SOLAR FORECASTING USING SATELLITE IMAGES

Authored by

Kalpalathika.N

Mufaddal.Kapasi

Under the mentorship of

Mr.RangarajA.G, Deputy Director (Technical), NIWE

Mr.Srinath Yelchuri, Assistant Director, NIWE

National Institute of Wind Energy (NIWE), Chennai – 600100

Ministry of New and Renewable Energy (MNRE), Government of India

MAY 2021-APRIL 2022

ACKNOWLEDGEMENT

We would like to thank Mr.K.BOOPATHI, Director & Division Head, (R&D, RDAF & SRRA), NIWE, for giving us opportunity to carry out this internship.

This internship was carried out under the mentorship of Mr.A.G.Rangaraj, Deputy Director (Technical), R&D and RDAF Division, NIWE, Chennai. We thank him for his constant support and mentorship throughout the internship period, without which we couldn't have come this far.

We are thankful to Mr. Srinath for his guidance and doubt clearing sessions which helped a lot while working on this project.

We would also like to thank Smt Shivani Shah, senior scientist ISRO, for her continued support in helping us in downloading and handling the satellite images.

We would also like to thank other staff members and colleagues at NIWE, specially Sheela mam for helping us with various administrative procedures during the course of this internship..

Finally, we would acknowledge our family and friends for motivating us to carry out this research work.

Kalpalathika.N
Mufaddal.Kapasi
25-04-2022

DECLARATION

We, Kalpalathika.N and Mufaddal.Kapasi solemnly declare that the project report "***SOLAR FORECASTING USING SATELLITE IMAGES"*** is based on our own work carried out during the course of our internship under the supervision of Mr.A.G.Rangaraj.

Kalpalathika.N
Mufaddal.Kapasi
25-04-2022

TABLE OF CONTENTS

LIST OF FIGURES

LISTOF TABLES

Abstract

Solar energy is the most important renewable energy resource in terms of realization of future clean energy goals and climate change mitigation. However, like everything one of the major drawbacks in solar energy is the variability in the input power. Clouds can seriously reduce the incoming solar irradiance and is a serious risk for grid stability. This is where the need for solar forecasting arises. With a proper solar forecasting model grid operator can before-hand prepare for the power fluctuations due to clouds and hence make the solar grid more reliable by avoiding blackouts. At present India's utility scale solar capacity is 50Gw and keeping in mind India's commitment at the COP26 and high capital investments in solar energy sector this number is predicted to rise exponentially, with increasing solar energy penetration in India's grid, the need for solar forecasting is more than ever important. We aim to build a satellite to solar irradiance model using various sophisticated algorithms like block matching and optical flow. The model results are compared and evaluated with the ground data using the performance metrics.

1. Introduction

Solar forecasting can be done using full sky cameras as well as satellite images. While many researchers have developed forecasting models based on full sky images, the forecast horizons achieved using the full sky camera is very less only to a few minutes compared to the forecast horizon achieved by satellite based forecasting. State load dispatch centers (SLDCS) require atleast hourly forecast and hence full sky image based forecasting techniques however accurate is of little use for the SLDCS. The satellite used in forecasting is INSAT-3D which is India's geostationary satellites launched and maintained by the Indian space research organization (ISRO). Geostationary satellites are used in forecasting because they appear to be stationary over a point on earth since their orbital period matches the rotational period of the earth and hence continuous data over any point on earth lying inside the satellites field of view can be accumulated. In this work, we have used HDF5 format of the INSAT-3D satellite data and we followed the following steps to obtain the satellite to solar irradiance model. In the first step, we have processed the HDF5 data and obtained the image. In the second step, we used the image for the cloud detection process using K-Means clustering algorithm. In the third step, we obtained the clear and the cloudy pixel values for calculating the cloud index. In the fourth step, we obtained the normal GHI values using the simplified solis algorithm and we implemented the block matching and optical flow forecasting algorithm to obtain the forecasted GHI values. In the fifth step, we have compared the forecasted GHI values with the ground data using the performance metrics: Root Mean Square Error (RMSE) and Mean Absolute Error (MAE).

A. Objective

The main objective of this work is to formulate a best suited algorithm for satellite to solar forecasting and solar resource assessment. The attempt is to make an automated framework that generates an accurate solar forecast with a very minimum input from the users end.

B. Motivation

With the scaling up of solar projects in India, the fraction of solar power in the national grid will increase exponentially and hence make the grid more vulnerable to power fluctuations due to the variable nature of solar energy. There is an urgent need to develop robust algorithms that can

be used to successfully forecast solar irradiance which can be used by grid operators to modify their operations and hence make the solar grid stable.

C. Research Gap

- Mostly Numerical values are used to forecast the GHI using Deep Learning Methods.

- Few used satellite data for GHI estimation but Most of the authors prefer to use LANSAT, Sky camera and Meteosat data and estimated the GHI using python library Methods. Forecasting is not performed.

- Only one author has calculated GHI using INSAT-3D satellite images. No forecasting has been done using INSAT-3D.

D. Problem Statement:

Implemented an automated Model which handles the data download, cloud detection using ml algorithm, GHI estimation using python library and also the GHI forecasting using OPEN-CV algorithms.

E. Research Challenges

The academic resources and research papers on solar forecasting is very rich and highly diverse in their approaches. However satellite based forecasting using INSAT-3D satellite is performed very less, unlike LANSAT,MODIS etc and hence there is no community support for the data. This was the biggest difficulty we faced. Moreover the complexity of downloading the data from mosdac website is much entangled and many times the servers are not working. The way the HDF file is structured added to the complexities and this was the biggest hurdle we had to face during this project. The data is missing for many timestamps, as in our case the data was missing for many time stamps in the month of september. This makes the coding more complex and further difficult to execute at run time especially when we are doing time series based forecasting. The satellite data has to undergo certain corrections like the cosine correction, elevation correction and satellite degradation correction before the irradiance forecast can be carried out. None of us were from a remote sensing background and hence it was very difficult for us to go in the details of remote sensing and use the different python libraries for remote sensing. If there was not such a vast open source python community, this project wouldn’t have been possible in the first place.

2. Literature Survey

S.No	Title	Year	Techniques
1	W. Xing, G. Zhang, and S. Poslad, "Estimation of global horizontal irradiance in China using a deep learning method," International Journal of Remote Sensing, vol. 42, no. 10, pp. 3899–3917, May 2021, doi: 10.1080/01431161.2021.1887539	2021	**Deep Belief Network** ● GHI is estimated Using DBN and the results are compared with the ground values by measuring Root Mean Square Error (RMSE) and Pearson correlation coefficient (r). ● The authors have considered No cloud and Cloud conditions for both summer and winter seasons for evaluating the DBN model.
2	R. A. Rajagukguk, R. Kamil, and H.-J. Lee, "A Deep Learning Model to Forecast Solar Irradiance Using a Sky Camera," Applied Sciences, vol. 11, no. 11, p. 5049, May 2021, doi: 10.3390/app11115049	2021	**LSTM Model - forecast the cloud cover** **Physical Solar Radiation Model to estimate the GHI** ● Cloud cover data were collected by image processing of sky images and using the LSTM model 10 min ahead cloud cover forecast is done. to forecast cloud cover. ● The forecasted cloud cover data were plugged into solar radiation models as input in order to predict global horizontal irradiance.

3	D. Pattanaik, S. Mishra, G. P. Khuntia, R. Dash, and S. C. Swain, "An innovative learning approach for solar power forecasting using genetic algorithm and artificial neural network," Open Engineering, vol. 10, no. 1, pp. 630–641, Jul. 2020, doi: 10.1515/eng-2020-0073	2020	**Genetic Algorithm and ANN Model** ● The author has compared the genetic algorithm based ANN model with the Standard statistical approaches such as F test, ANNOVA for forecasting the solar power (AC power). ● Based on the comparison it is proved that GA with ANN approach has given better results and reduced RMSE score. ● They have considered the two intermittent factors solar isolation and temperature.
4	S. Mahajan and B. Fataniya, "Cloud detection methodologies: variants and development—a review," Complex Intell. Syst., vol. 6, no. 2, pp. 251–261, Jul. 2020, doi: 10.1007/s40747-019-00128-0	2020	**Classical and Machine Learning Approaches** ● The author has discussed and compared various classical and machine learning based approaches for cloud detection from the year 2004 till 2018. ● They have also considered various forms of cloud such as cloud/no cloud, thin/thick cloud, snow/ice detection, and cloud/cloud shadow.

5	V. Kallio-Myers, A. Riihelä, P. Lahtinen, and A. Lindfors, “Global horizontal irradiance forecast for Finland based on geostationary weather satellite data,” Solar Energy, vol. 198, pp. 68–80, Mar. 2020, doi: 10.1016/j.solener.2020.01.008	2020	**Clear Sky Model :McClear, SPECMAGIC, Pvlib Ineichen, Pvlib Simplified Solis** **All-Sky Estimate Model** **Forecast Model for GHI : Solis Heliosat, Heliosat-4** ● The author has compared all the three models and validated their results in terms of Root Mean Square Error and Mean Bias Error. ● Additionally, two different datasets are taken into consideration and compared for changing cloud conditions.
6	B. Benamrou, M. Ouardouz, I. Allaouzi, and M. Ben Ahmed, “A Proposed Model to Forecast Hourly Global Solar Irradiation Based on Satellite Derived Data, Deep Learning and Machine Learning Approaches,” J. Ecol. Eng., vol. 21, no. 4, pp. 26–38, May 2020, doi: 10.12911/22998993/119795	2020	**RFE-CV with XGB, SVR, RF and LSTM** ● The author has evaluated LSTM model in different scenarios in terms of RMSE, MAE and R2 metrics and identified best suited scenario for accurate prediction of GHI. The best scenario is chosen using Grid Search Algorithm. ● They used two different data 1. past data and 2. surrounding pixel data

7	P. A. G. M. Amarasinghe, N. S. Abeygunawardana, T. N. Jayasekara, E. A. J. P. Edirisinghe, S. K. Abeygunawardane, and Department of Electrical Engineering, University of Moratuwa, Sri Lanka, “Ensemble models for solar power forecasting—a weather classification approach,” AIMS Energy, vol. 8, no. 2, pp. 252–271, 2020, doi: 10.3934/energy.2020.2.252	2020	**Weather Classification Approach, Ensemble- DBN, SVM, RF** ● The author has proposed combination of weather classification approach and ensemble learning technique to provide more accurate solar power generation and compared the results with single ML algorithms in terms of RMSE. ● They have implemented different approaches and identified the best combination model for accurate forecast.
8	E. T. Velasco and I. B. Salbidegoitia, “New methodology for solar irradiance calculation using Meteosat satellite imagery,” Casablanca, Morocco, 2019, p. 190016. doi: 10.1063/1.5117713	2019	**Clear Sky Radiation Calculation, Cloudiness Index, Meteosat** ● Solar resource assessment has been performed by the author using a web platform known as Meteosat. ● This platform helps to identify the viability of the location.

9	J. H. Jeppesen, R. H. Jacobsen, F. Inceoglu, and T. S. Toftegaard, "A cloud detection algorithm for satellite imagery based on deep learning," Remote Sensing of Environment, vol. 229, pp. 247–259, Aug. 2019, doi: 10.1016/j.rse.2019.03.039	2019	**U-NET, RS-NET, FMASK** ● A deep Learning based method is used to detect the cloud in satellite imagery.
10	B. Ameen, H. Balzter, C. Jarvis, and J. Wheeler, "Modelling Hourly Global Horizontal Irradiance from Satellite-Derived Datasets and Climate Variables as New Inputs with Artificial Neural Networks," Energies, vol. 12, no. 1, p. 148, Jan. 2019, doi: 10.3390/en12010148	2019	**Artificial Neural Network with the Levenberg–Marquardt** ● The author has obtained hourly GHI from the satellite data using ANN with L-M algorithm and evaluated the model results with RMSE, R2 and bias values.

