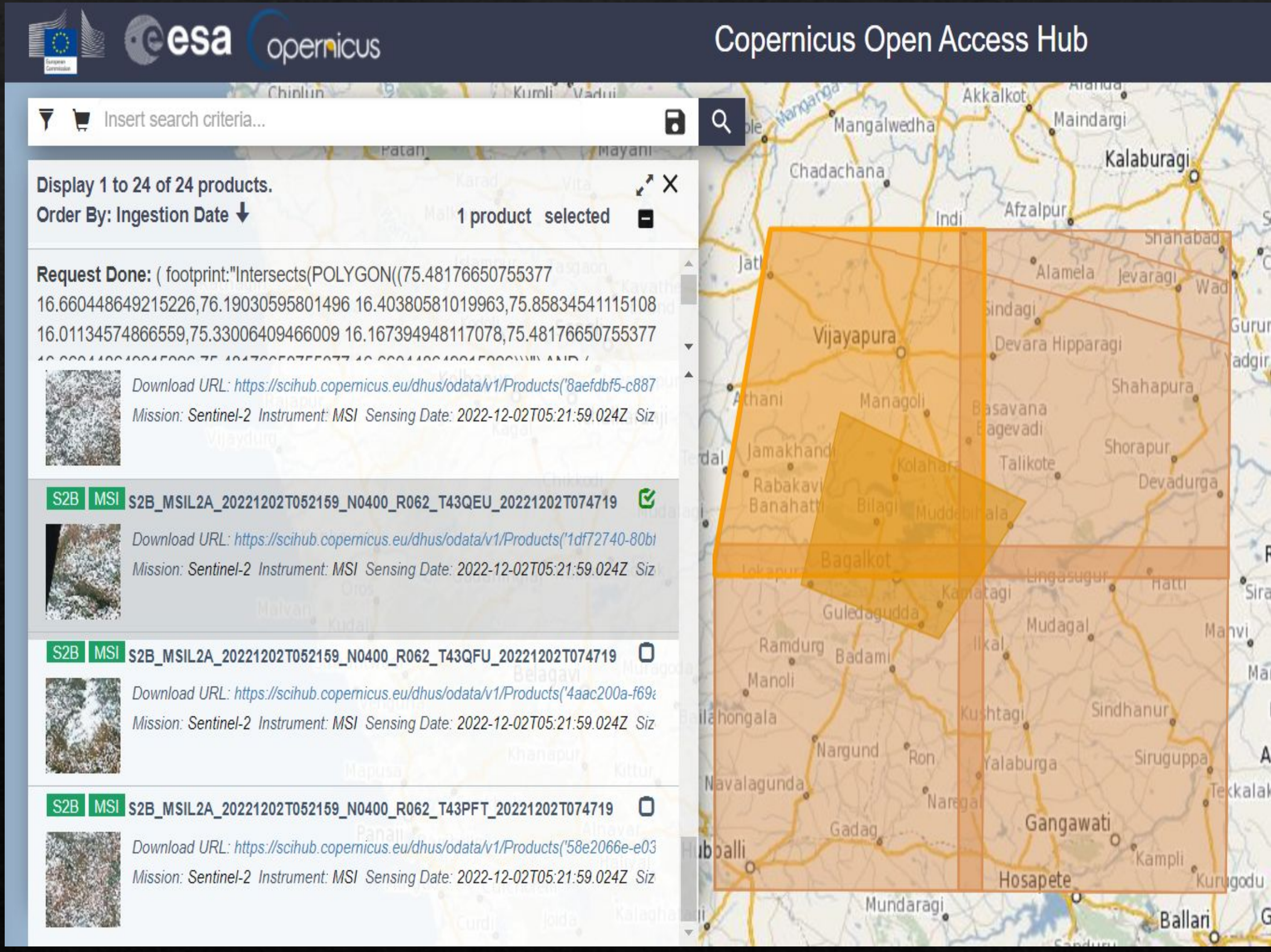
- Data Collection
- QGIS Software
- Cloud Removal Model

Sentinel-2

- Earth observation mission from the Copernicus Programme and acquires optical imagery at high spatial resolution(10 m to 60m) over land and water bodies
- Carries multi-spectral imager
- 13 spectral bands with each band 10,20 or 60 meters in pixel size
- Samples 13 spectral bands: four bands at 10 m, six bands at 20 m and three bands at 60 m spatial resolution
- Blue (B2), green (B3), red (B4), and near-infrared (B8) channels have a 10-meter resolution

Data Collection

- Used Copernicus Open Access Hub which has free access to Sentinel-2 data
- Downloaded Sentinel-2 dataset by selecting Almatti Dam region for each month of 2021 and 2022
- The selected location has a particular tile ID associated with it
- Downloaded only Band-3 and Band-8 images using Python API script



