Cairo University-Faculty of Engineering
Computer Department
Satellite Imagery

AI-Powered Cloud Masking Satellite Imagery Project

Team 6

Full Name	ID	SEC	BN
اسامة صالح فرج السيد	9210195	1	12
عبدالرحمن محمد عبدالفتاح محمود	9210587	1	23
عبدالرحمن محمد حفني	9210584	1	22
عمرو صلاح الدين فؤاد	9210774	1	30

Submitted to:

Eng. Noran Hany

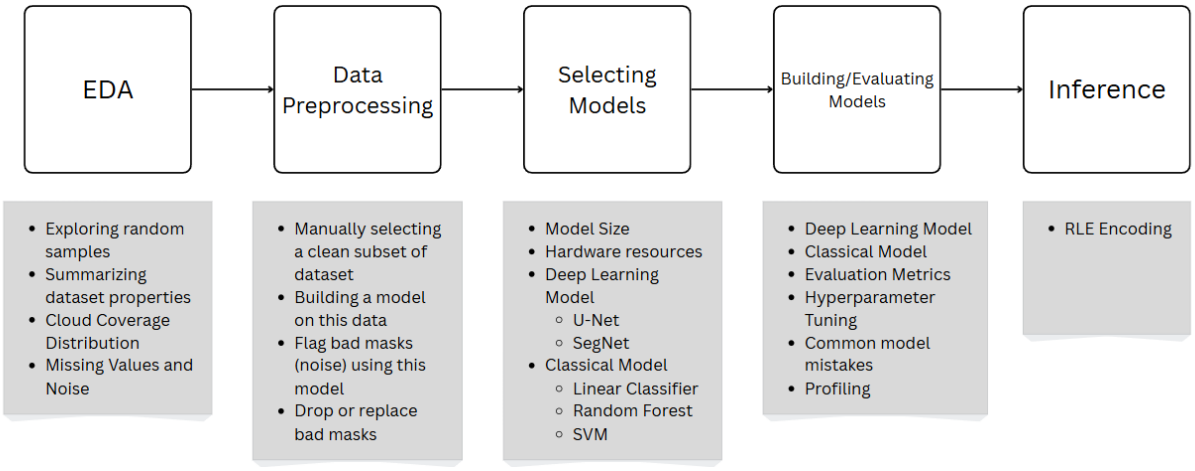
2025

Introduction

In this project, we implemented a complete machine learning pipeline that takes satellite images as input, identifies clouds, and generates a cloud mask as its output.

This helps in solving the cloud obstruction issue in optical satellite imagery, as identified clouds can be removed, ensuring clearer and more usable images for analysis.

Project Pipeline



Exploratory Data Analysis (EDA)