11	S. Mohajerani, T. A. Krammer, and P. Saeedi, “Cloud Detection Algorithm for Remote Sensing Images Using Fully Convolutional Neural Networks,” arXiv:1810.05782 [cs], Oct. 2018, Accessed: Jul. 29, 2021. [Online]. Available: http://arxiv.org/abs/1810.05782	2018	**Fully Convolutional Neural Networks** ● A threshold and deep learning based hybrid method has been proposed by the author in order to identify the regions such as snow/icy and also to detect cloud pixels using Landsat 8 images.
12	B. Sivaneasan, C. Y. Yu, and K. P. Goh, “Solar Forecasting using ANN with Fuzzy Logic Pre-processing,” Energy Procedia, vol. 143, pp. 727–732, Dec. 2017, doi: 10.1016/j.egypro.2017.12.753	2017	**Feed Forward Neural Network with Back Propagation, Fuzzy Logic** ● Solar forecasting is done using ANN based model along with fuzzy Logic pre-processing. ● The author has also included an error correction factor in order to improve the accuracy

3. Description of the Dataset

- The INSAT-3D Satellite data for the asia region is downloaded from the mosdac website in the HDF5 and Geo-Tiff file format.
- The INSAT-3D satellite data is received at the frequency of 30 mins.
- The ground data for each area of interest was received from the ground station maintained by NIWE.
- The ground data is received at a frequency of 1 minute and hence to match the ground data with the satellite data the ground data frequency is averaged to 30 minutes.

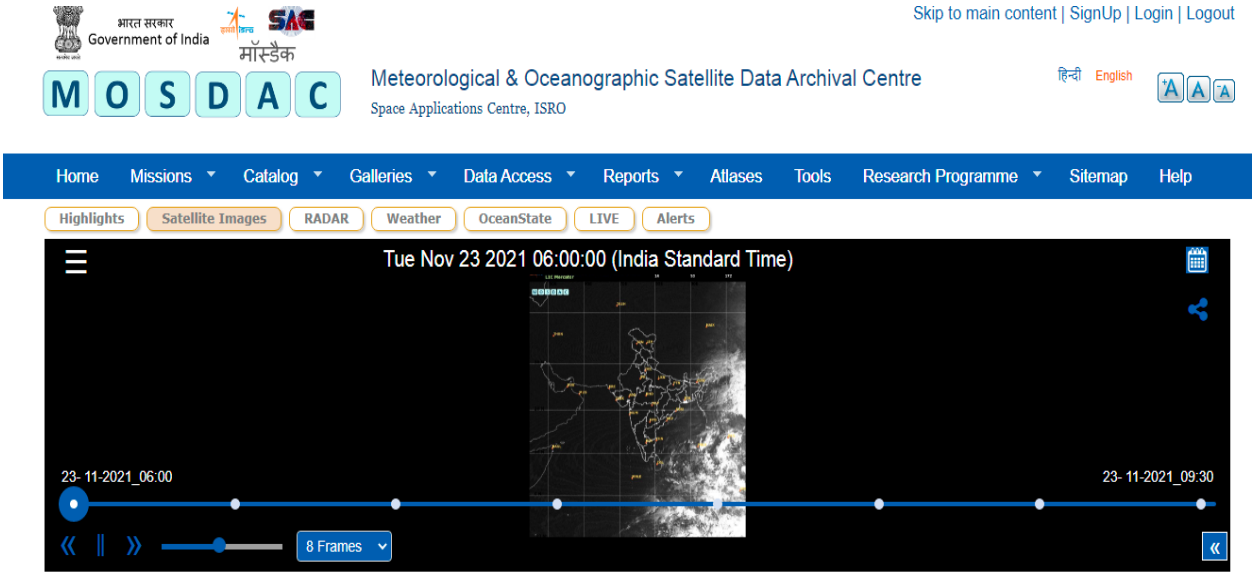


Figure 1. MOSDAC Website

INSAT 3D

- INSAT-3D is a meteorological satellite of ISRO.
- INSAT-3D is an advanced weather satellite with improved Imaging System and Atmospheric Sounder.
- INSAT-3D is designed for enhanced meteorological observations, monitoring of land and ocean surfaces, generating vertical profile of the atmosphere in terms of temperature and humidity for weather forecasting and disaster warning.
- It carries four payloads -
    - 6 channel multi-spectral Imager
    - 19 channel Sounder
    - Data Relay Transponder (DRT)

- Search and Rescue Transponder

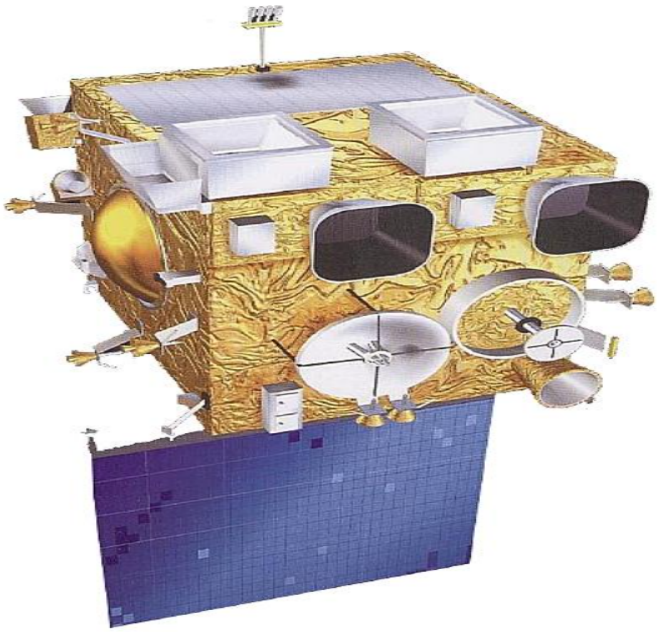


Figure 2. Illustration of the deployed INSAT-3D spacecraft (image credit: ISRO)

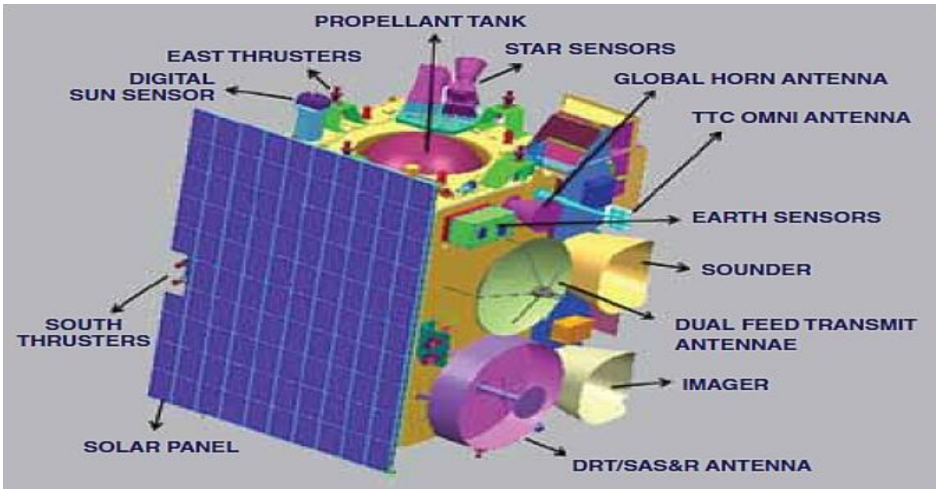


Figure 3 INSAT 3D Configuration Details (image credit: ISRO)

Number of bands (6): 0.52 - 0.72 µm, VIS (Visible) 1.55 - 1.70 µm, SWIR (Short Wave Infrared) 3.80 - 4.00 µm, MWIR (Mid Wave Infrared) 6.50 - 7.00 µm, WV (Water Vapor) 10.2 - 11.2 µm, TIR-1 (Thermal Infrared) 11.5 - 12.5 µm, TIR-2 (Thermal Infrared).

Spatial resolution: 1 km for VIS and SWIR, 4 km for MWIR, 8 km for WV, 4 km for TIR-1 and TIR-2.

For satellite to irradiance forecasting only the visible band is required, thermal infrared bands can be used for early morning forecasting but we have not pertained our research to early morning forecasting and hence we have only used the visible band data for this project.

HDF5 File Format

- The Hierarchical Data Format version 5 (HDF5) is an open-source file format that supports large, complex, heterogeneous data.
- HDF5 uses a "file directory" like structure that allows you to organize data within the file in many different structured ways, as you might do with files on your computer.
- The HDF5 format also allows for embedding of metadata making it self-describing.
- To access HDF5 data files, a python library called h5py is used.
- Visible data is considered for the process.
- Azimuth and Elevation are measures used to identify the position of a satellite flying overhead.
- Sun elevation is the elevation of the sun and is an important data in the satellite. Since the geostationary satellite is high in the orbit it has a large field of view and therefore each pixel will have a corresponding sun elevation.
- Azimuth tells you what direction to face and Elevation tells you how high up in the sky to look. Both are measured in degrees.
- As mentioned earlier before the irradiance forecasting there are certain corrections that have to be carried out and one of the correction is the cosine correction of the pixels.
- In INSAT-3D sun elevation angle data is given and this data can be converted into a numpy array. Then this elevation angle is converted into the solar zenith angle which is done by subtracting it with 90 degrees.
- This gives us the solar zenith angle θz and then in the final step we calculate the cosine of the zenith angle.
- This cosine of the zenith angle is used for cosine correction of each pixel before irradiance assessment and forecasting can be performed.

Attribute	Value
Acquisition_Date	01MAY2021
Acquisition_End_Time	01-MAY-2021T09:26:53
Acquisition_Start_Time	01-MAY-2021T09:00:03
Acquisition_Time_in_GMT	0900
Datum	WGS84
Ellipsoid	WGS84
Ground_Station	BES,SAC/ISRO,Ahmedabad,INDIA.
HDF_Product_File_Name	3DIMG_01MAY2021_0900_L1C_ASIA_MER
Nominal_Altitude(km)	36000.0
Nominal_Central_Point_Coordinates(degrees)_Latitude_Longitude	0.0,82.0
Output_Format	hdf5-1.8.8
Processing_Level	L1C
Product_Creation_Time	2021-05-01T15:05:36
Product_Type	SECTOR
Satellite_Name	INSAT-3D
Sensor_Id	IMG
Sensor_Name	IMAGER
Software_Version	1.0
Station_Id	BES
Unique_Id	3DIMG_01MAY2021_0900_ASIA_MER
conventions	CF-1.6
institute	BES,SAC/ISRO,Ahmedabad,INDIA.
left_longitude	44.5
lower_latitude	-10.0
right_longitude	105.5
source	BES,SAC/ISRO,Ahmedabad,INDIA.
title	3DIMG_01MAY2021_0900_ASIA_MER_L1C
upper_latitude	45.5

Figure 4 Sample HDF5 Metadata

Name	Data
GreyCount	[0 1 2 ... 1021 1022 1023]
IMG_VIS	[[[66 74 73 ... 85 84 85] [78 75 68 ... 85 87 87] [91 75 78 ... 85 86 86] ... [38 66 50 ... 48 49 48] [46 53 45 ... 50 49 46] [45 121 47 ... 52 50 45]]]
IMG_VIS_ALBEDO	[0. 0.201304 0.402608 ... 100. 100. 100.]
IMG_VIS_RADIANCE	[0. 0. 0. ... 52.4 52.4 52.4]
Projection_Information	[0]
Sat_Azimuth	[[[13371 13375 13379 ... 21202 21207 21211] [13370 13373 13378 ... 21204 21208 21213] [13370 13373 13378 ... 21205 21209 21214] ... [7845 7843 7841 ... 29269 29266 29262] [7840 7838 7837 ... 29276 29273 29270] [7836 7834 7832 ... 29284 29281 29277]]]
Sat_Elevation	[[[2693 2695 2696 ... 3380 3379 3377] [2695 2697 2700 ... 3382 3381 3380] [2698 2699 2701 ... 3386 3384 3383] ... [4613 4617 4622 ... 6165 6161 6157] [4611 4616 4621 ... 6163 6159 6156] [4611 4615 4620 ... 6162 6158 6155]]]

Figure 5 Sample HDF5 Data

Geo-Tiff File Format

- Geo-Tiff is a common domain metadata standard that provides geo-referencing information to be embedded within an image file.
- The metadata includes details related to satellite imagery, aerial photography and digitized maps that are used in GIS applications.