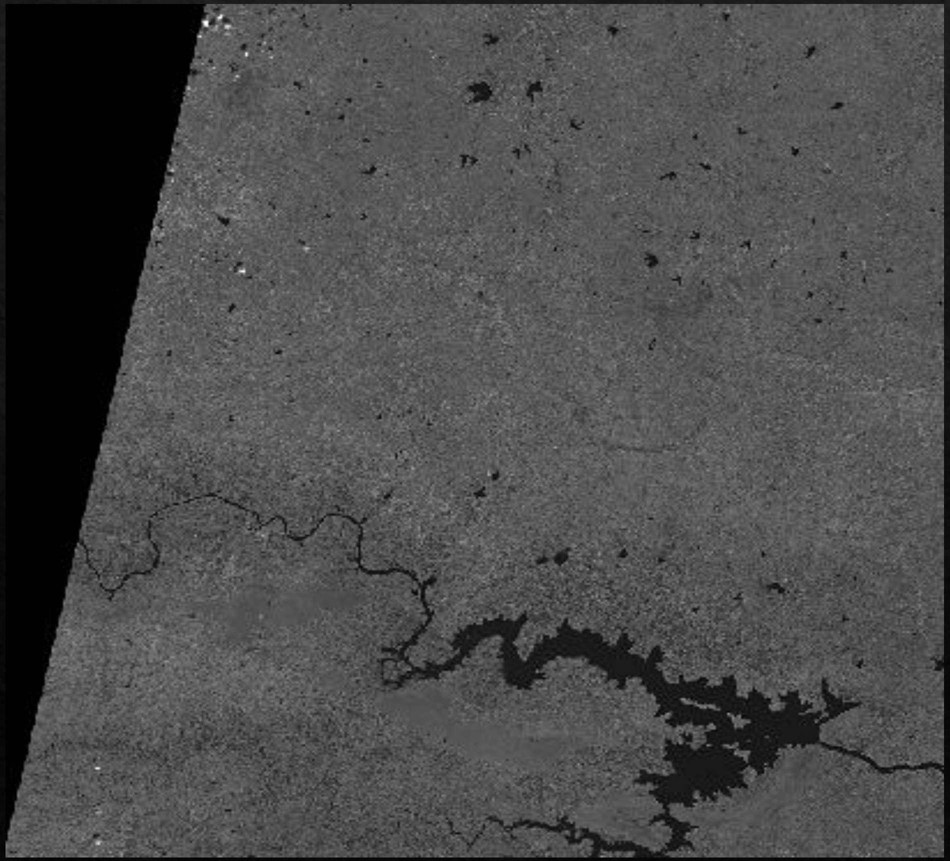
Normalised Difference Water Index (NDWI)

- Used to monitor changes related to water content in water bodies
- Highlights open water features in a satellite image, allowing a water body to stand out against the soil and vegetation.
- As water bodies strongly absorb light in visible to infrared electromagnetic spectrum, NDWI uses green and near infrared bands to highlight water bodies
- NDWI = (Green - NIR) / (Green + NIR)
- For Sentinel-2,

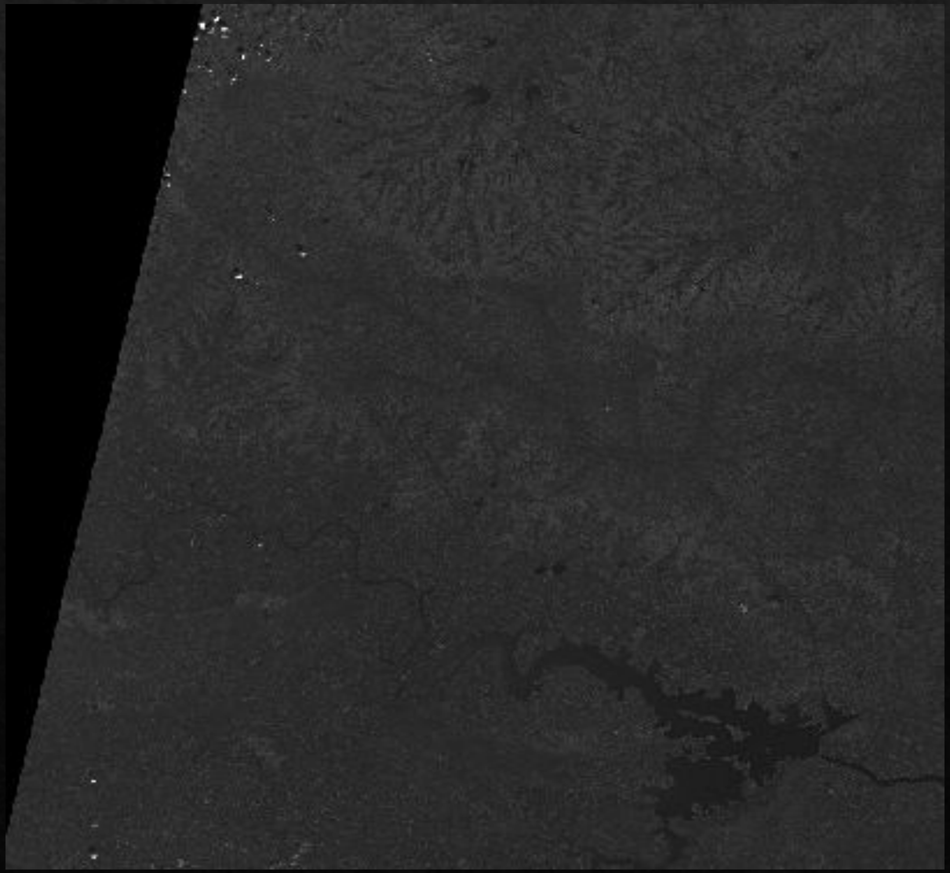
  NDWI = (B03 - B08) / (B03 + B08)