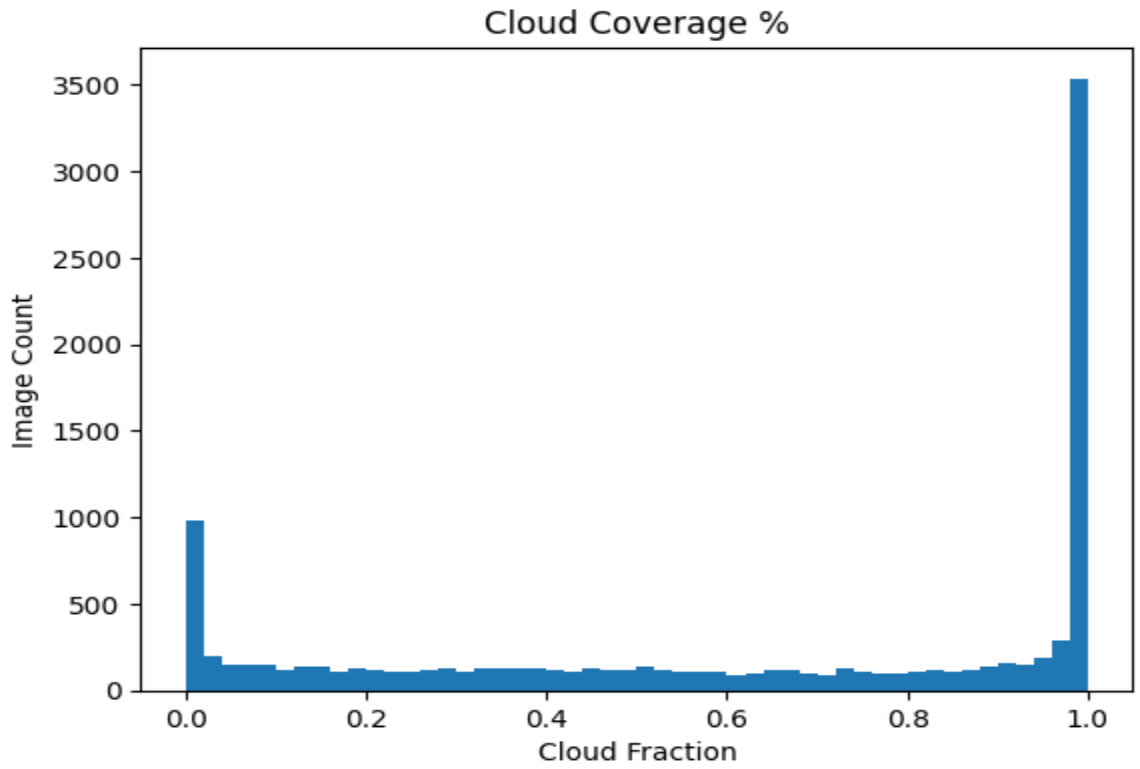
Dataset consists of nearly 10500 Satellite images with 4 bands, RGB and IR, each paired with a cloud mask with shape (512, 512).

After looking at random samples, we found that there are:

- **Mislabeled data**: mask is completely wrong or has noise.
- **Missing values**: mask is missing (completely black) although there are clouds in image.

Also, after calculating cloud coverage for each image, we found that:

- Dataset consists of 2730 **Fully clouded** images, 701 **Free-cloud** images, and 7142 **partially clouded** images. Meaning we need to make our model learn from each type and not focus on one type only.
- Positive class covers about 62% of dataset, so dataset doesn't suffer from imbalance classes



Data Preprocessing

Now, we need to handle noise and missing values in dataset, which we will do by following these steps:

1) Manually Selecting a clean subset of dataset

Reviewing a small subset of the dataset and choosing data that has correct masks.

2) Build a model on selected data

- Training a model on this small and clean subset of data till we get results that are good enough for the next step.
- We chose U-Net architecture for this task as it performs well with small data.

3) Flag data as clean or bad using this model

- Infer on the whole dataset using the model built in step 2, compare output with ground truth mask, and flag masks that have a matching percentage under some chosen threshold.
- Matching threshold was tuned till we flagged a reasonable amount of data.

4) Handle flagged data

Drop flagged data or replace bad masks with the model's output.

Model Selection and Training

1) Classical Model

While RFs (Random Forests) and Non-Linear SVM models are good with small datasets and can capture non-linear relations, their training is memory heavy and time consuming while we are limited to hardware resources.

So, we chose a linear model optimized with SGD (Stochastic Gradient Descent) from ***sklearn***, as it can do online learning using ***partial_fit()*** which makes training faster and memory efficient.

2) Deep Learning Model

- U-Net
  - More accurate
  - Higher memory usage
  - Slower training
  - Performs well with small datasets

- SegNet
  - Less accurate
  - Lower memory usage
  - Faster training
  - Can handle very large datasets

While both U-Net and SegNet architectures are both good for this task, we chose to use U-Net as it is more accurate, and dataset contains different types of complex clouds that SegNet can miss.

Performance Analysis

Dice coefficient was used as the main evaluation metric as its best capture overlap between predicted and ground-truth masks.

A validation set was used to apply early stopping to avoid overfitting, and to select the threshold that results in highest dice score.

Model performed badly on images where snow or sand covers a large area, likely due to insufficient representation in the training data and as they have close values in different channels and similar features.

UNet model achieved a dice score of 92.7% on test set, while SGD model achieved 71.4%.

Enhancements and Future Work

- Enhancing models' generalization.
- Trying better evaluation metrics, like combining Dice Coefficient and Jaccard Index
- Building more complex and highly accurate models after acquiring more or better hardware resources.
- Training on more data samples that contain snowy and sand areas