- Clipping the area of interest data from the whole satellite data is one among the important step to be carried out. To do that we need to identify the CRS of the data.
- Since few parameters are not available in the HDF file format. We have downloaded a sample tiff file from the mosdac website and then the visible data array obtained from the HDF5 file was converted into a geotiff file with the sample file parameters for clipping the data for the area of interest.
- Therefore, it is advisable to download the data in hdf format only and then any band data in the form of a numpy array can be converted into a geotiff file using the gdal or rasterio library.

{'count': 1,
'crs': CRS.from_wkt('PROJCS["Mercator",GEOGCS["WGS
84",DATUM["WGS_1984",SPHEROID["WGS84",6378137,298.257223563,AUT
HORITY["EPSG","7030"]],AUTHORITY["EPSG","6326"]],PRIMEM["Greenwic
h",0],UNIT["degree",0.0174532925199433,AUTHORITY["EPSG","9122"]],AUT
HORITY["EPSG","4326"]],PROJECTION["Mercator_2SP"],PARAMETER["stand
ard_parallel_1",17.75],PARAMETER["central_meridian",75],PARAMETER["false
_easting",0],PARAMETER["false_northing",0],UNIT["metre",1,AUTHORITY["E
PSG","9001"]],AXIS["Easting",EAST],AXIS["Northing",NORTH]]'),
'driver': 'GTiff',
'dtype': 'uint16',
'height': 1616,
'nodata': 0.0,
'transform': Affine(3998.2979035067983, 0.0, -3234623.003937, 0.0,
-3997.9876904022276, 5401854.420193),
'width': 1618}

Figure 6Sample Geo-Tiff Metadata


```
<xarray.DataArray (y: 1616, x: 1618)>
[2614688 values with dtype=float32]
Coordinates:
    band         int64 1
  * x            (x) float64 -3.233e+06 -3.229e+06 ... 3.229e+06 3.233e+06
  * y            (y) float64 5.4e+06 5.396e+06 ... -1.053e+06 -1.057e+06
    spatial_ref  int64 0
Attributes:
    scale_factor:  1.0
    add_offset:    0.0
```


Figure 7 Sample Geo-Tiff Data

Coordinate Reference System (CRS)

- A coordinate reference system (CRS) is a coordinate-based local, regional or global system used to locate geographical entities.

- A CRS defines the translation between a location on the round earth and that same location, on a flattened, 2-dimensional coordinate system
- There are many different CRS systems but the most popular and the widely used one is the web mercator that has the code EPSG:3857. This is the most common CRS system and represents a 3 dimensional earth on a 2 dimensional plain.
- The second most common CRS system used is the EPSG:4326.EPSG:4326 - WGS 84, latitude/longitude coordinate system based on the Earth's center of mass, used by the Global Positioning System among others .

- In this model we have used the INSAT-3D ASIA MERCATOR product which has the CRS EPSG:3857 web mercator.

- For clipping the data for the area of interest from the entire data we have to use shape files which can be created on google earth.

- The CRS of these shape files is EPSG:4326 and hence before the clipping function we have to make sure that the CRS of both the files are same and this can be easily done using rasterio.

Area of Interest and Clipping

The model developed by us was implemented over three places in INDIA and these places were:

1) Kutch, Gujarat

2) Jaisalmer, Rajasthan

3) Tirupati, Andhra Pradesh

- The reason for choosing these locations are as follows:
- They cover western, northern and southern parts of India and hence we have diversity in the data.
- Since the model depends upon many atmospheric properties like the Aerosol optical depth and turbidity which varies from region to region and different climate zones, the model is verified in each climatic zone of peninsular India namely tropical, arid and semi-arid.

- Data has shown that these locations have high GHI values throughout the year and has functioning solar parks, moreover there are many more solar projects that are in pipeline to be built in these places.
- The data from the visible band contains pixels that represents the entire area of India and eastern asia which is the satellites entire field of view however we only require the pixels representing our area of interests since this will not only save computational power but also give us an area specific forecasting and hence, we need to clip the pixels representing the area of interest from all the pixels and this is carried o.ut with the help of shape files.
- Shape files for any region can be created on google earth and these shape files can be used to clip the pixels representing the AOI with the help of rasterio.clipped() function in the rasterio module.
- In the INSAT-3D asia mercator product the CRS is not given in the metadata. If the CRS of a satellite data is missing from the metadata then the CRS can be easily written using the function rasterio.writeCRS() in the rasterio module, however for reasons not known this function does not work with the INSAT-3D hdf data and hence for clipping the area of interest the visible array is converted into a tiff file and this file is then used to clip the pixels representing the area of interest

4. Methodology and Algorithm Used

A. Experimental Setup

In this work, we have used Google Drive as our cloud storage and Google COLAB as our platform for performing our experiments. A windows 10/MAC OS 10.11/Linux operating system with x86 64-bit processor, 8 GB of RAM and 20 GB of disk space are minimum required to support our tools for executing our experiments

B. Flow Chart

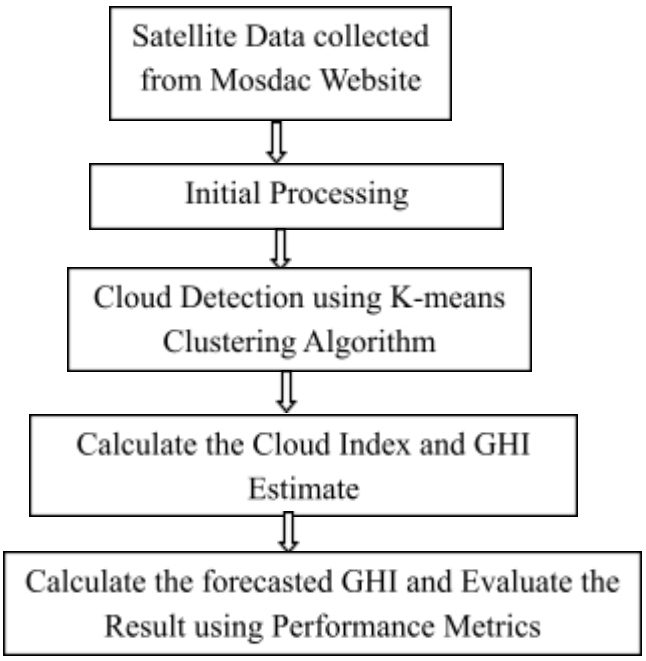
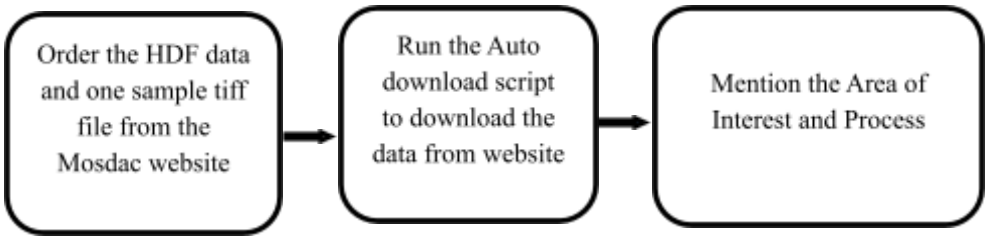


Figure 8 Flow Chart of the Model

The Model flow is as follows:

1. Download the data from the mosdac server.
2. Convert the HDF5 files into images using the Initial Processing Module.
3. Calculate the Cloud Mask values using the K-Means Clustering Algorithm using Cloud Detection Module.
4. Calculate the cloud Index and GHI Using the GHI Calculation Module.
5. Forecast the GHI values using the Block Matching and Optical Flow Algorithms using Forecast GHI calculation Module.
6. Evaluate the Forecasted GHI values with the ground data values using the Performance metrics.

C. Overall Architecture Diagram



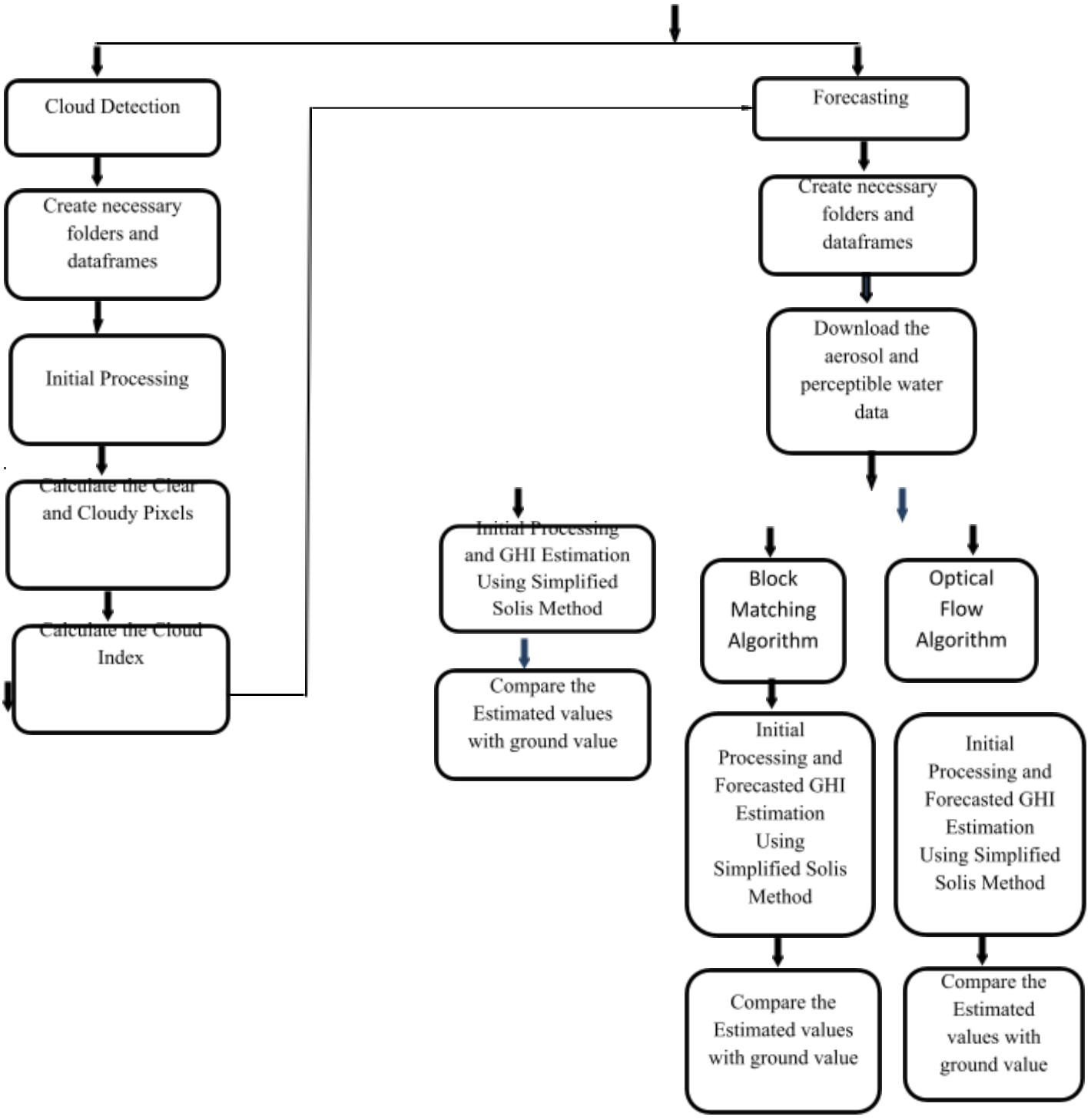


Figure 9 Overall Architecture of the Model

The entire model can be majorly divided into two sections:

1. Cloud Detection
2. Forecasting

Cloud Detection:

- Section 1 involves calculation of the cloud and ground albedo aka lower and upper bounds of the satellite dynamic range for each area of interest.