Dataset for Almatti Dam

jp2 file for Band-8 of Almatti Dam



jp2 file for Band-3 of Almatti Dam

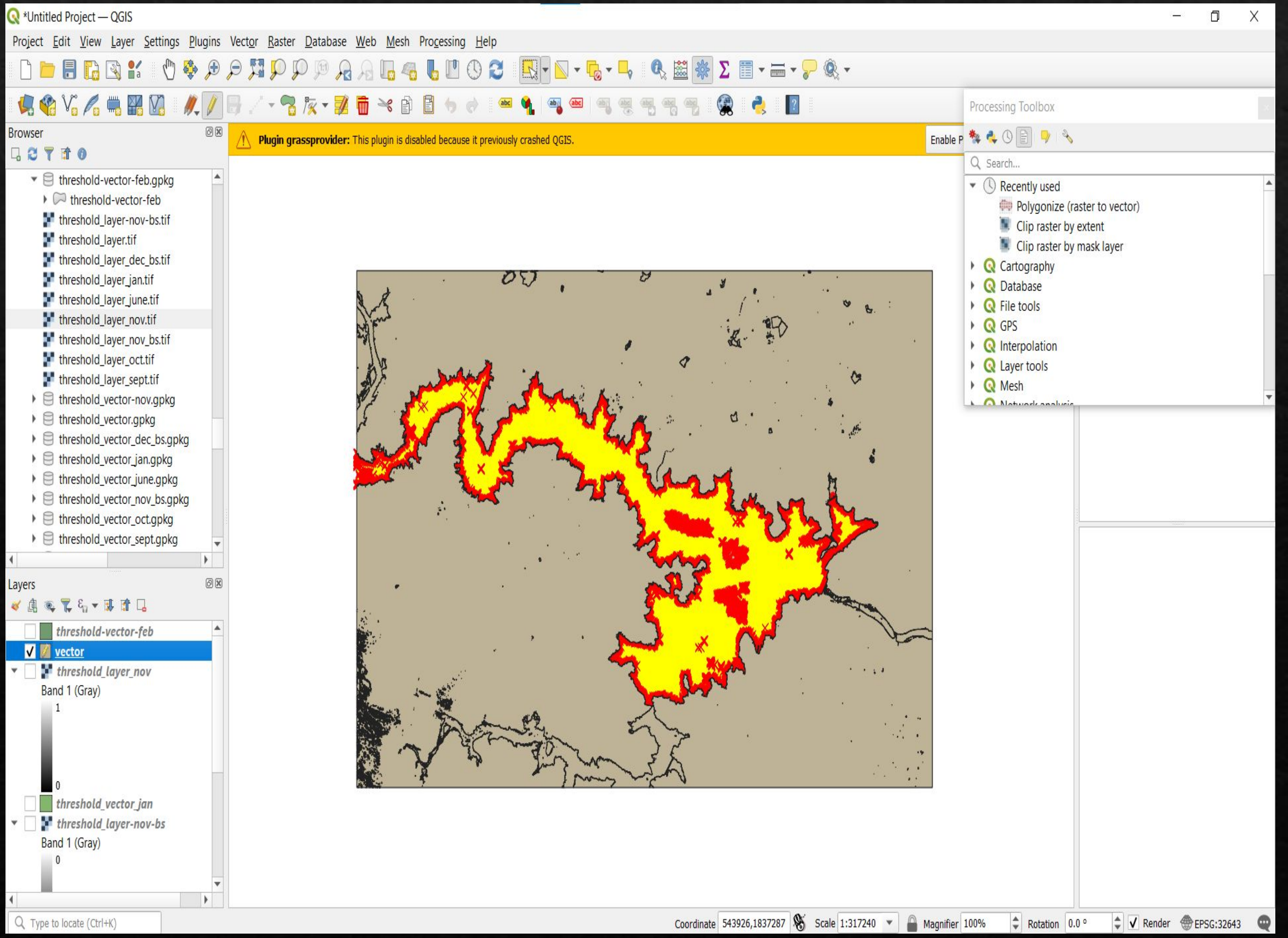


QGIS Software

- Free and open-source desktop geographic information system application that supports viewing, editing, printing, and analysis of geospatial data
- Used downloaded .jp2 files of Band 3 and Band 8 of 10 m resolution
- Calculated NDWI for the water body using Band 3 and Band 8
- Used raster calculator and various clip raster tools to crop out the land part
- Output is .tif file for with only water part

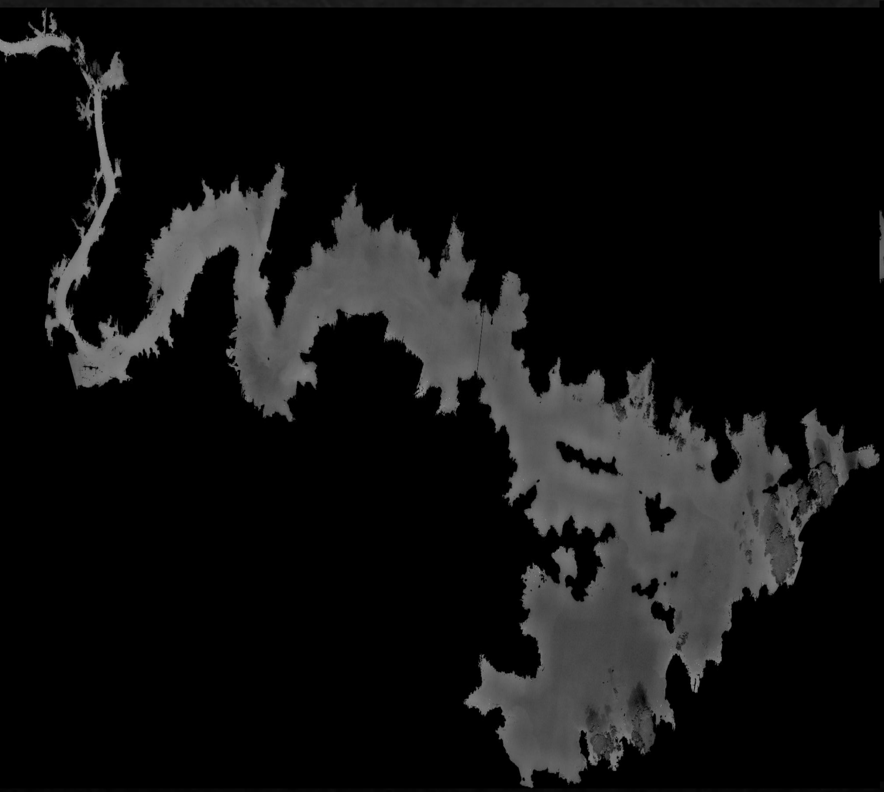
Steps involved in QGIS software

- Selecting required area for both the bands
- Calculating NDWI using raster calculator from the clipped images
- In the NDWI image, set pixel value 1 for water while 0 for land
- Converting the layer to vector using Polygonize Raster to vector
- Cropping out unwanted part from this vector