- This a very important part since to perform the actual forecasting cloud index is required and for calculating the cloud index the surface albedo and cloud albedo is required.
- The value of the cloud albedo will always be greater since clouds have a high reflectivity and hence higher count is recorded over clouds, therefore the cloud albedo is called the Nmax.
- The value of the surface albedo will always be lower since the reflectivity of surface is lower as compared to the clouds reflectivity and hence low counts are recorded over land and therefore it will be called as Nmin.
- If the ground is covered by snow, then the ground reflectivity matches that of cloud reflectivity since snow is highly reflective and clouds cannot be detected, in this case the thermal infrared channels are used and their difference is used to detect clouds and hence calculate the ground albedo. Since all our area of interests lies in non-snowy regions this aspect is not incorporated into the model.
- 2 months of data is used to calculate the Nmax and Nmin, this is done because increasing the data frequency reduces the chances of the error metrics. Since we take the 98th percentile of all the cloudy pixels over the region for a span of 2 months any discrepancies in the data will be diluted and we get the true cloud albedo(Nmax).
- The same goes ground albedo, since we take the 5th percentile of all the clear pixels for a span of 2 months any varied data due to surface changes will included and we get the true ground albedo (Nmin).

Forecasting

- Section 2 is where the actual forecasting is carried out.
- The values of the Nmax and the Nmin are used to calculate the cloud indexes.
- Cloud indexes of 2 consecutive timestamps are feed in into the BM or the optical flow algorithm which gives the forecasted cloud index.
- Once the cloud index is forecasted calculating the forecasting GHI can be done very easily

D. Modules

Module 1 – Initial Processing

- The raw data from the HDF5 file and Geo-Tiff file consists of the data in the pixel format.
- 2 Months of data for each area of interest is downloaded to calculate the Nmax and Nmin values of that region. The hdf files of only the visible channel is downloaded.

- Implemented an automated script using pysftp for automatically downloading the data from website to local directory.
- The data in HDF5 file format consists of visible counts, sun azimuth, time, solar azimuth, sun elevation, solar elevation, projection information including the meta data.
- The hdf5 file is taken as input for the Image processing.
- Visible counts are extracted from the file and converted into a numpy array.As mentioned earlier the sun elevation data is extracted and is converted into sun zenith angle by subtracting it with 90 degrees.
- The incoming satellite visible count (nraw) is initially cosine corrected to account for the first order solar geometry effect as

$$n = (nraw * R)/cos(\Theta z) \text{---------- (1)}$$

Where z is the solar zenith angle and R is a function accounting variation in Sun-Earth distance.

- The cosine corrected visible count is then converted to visible radiance using the satellite sensor coefficient. This is done to take account for the satellite sensor degradation effect.
- The cosine corrected visible radiance numpy array is then converted into an image using plt.imshow () function of the matplotlib module.

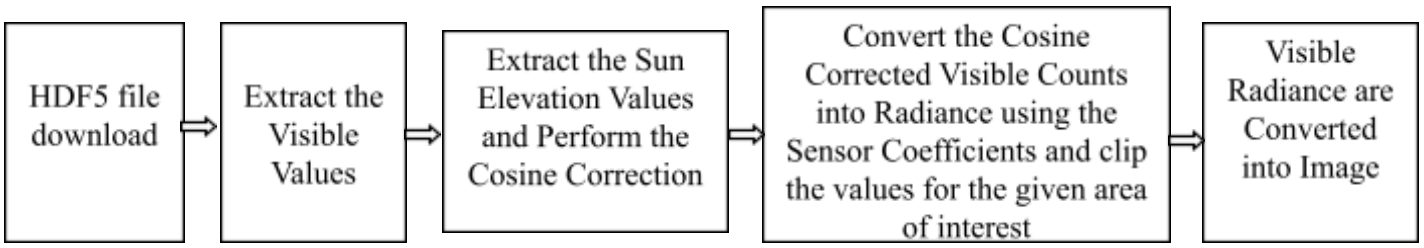


Figure 10 Flow of the Module 1

Module 2 – Cloud Detection

- Cloud Detection is a process which detects clear and cloudy pixels for calculating Nmax and Nmin
- This is one of the most important step in developing a successful model, Since the forecast accuracy depends upon the accuracy of the forecasted cloud index.

- The Cloud detection is implemented using K-Means unsupervised clustering approach with the adaptive threshold mean calculation for identifying the cloud mask values for the visible channel data.
- The mask values are the final output obtained from the cloud detection algorithm.
- The mask values are separated into clear and cloudy pixels.
- The clear and cloudy pixels values of the area of interest will be clipped, which will give the Nmax and Nmin which will be the cloud and ground albedo of each of our regions respectively.
- The Nmax and the Nmin will be used to calculate cloud index for GHI calculations.
- It should be noted that the value of the Nmax and Nmin will differ from region to region.
- The variation in the value of Nmax for different regions will be not that much, the reason being the type and the thickness of clouds region to region over small distances don't vary that much.
- The value of Nmax can also be considered for a long time and need not to be re calculated since the cloud albedo remain the same over a region unless there is a major climatic anomaly.
- The value of Nmin on the other hand will be highly varied since each region can have different surface properties. Deserts have a very high albedo while trees and plants have very less albedo therefore the value of Nmin will be very high for Jaisalmer while very low for Tirupati.
- The value of Nmin can change with time and hence it is required that it should be recalculated after some period of time, this is because with time the surface can change and hence this will cause the albedo to be changed, for example if trees grow in a before barren area then the value of Nmin will change drastically.

K-Means Algorithm

- K-means clustering uses vector quantization Methodology for clustering the data.
- It segregates n observations into k clusters in which each observation belongs to the cluster with the nearest mean.
- Given an initial set of k means. The algorithm follows two steps:
    - Assignment step: Assign each observation to the cluster with the nearest mean.

- Update step: Recalculate means (centroids) for observations assigned to each cluster.

- The algorithm converges when the assignments no longer changes.

Algorithm

- Convert the input image into image array
- Assign the cluster value and criteria
- Calculate the centroid using the Adaptive threshold technique.
- The centroid of each of the k clusters becomes the new mean.
- Steps 2 and 3 are repeated until convergence has been reached.
- Obtain the mask values.

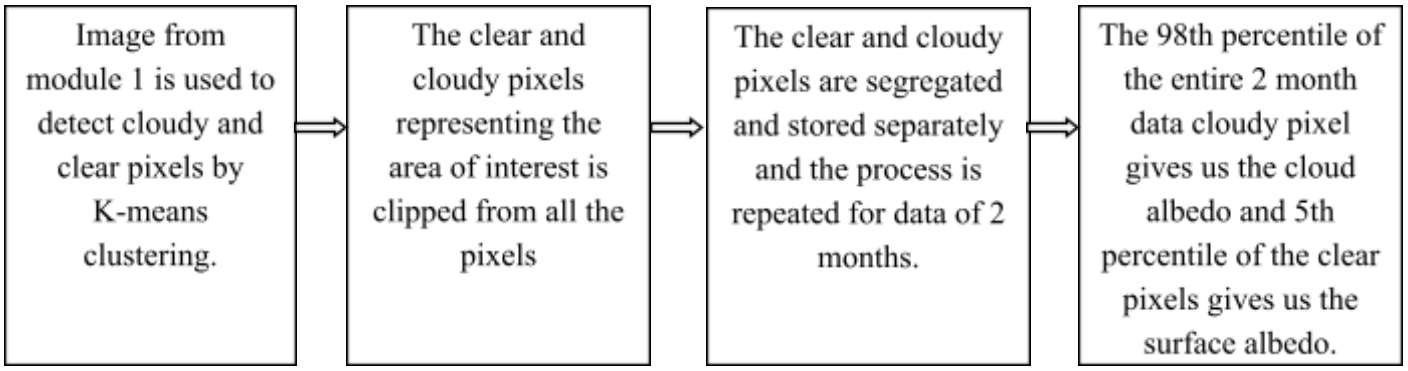


Figure 11 Flow of the Module 2

Module 3- Calculation of the GHI

- The Overall GHI calculation process used to build our model is known as HELIOSAT Method.
- The following are the steps involved in HELIOSAT method:
  1. Calculation of the cloud index
  2. From the cloud index the sky clearness index is calculated

3. Clear sky ghi is calculated for the given timestamp of which the cloud index is calculated.
4. The sky clearness index is then multiplied by the clear sky GHI to calculate the actual ghi.

Calculation of the Cloud Index

- The cloud index is calculated by using the Nmin and Nmax values.
- The formula for calculating the cloud index CI is:

$$CI = (N - Nmin)/(Nmax - Nmin)$$ ---------- (2)

Where Nmin is the ground albedo; Nmax is the cloud albedo; N is the cosine corrected visible radiance.

Calculation of the Clear Sky Index

- The clear sky index (Kc) is calculated from the cloud index (CI).
- The empirical relation between Kc and CI is of the form similar to the one originally derived in the HELIOSAT method derived by perez

$$Kc = a + b * (1 - CI)$$)---------- (3)

Where Kc is the clear sky index and CI is the cloud index; a and b are the constants from linear regression at 3 climate classification in India (arid, tropical and temperate).

- The values of a are 0.04, 0.03 and 0.03 for arid, tropical and temperate climates, respectively while the values of b are 0.96, 0.97 and 0.97.
- It is worthwhile to be noted that the value of the clear sky index (Kc) should be between 1 and 0 where 1 represents clear sky and 0 represents cloudy sky.
- If the value of Kc is less than 0 then this indicates that the value of visible radiance was more than the cloud albedo.
- If the value of Kc is more than 1 than this indicates that the value of the visible radiance was less than Nmin.
- Any of the above situations can be possible, during the model implementation we too had some instances when this happened and the only solution is to integrate in the code a condition which automatically convert the value of Kc as zero if it is negative and to 1 if the value is greater than 1.

Global Horizontal Irradiance

- The radiation reaching the earth's surface can be represented in a number of different ways. Global Horizontal Irradiance (GHI) is the total amount of shortwave radiation received from above by a surface horizontal to the ground.
- This value is of particular interest to photovoltaic installations and includes both Direct Normal Irradiance (DNI) and Diffuse Horizontal Irradiance (DHI).
- DNI is solar radiation that comes in a straight line from the direction of the sun at its current position in the sky.
- DHI is solar radiation that does not arrive on a direct path from the sun, but has been scattered by molecules and particles in the atmosphere and comes equally from all directions.
- The global horizontal irradiance (GHI) is the sum of DNI and the DHI values.
- The GHI value is calculated for this model using the HELIOSAT model, which directly gives the GHI value. In order to extract the DNI and DHI values from the GHI values further processing steps have to be done however we have avoided this since for solar power forecasting for photovoltaics (PV), GHI values are sufficient to predict the plant output forecast and DNI values are only required in the case of Concentrating photovoltaics (CSPV's) since they rely on the direct component of the incoming irradiance in order to perform energy.

Calculation of the Clear Sky GHI

- To calculate the value of the actual ghi we first need to calculate the clear sky ghi.
- The clear sky ghi can be easily calculated using the clear sky module in the PVLIB library.
- The user inputs for calculating the clear sky ghi are the latitude, longitude, frequency, elevation, date-time and model.
- The geographical parameters like latitude and longitude and temporal parameters like date-time can be easily found out by the user and will depend upon the area of interests.
- Another input is the model, this simply represents the model which will be used to calculate the clear sky ghi. There are many models which can be used to calculate the clear sky ghi the two most accurate and popular ones are the:
    1. Ineichen model
    2. Simplified solis

- Ineichen model uses the link turbidity factor to calculate the ghi values while the simplified solis model uses the Aerosol optical depth (AOD) and perceptible water (PW) to calculate the ghi.
- In our model we have used the simplified solis model to calculate the ghi values because studies have shown that this model works better for india due to the complexity in the aerosols content above India.
- As mentioned the input for the simplified solis model is the AOD and PW
- The monthly average value of the AOD is given by default in the clear sky module however we downloaded the AOD hourly data from the MEERA-2 satellite an atmospheric data monitoring satellite launched and maintained by NASA.
- This hourly data is downloaded and used to increase the accuracy of the model since the by default the model only uses the monthly AOD values, this increase the accuracy of the clear sky ghi.
- Similarly we calculated the PW monthly value from MERRA-2, since the model by default gives the yearly average value for ghi calculation.