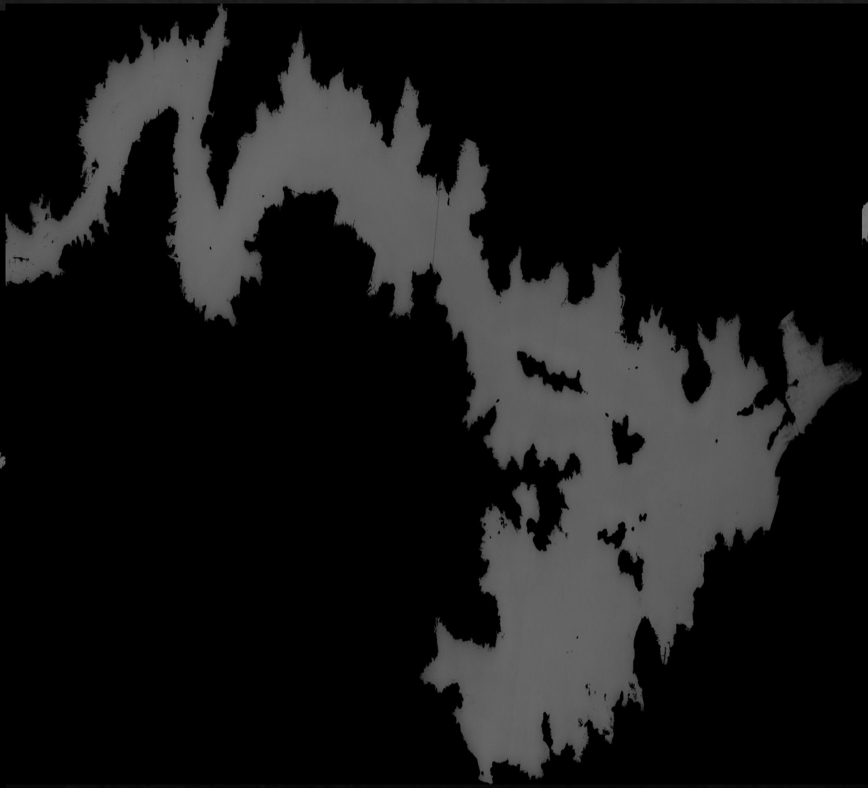


Output .tif Images

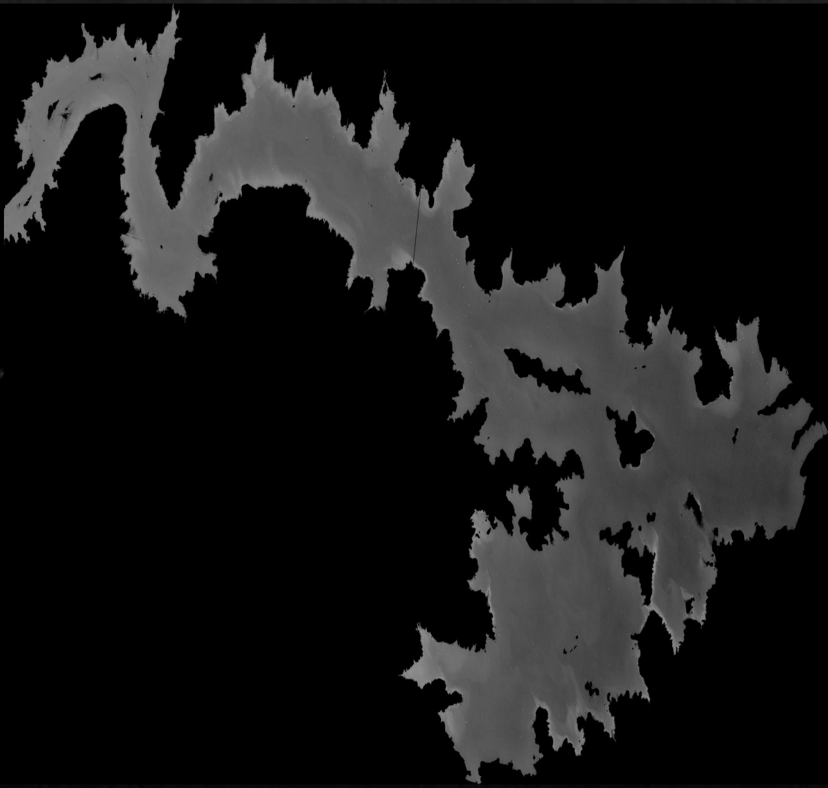
Almatti Dam for October



Almatti Dam for November

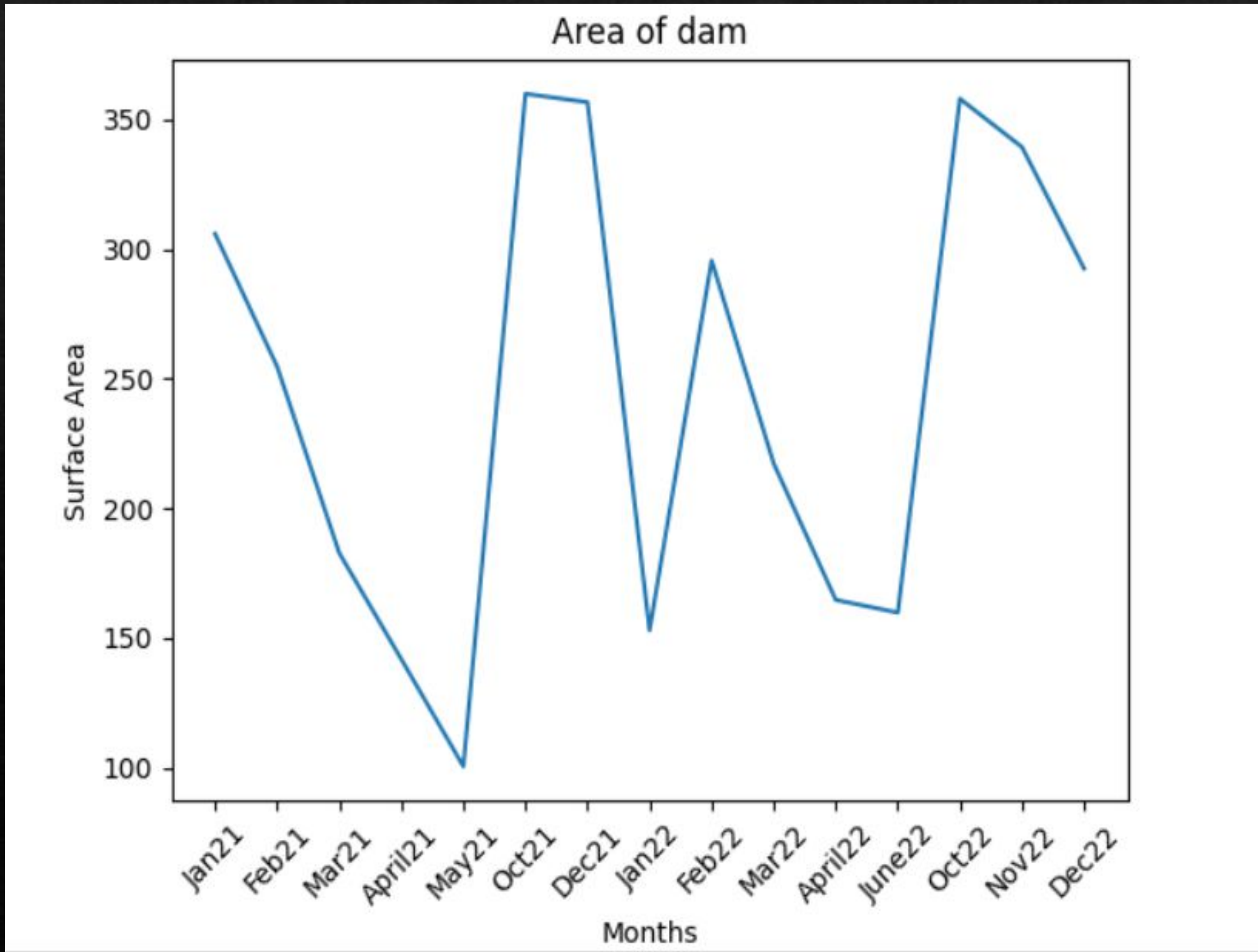


Almatti Dam for December



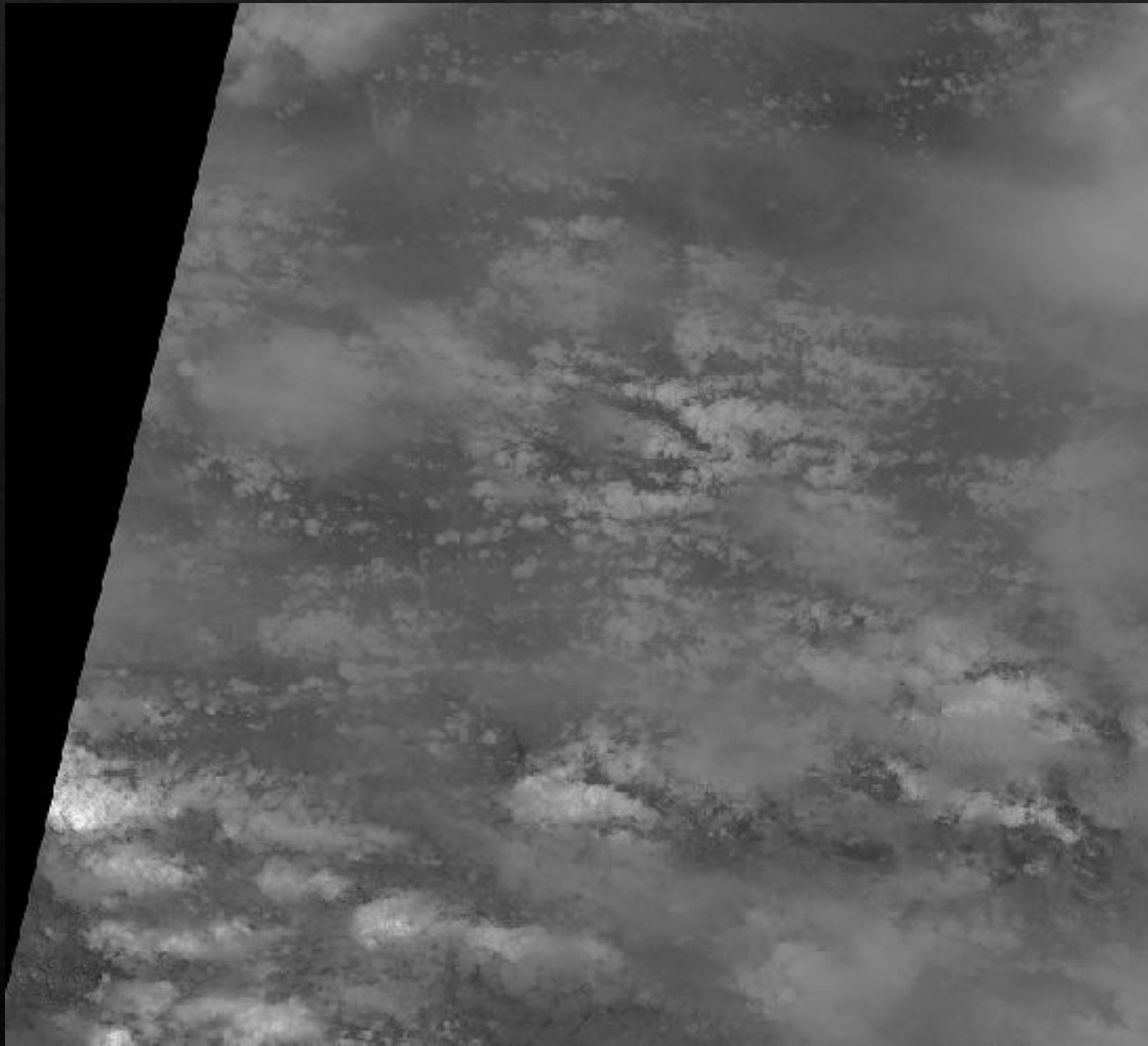
Calculating surface area of water bodies and plotting the data