Calculation of the Actual GHI

- After calculating the clear sky GHI, we calculate the actual GHI by multiplying the clear sky GHI with the clear sky index (Kc).

$$GHI = Kc * Clear\ Sky\ GHI \text{---------- (4)}$$

Where Kc is the clear sky index

- In an ideal case the value of the clear sky ghi and the actual ghi will match when the sky is clear, however no model that accurate has ever been built. The atmospheric dynamics and variability is complex to generate such an accurate model.

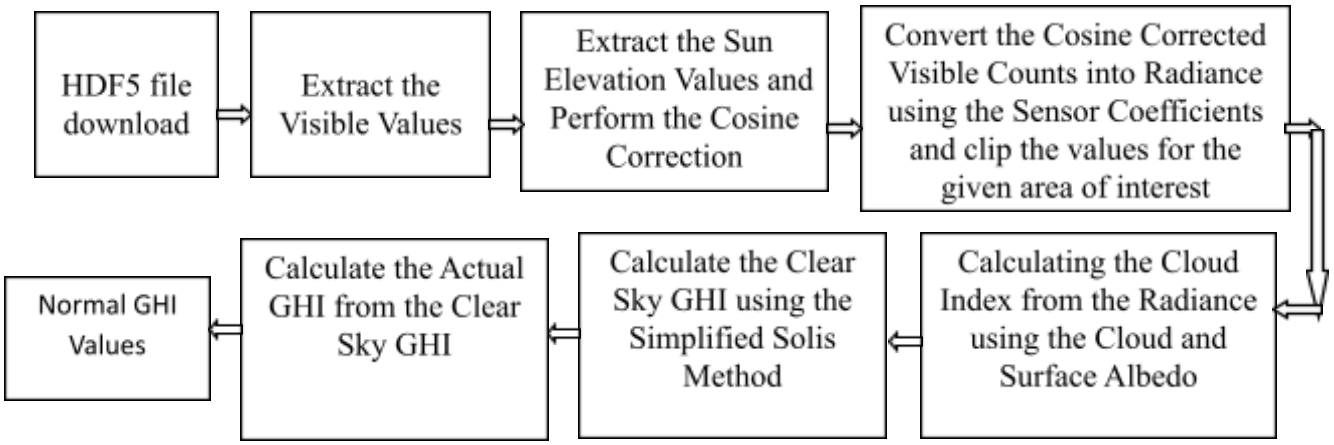


Figure 12 Flow of the Module 3

- We have implemented two forecasting algorithms namely block matching and optical flow to obtain the forecasted image frame from the given two consecutive frames.
- It is important to note that both the block matching and optical flow algorithm gives the output which is known as the cloud motion vectors (CMVS).
- These cloud motion vectors are then applied on the second image to extrapolate the second image into the future using these motion vectors
- This gives us the third image which is the forecast image.
- For example to find the forecast for the timestamp of 10:30 the images from 9:30 and 10:00 timestamps are passed to the BM or the optical flow algorithm, where these algorithms gives the cloud motion vectors.
- This cloud motion vector is then applied on the second image which is the image from 10:00
- This gives the third images which will be the forecasted image for the timestamp of 10:30.
- Since Open CV library is used for the Block Matching and the Optical Flow algorithm. it is important to note that Open CV only supports uint8 and uint16 integer types and hence the numpy arrays has to be converted into these integer types before being passed as the input for these algorithms.
- In the block matching algorithm, the input are the visible clipped values of the two consecutive timestamp.
- Once the forecasted visible clipped values are obtained we have to calculate the cloud index using the clear and cloudy pixel values.
- Then the normal GHI calculation process is followed and then we can obtain the forecasted GHI.
- This is repeated for further timestamps.
- In the optical flow algorithm, the inputs are the cloud index of the two consecutive timestamp.

- We get the forecasted Cloud Index. Then the normal GHI calculation process is followed and then we can obtain the forecasted GHI.
- This is repeated for further timestamps.
- Since the satellite data frequency is 30 minutes, the ideal forecast horizon is 30 minute for the most accurate data and we also kept a forecast horizon of 30 minutes.
- Ideally the forecast horizon can be increased to many hours but since the satellite resolution is not very high this is not very accurate. Also clouds not only move laterally but also disperse into different shapes and sizes within a span of some hours. If the forecast horizon is very high then the dispersive motion of the cloud is not taken into account and hence the accuracy further decreases.
- However with satellite images a highly accurate model with a forecast horizon of couple of hours can be successfully implemented with the right techniques.
- To achieve a forecast horizon of a few days the satellite model has to be integrated with the NWP.

Block Matching Algorithm

- A Block Matching Algorithm helps to locate matching macroblocks in a sequence of two consecutive image frames to attain the motion estimation between the frames.
- A block matching algorithm divides the given frame into macroblocks and compare each of the macroblocks with a corresponding block and with the adjacent frame.
- The movement of a macroblock from one location to another between two frames is captured in a vector. This process is followed for all the macroblocks.
- The complete procedure is repeated for the future frames.

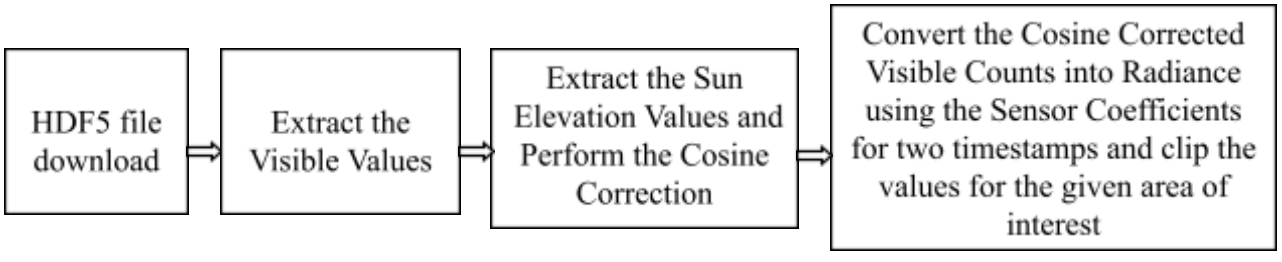
Algorithm

- Input the vis clipped image array values of two frames.
- Divide the area of search in each frame in four quadrants using clustering.
- Use the background subtractor to identify the separate the background and foreground values from the two frames.
- Mention the width and height.
- Start search with three locations, one at center (A) and others (B and C), S=4 locations away from A in orthogonal directions.

- Find points in search quadrant for second phase using the weight distribution for A, B, C:
- If (M(A)>=M(B) and M(A)>=M(C)), select points in second phase quadrant IV.
- If (M(A)>=M(B) and M(A)<=M(C)), select points in second phase quadrant I.
- If (M(A)<M(B) and M(A)<M(C)), select points in second phase quadrant II
- If (M(A)<M(B) and M(A)>=M(C)), select points in second phase quadrant III
- Find the location with lowest weight.
- Set the new search origin as the point found above.
- Set the new step size as S = S/2.
- Repeat the SES search procedure until S=1.
- Select the location with lowest weight as motion vector.
- Return the vector values as output.

Optical Flow Algorithm

- Optical flow is the pattern of apparent motion of image objects between two consecutive frames caused by the movement of object or camera.
- It is 2D vector field where each vector is a displacement vector showing the movement of points from first frame to second.
- It can be of two types-Sparse Optical flow and Dense Optical flow.
- Optical flow algorithm works on the concept of water flow.
- The algorithm detects the change in brightness intensity and on the basis of that it detects the direction of motion.
- Optical flow has been historically used in solar forecasting by many scientists all over the world, the reason being the easy to understand algorithm and the proven accuracy of this technique.
- In case of satellite images, the brightness of each image will change with the motion of clouds and the algorithm detects this change and on the basis of that it predicts the future motion of the cloud.
- The popularly used optical flow algorithms for solar forecasting are:
  1. Lucas kanade method
  2. Dualtvl1 optical flow
- While Lucas Kanade method is a very reliable model to use, it is not that accurate over south east asia and hence we have used the DualTvl1 optical flow.



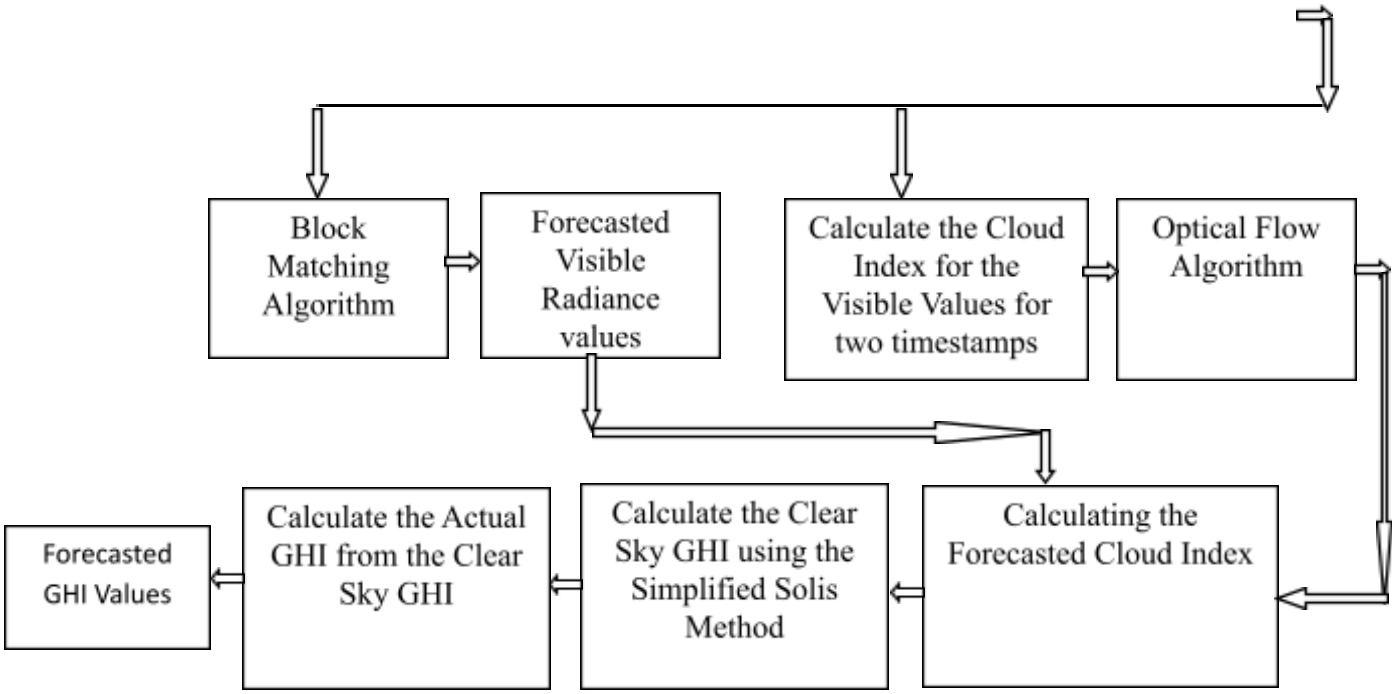


Figure 13 Flow of the Module 4

5. Results & Discussions

A. Performance Metrics

The Performance metrics which is used to evaluate the models to identify the errors between the actual and the predicted values. In this paper, three metrics are used for evaluation. They are Root Mean Squared Error (RMSE), Mean Absolute Error (MAE). The formulas are as follows

$$RMSE = \sqrt{(i/n)\sum_{i=1}^{n}(y_i - y_i')^2} \quad \text{---------- (5)}$$

Where y_i represent an actual value y_i' represents the predicted value.

$$MAE = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n} \quad \text{---------- (6)}$$

Where y_i represents the predicted values and the x_i represents the actual values.