- Calculated the surface area of every generated .tif image
- Plotted area vs month graph
- The images for the rest of the months are cloudy



Cloudy Images

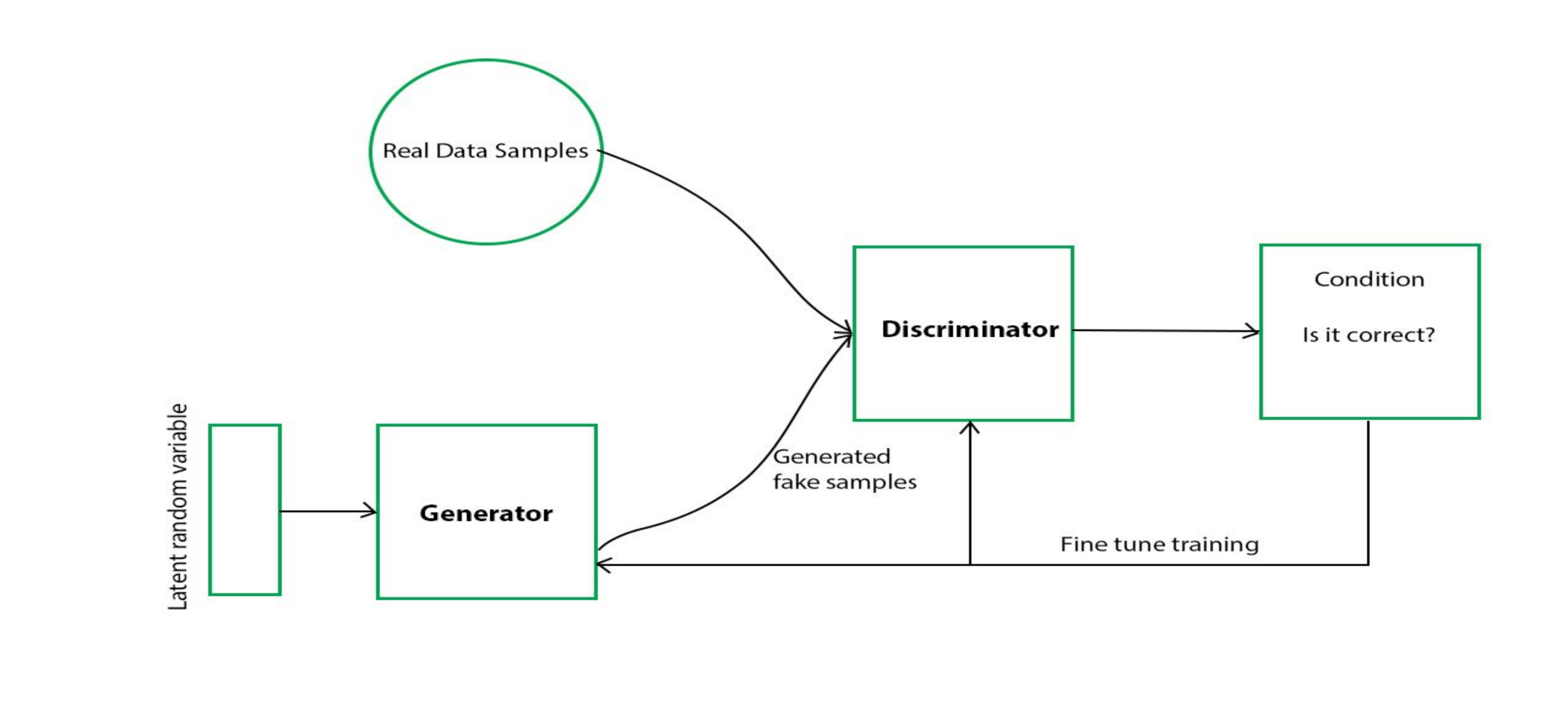
- Covers the desired area
- Makes it difficult to get any information of the required area
- Around 3-4 images in a year are covered with clouds
- Presence of clouds changes the pixel value which affects the calculation of NDWI



Cloud Removal Model

- Used GAN(Generative Adversarial Network), a type of machine learning model
- Two neural networks, a generator and a discriminator, compete with each other to produce new data that resembles a training dataset
- Generator takes the masked image as input to generate a new image with filled missing parts of the masked image
- Discriminator evaluates the generated image and gives feedback to the generator on how to improve the performance
- Generated cloud masks by setting pixel value to 0 in presence of clouds