B. Include Graphs, Screenshots and Tables

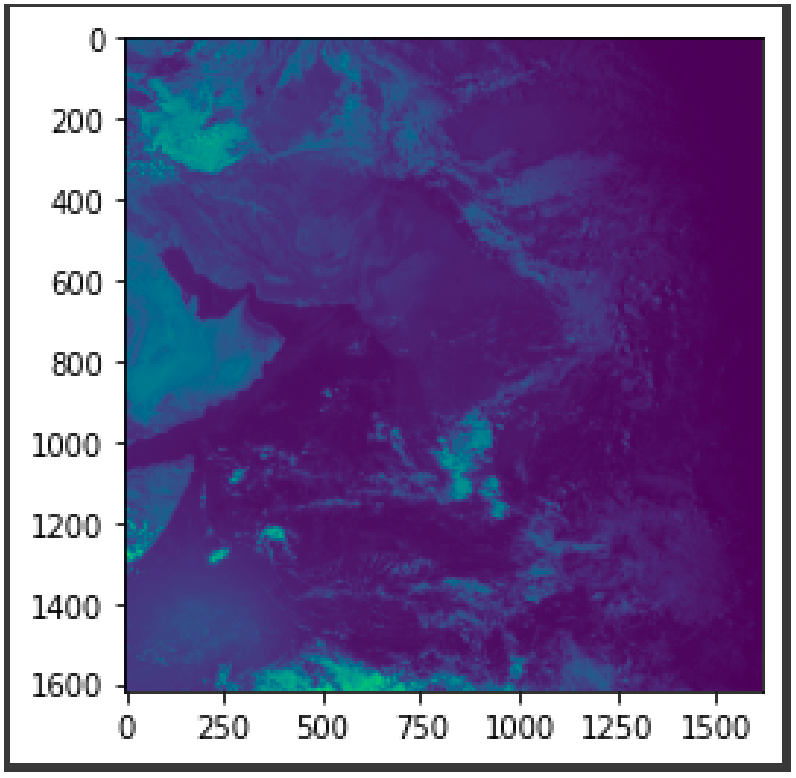


Figure 14 Cosine Corrected VIS Image

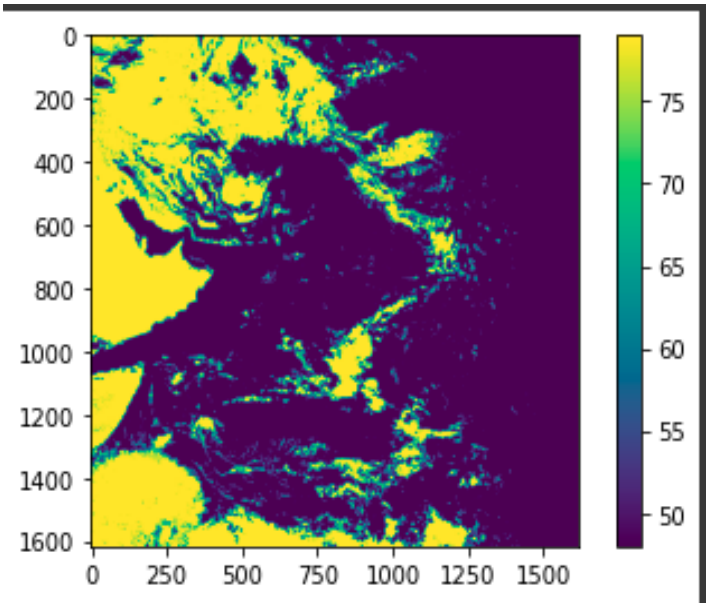


Figure 15 Cloud Detection using K-Means Clustering

The yellow portions in the image are the clouds and the other portions are clear

```
[[1 1 1 ... 0 0 0]
 [1 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
 [1 1 1 ... 0 0 0]
 [1 1 1 ... 0 0 0]
 [1 1 1 ... 0 0 0]]
```

Figure 16 Result of K-Means: Mask Values

1 represents the cloudy regions and 0 represents the clear sky regions

Ground Data Plots

Ghi values measured by the ground station for Kutch, Jaisalmer and tirupati for the month of september. Ideally daily ghi curve should look like a hill with ghi values very low in the morning increasing in the afternoon and then decreasing again in the evening. If clouds are there this hilly pattern will no longer be prevalent and clouds decrease the ghi value drastically. Hence in this diagram the hilly curves can be concluded to have clear sky conditions and distorted curves can be considered to be the cloud sky conditions. From Tirupati ground data plot, we can say that GHI CURVE in that ground station is far from a perfect hill, this means that cloudiness is very prevalent in tirupati as compared to Jaisalmer and Kutch which has many clear sky days.