Overview of GAN model



Cloud Masking

Before Cloud Masking

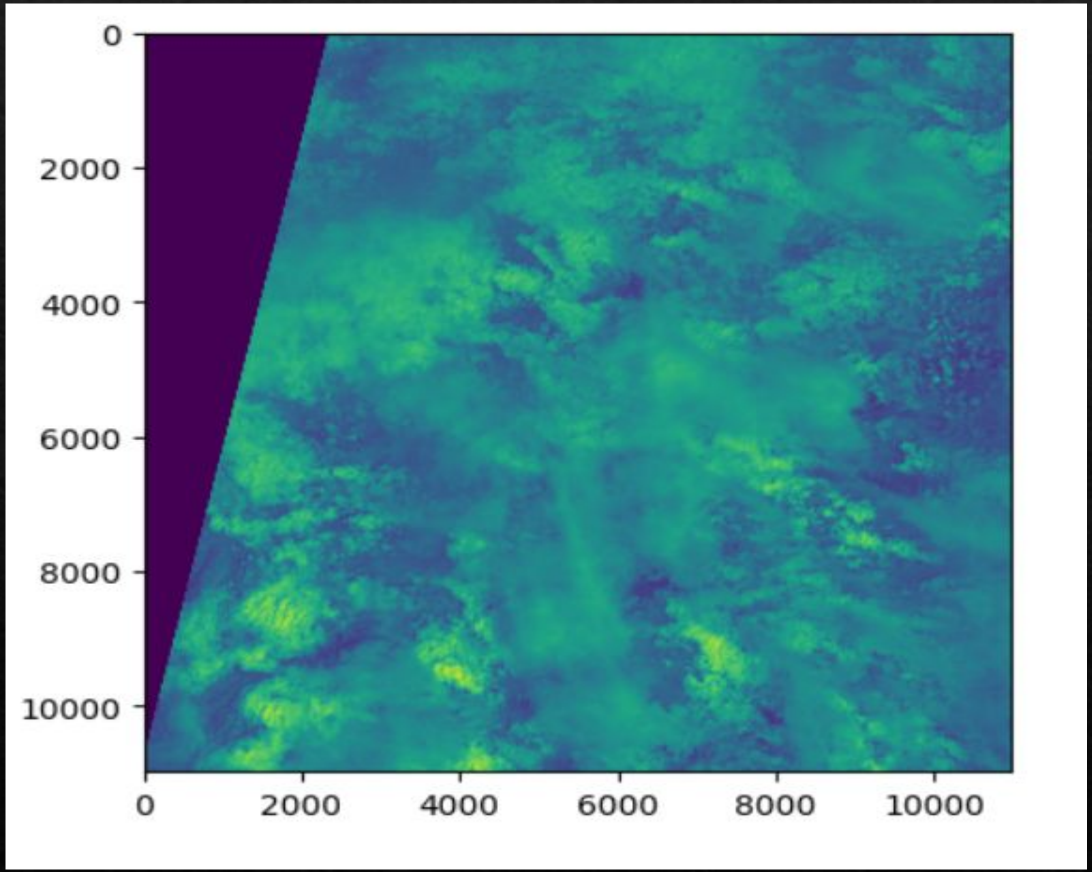
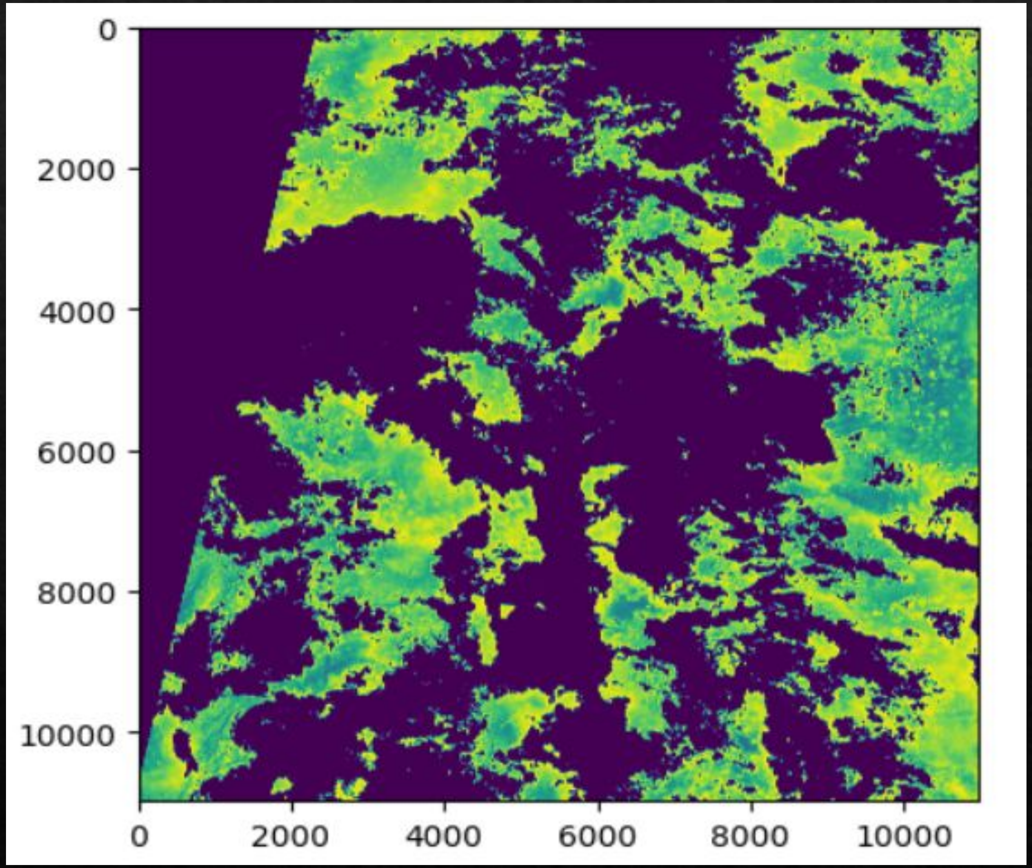