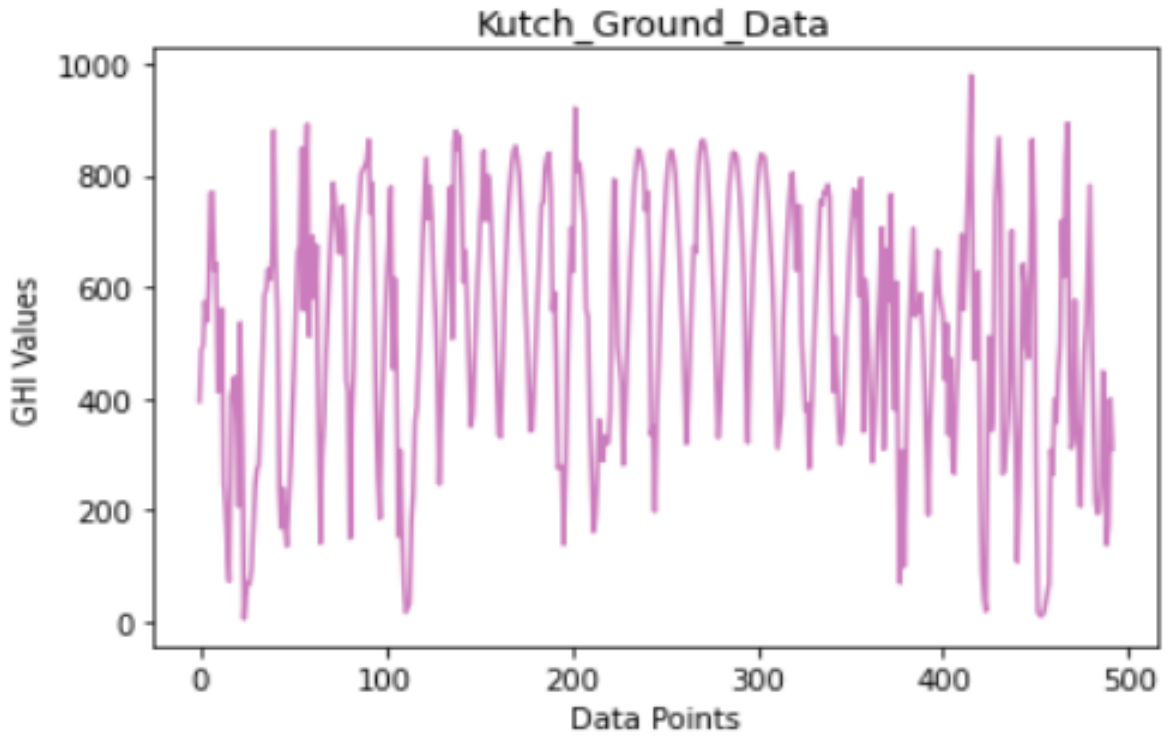


Figure 17 Kutch Ground Data

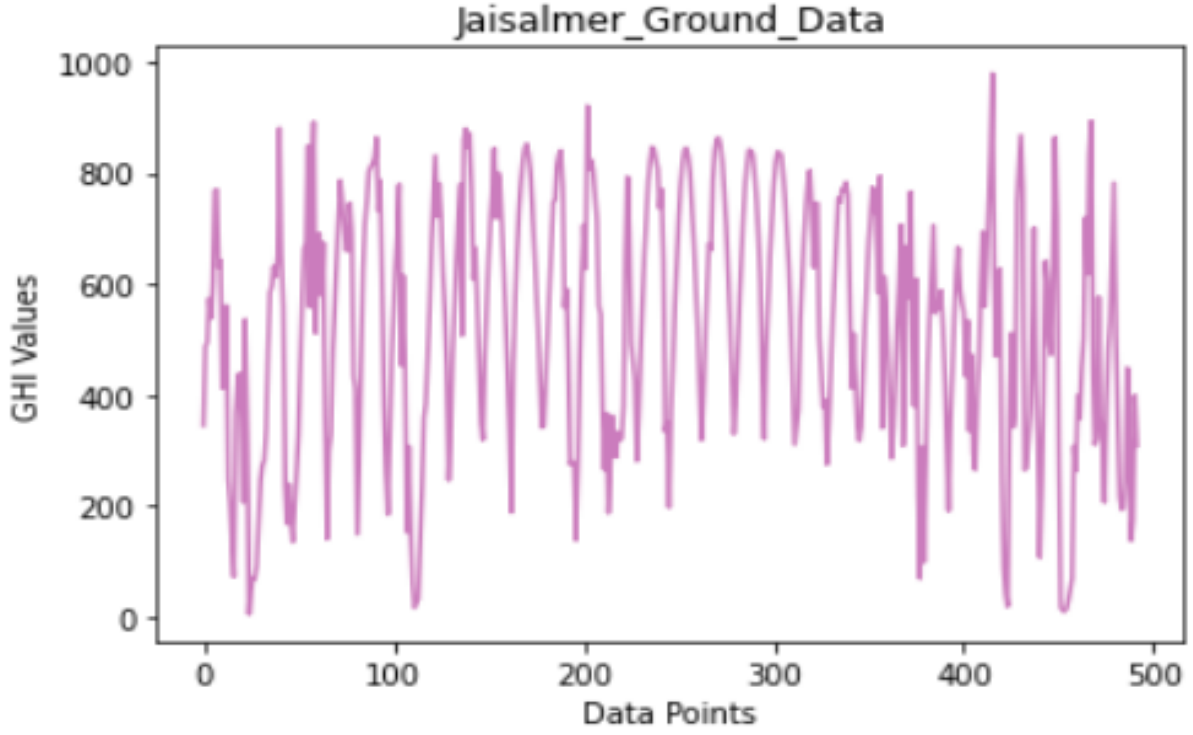


Figure 18 Jaisalmer Ground Data

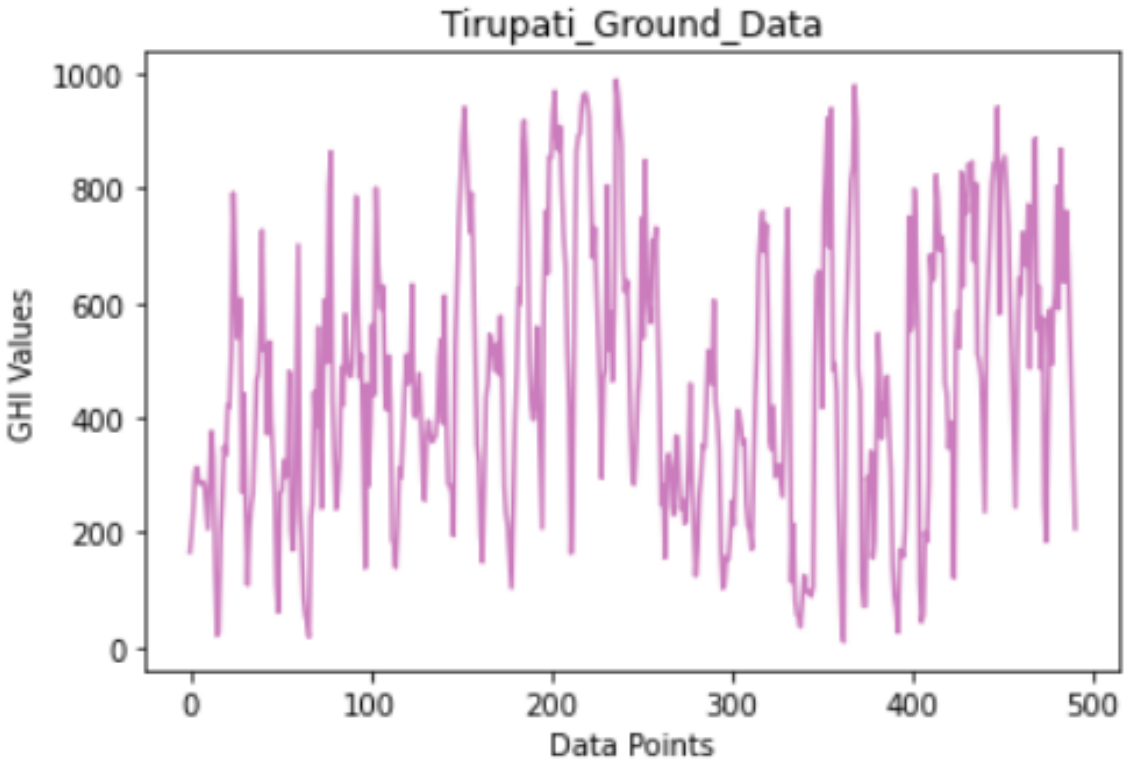


Figure 19 Tirupati Ground Data

Block Matching Algorithm Plots

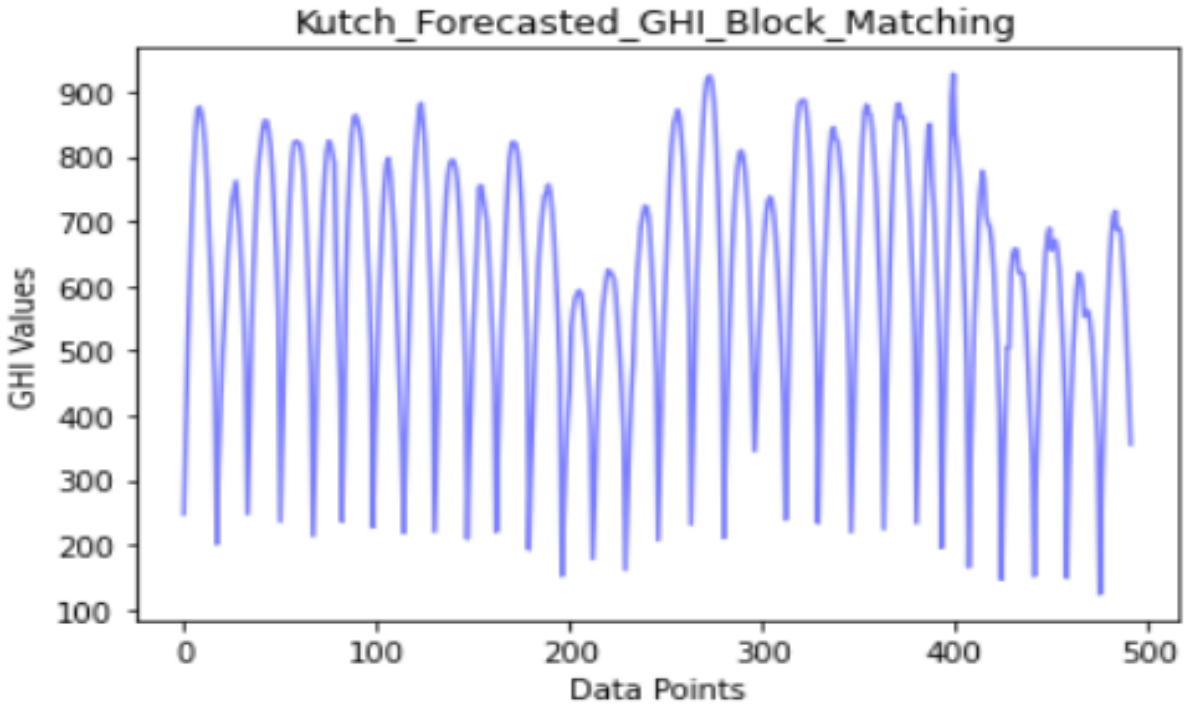


Figure 20 Kutch Block Matching Forecasted Result

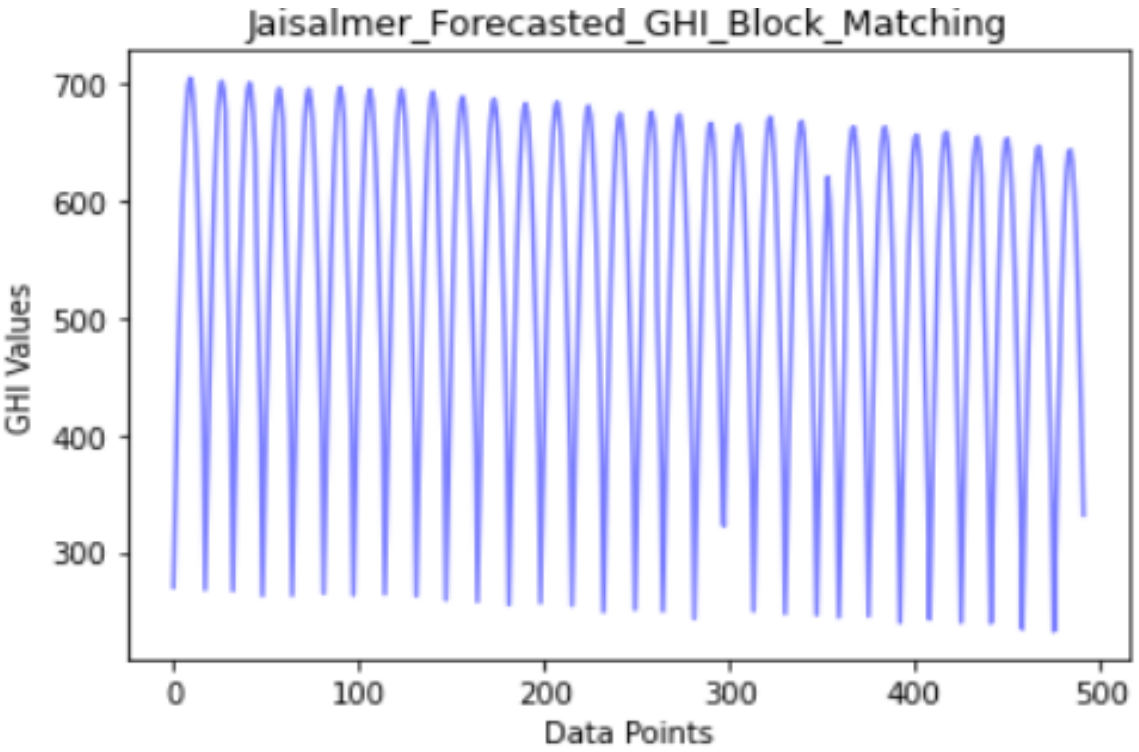


Figure 21 Jaisalmer Block Matching Forecasted Plot

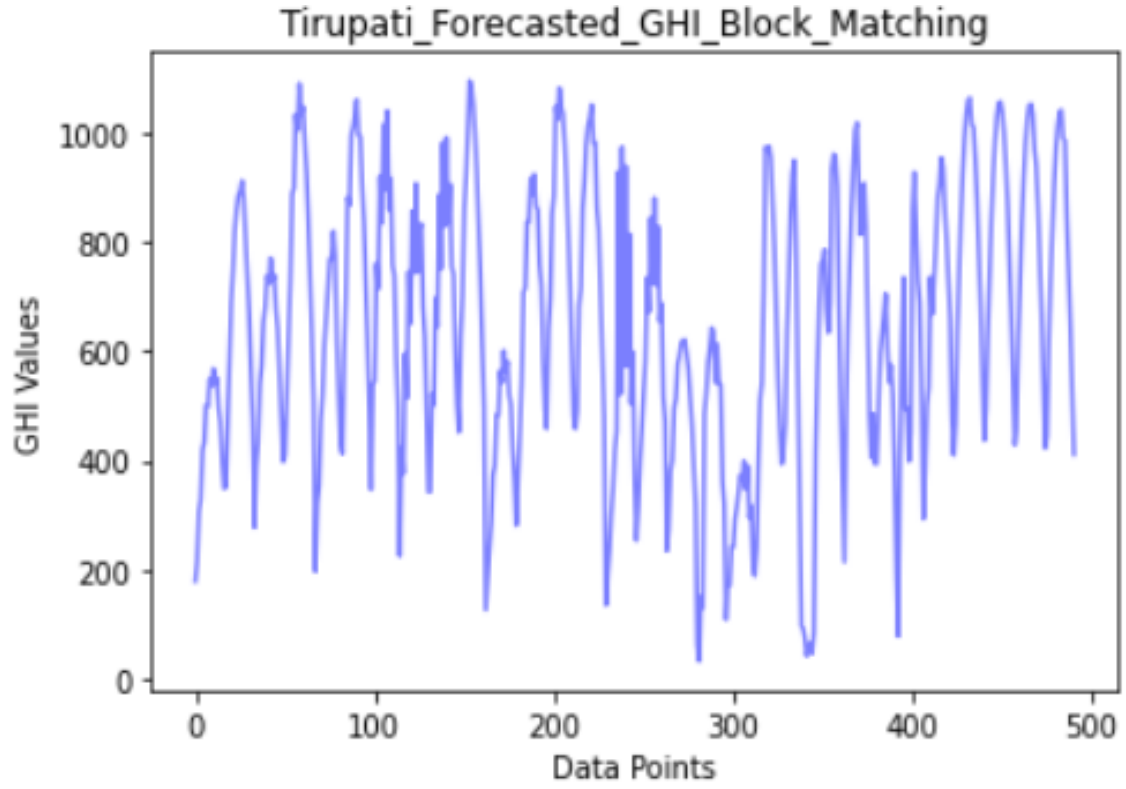


Figure 22 Tirupati Block Matching Forecasted Plot

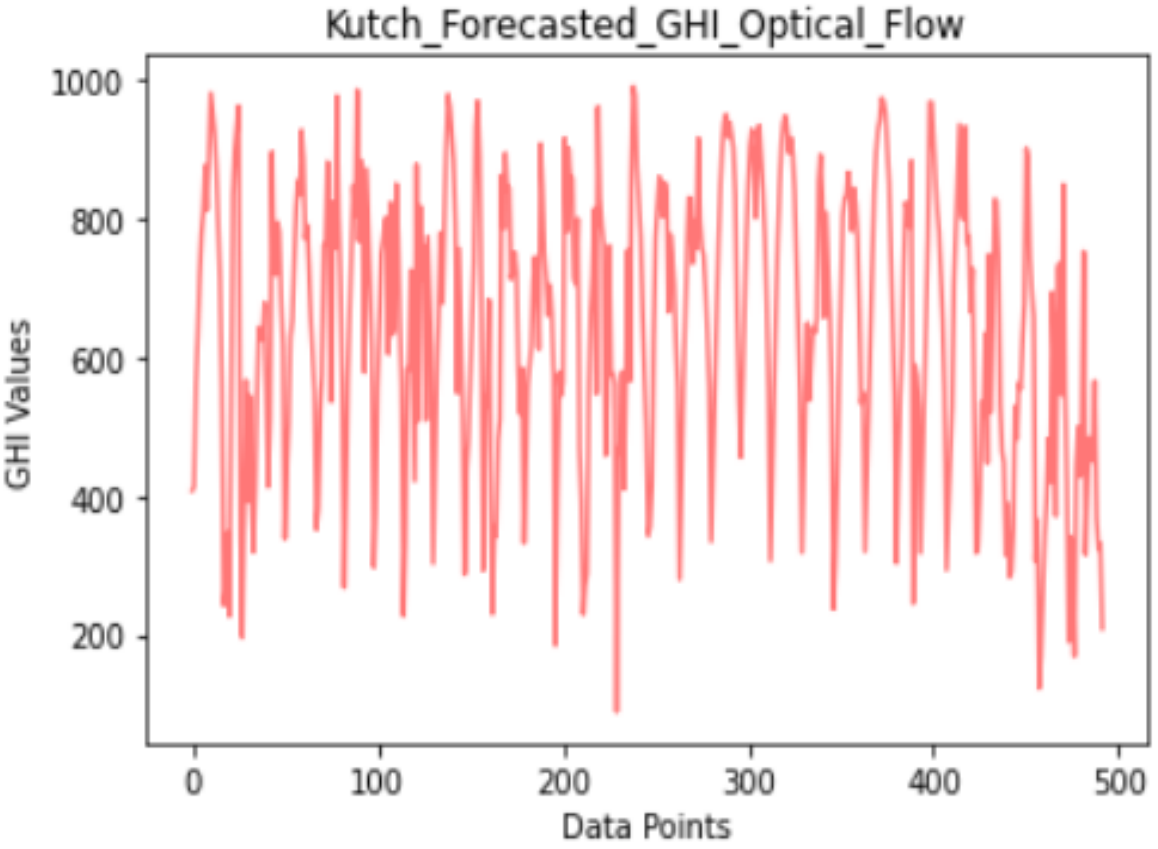


Figure 23 Kutch Optical Flow Forecasted Plot

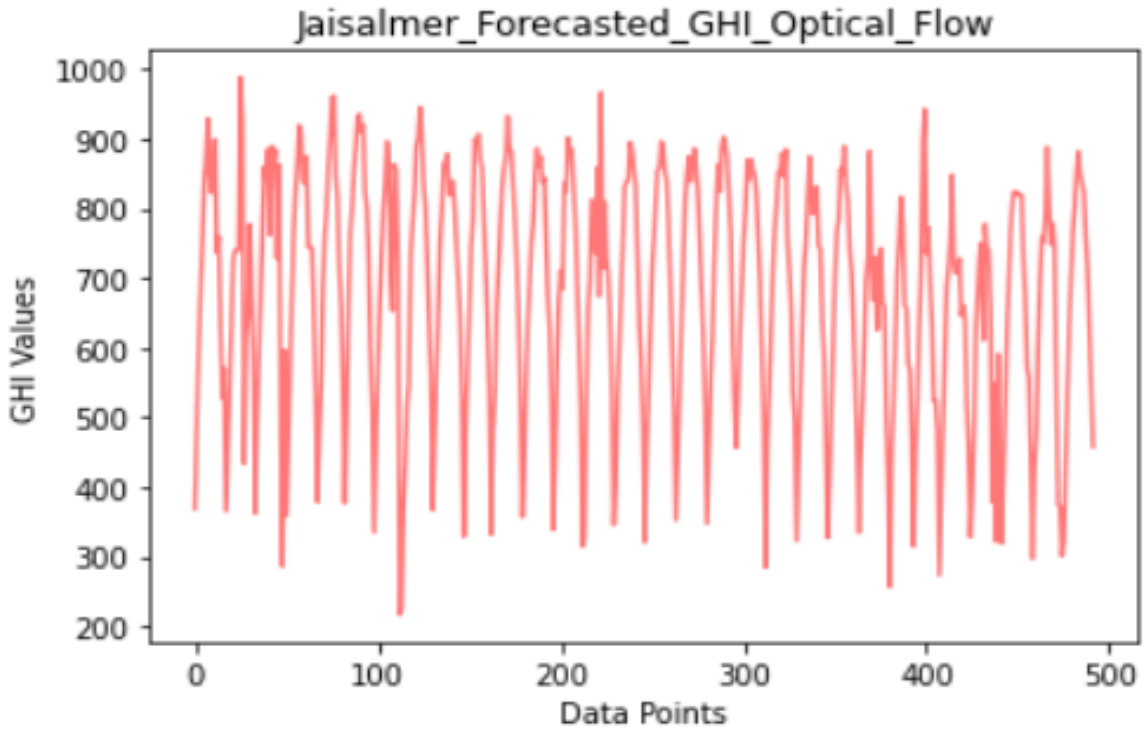


Figure 24Jaisalmer Optical Flow Forecasted Plot

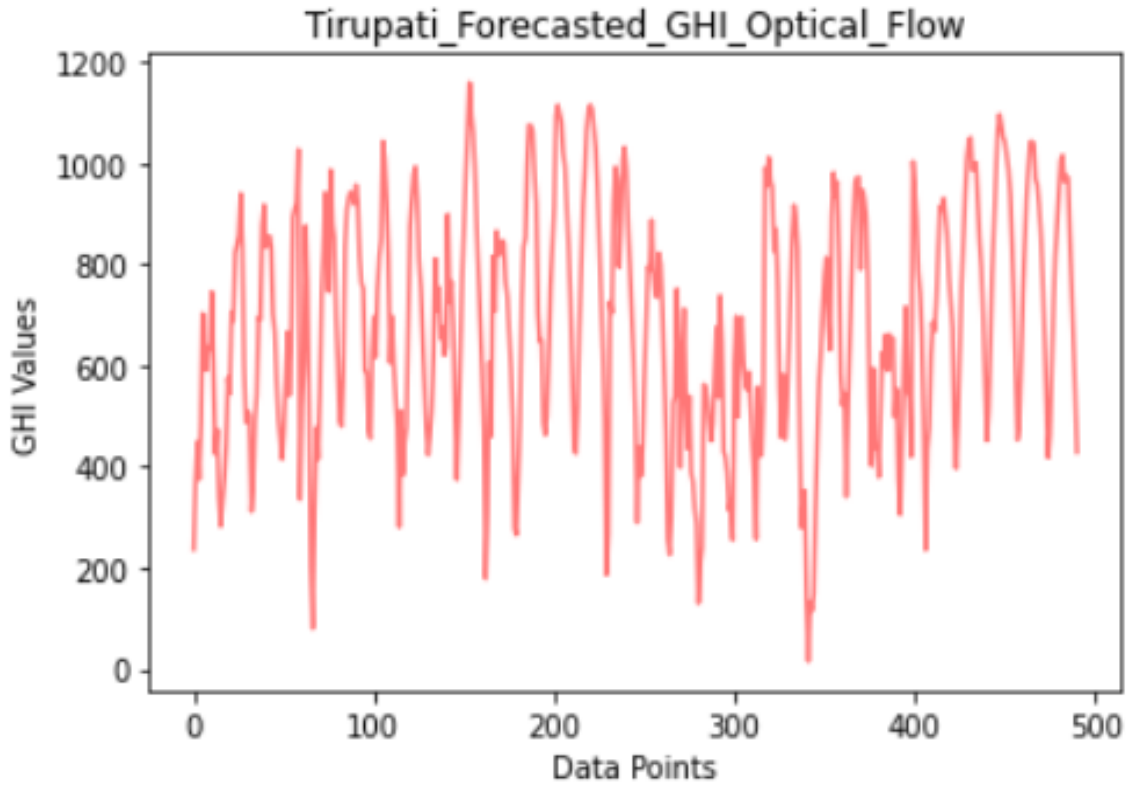


Figure 25Tiupati Optical Flow Forecasted Plot

Ground Data GHI Values

Forecasted GHI Values

Performance Metrics Results

Location	RMSE	RMSE %	MAE	MAE %
Kutch	224.37	42.49	170.83	32.35
Jaisalmer	212.116	40.23	169.53	32.16
Tirupati	276.614	59.60	220.49	

Table 1. Performance Metrics Results Block Matching Algorithm

Location	RMSE	RMSE %	MAE	MAE %
Kutch	252.63	47.84	184.56	
Jaisalmer	243.72	46.22	169.83	
Tirupati	296.0	63.80	219.53	

Table 2 Performance Metrics Results Optical Flow Algorithm

Ground Data Measurements

Date	kutch_Ground_Data
9/1/2019 8:00	394.8654643
9/1/2019 8:30	487.2270915
9/1/2019 9:00	495.4251851
9/1/2019 9:30	574.1589589
9/1/2019 10:00	538.7559031
9/1/2019 10:30	603.3165009
9/1/2019 11:00	768.839978
9/1/2019 11:30	770.427713
9/1/2019 12:00	629.397701
9/1/2019 12:30	642.0499847
9/1/2019 13:00	412.672937
9/1/2019 13:30	455.7621033
9/1/2019 14:00	561.318217
9/1/2019 14:30	248.2773321
9/1/2019 15:00	206.3469732
9/1/2019 15:30	131.9709226
9/1/2019 16:00	72.53506673
9/2/2019 8:00	404.2268656
9/2/2019 8:30	420.029184
9/2/2019 9:00	439.3201426
9/2/2019 9:30	350.4703807
9/2/2019 10:00	207.1142802