Image covered with clouds

After Cloud Masking



Blue - Removed cloudy part
Green - Part without clouds

Steps to build the model

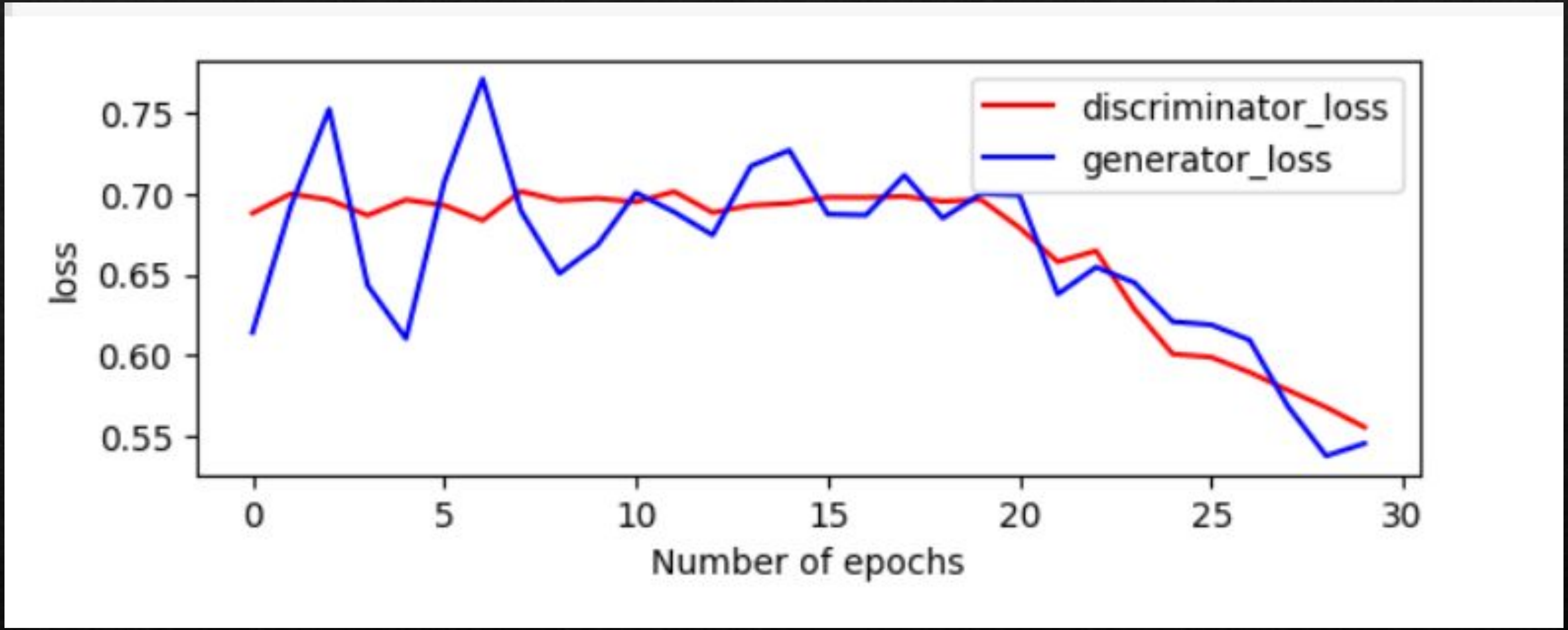
- Pre-processing of data : Resized and divided the image to smaller pixel values, normalised the pixel values
- Designed the discriminator model with two convolutional layers with 64 filters each, a single node in the output layer with the sigmoid activation function
- Designed the generator model with one input layer, 2 convolutional hidden layers with 128 nodes each and a single output layer
- Generated random data points as an input to the generator
- Fed the actual image data to the discriminator so that it can distinguish between real and generated data

Training of model

- Trained the GAN model by alternating the training between the generator and discriminator
- In each training iteration, the generator network generates some samples and the discriminator network tries to distinguish between the real samples and the generated ones
- Number of epochs : 20, batch-size : 8

```
>3, 9/600, d=0.684, g=0.712
4/4 [==============================] - 2s 380ms/step
>3, 10/600, d=0.689, g=0.710
4/4 [==============================] - 1s 195ms/step
>3, 11/600, d=0.673, g=0.691
4/4 [==============================] - 1s 195ms/step
>3, 12/600, d=0.687, g=0.683
4/4 [==============================] - 1s 196ms/step
>3, 13/600, d=0.679, g=0.670
4/4 [==============================] - 1s 326ms/step
>3, 14/600, d=0.693, g=0.668
4/4 [==============================] - 1s 341ms/step
>3, 15/600, d=0.694, g=0.664
4/4 [==============================] - 1s 318ms/step
>3, 16/600, d=0.688, g=0.668
4/4 [==============================] - 1s 192ms/step
>4, 1/600, d=0.692, g=0.670
4/4 [==============================] - 1s 191ms/step
>4, 2/600, d=0.699, g=0.673
4/4 [==============================] - 1s 209ms/step
>4, 3/600, d=0.702, g=0.678
4/4 [==============================] - 2s 365ms/step
>4, 4/600, d=0.703, g=0.677
4/4 [==============================] - 1s 191ms/step
>4, 5/600, d=0.702, g=0.685
4/4 [==============================] - 1s 207ms/step
>4, 6/600, d=0.702, g=0.673
```

Performance of the model



We can further increase the performance of the model by increasing the number of epochs

Analysis of the model

- Real accuracy of the discriminator decreases from 100% gradually as it learns accurately identify real data from the training set
- Fake accuracy of discriminator increases from 0% indicating the discriminator is not able to distinguish between real and fake data
- Discriminator loss decreases as the discriminator gets better at distinguishing real and generated image data
- Generator loss decreases as the generator gets better at generating images that can fool the discriminator by classifying as real

Conclusion & Future Work

THANK YOU