9/2/2019 10:30	536.7526072
9/2/2019 11:00	331.3286552
9/2/2019 11:30	5.683643688
9/2/2019 12:00	49.86886648
9/2/2019 12:30	71.32095159
9/2/2019 13:00	69.33854523
9/2/2019 14:00	93.12641525
9/2/2019 14:30	167.424673
9/2/2019 15:00	246.0051443
9/2/2019 15:30	274.0596649
9/2/2019 16:00	282.6761393

Table 3 Kutch Ground Data Measurements

Date	Jaisalmer_Ground_data
9/1/2021 8:00	345.7229919
9/1/2021 8:30	487.2270915
9/1/2021 9:00	495.4251851

9/1/2021 9:30	574.1589589
9/1/2021 10:00	538.7559031
9/1/2021 10:30	603.3165009
9/1/2021 11:00	768.839978
9/1/2021 11:30	770.427713
9/1/2021 12:00	629.397701
9/1/2021 12:30	642.0499847
9/1/2021 13:00	412.672937
9/1/2021 13:30	455.7621033
9/1/2021 14:00	561.318217
9/1/2021 14:30	248.2773321
9/1/2021 15:00	206.3469732
9/1/2021 15:30	131.9709226
9/1/2021 16:00	72.53506673
9/2/2021 8:00	365.5367126
9/2/2021 8:30	420.029184
9/2/2021 9:00	439.3201426
9/2/2021 9:30	350.4703807
9/2/2021 10:00	207.1142802
9/2/2021 10:30	536.7526072
9/2/2021 11:00	331.3286552
9/2/2021 11:30	5.683643688
9/2/2021 12:00	49.86886648
9/2/2021 12:30	71.32095159
9/2/2021 13:00	69.33854523
9/2/2021 13:30	93.12641525
9/2/2021 15:00	167.424673
9/2/2021 15:30	246.0051443
9/2/2021 16:00	274.0596649

Table 4 Jaisalmer Ground Data Measurements

Date	Tirupati_Ground_data
9/1/2019 8:00	165.8087331
9/1/2019 8:30	196.5782995
9/1/2019 9:00	245.7539622
9/1/2019 9:30	306.9275737

9/1/2019 10:00	313.5953237
9/1/2019 10:30	289.1922831
9/1/2019 11:00	283.9182902
9/1/2019 11:30	288.9764628
9/1/2019 12:00	284.8119222
9/1/2019 12:30	243.5460093
9/1/2019 13:00	206.1222427
9/1/2019 13:30	285.0650045
9/1/2019 14:00	376.6465718
9/1/2019 14:30	331.9203969
9/1/2019 15:00	140.0060913
9/1/2019 15:30	20.4063097
9/1/2019 16:00	25.28011824
9/2/2019 8:00	198.4955582
9/2/2019 8:30	262.1983042
9/2/2019 9:00	349.989918
9/2/2019 9:30	334.9771637
9/2/2019 10:00	423.2152344
9/2/2019 10:30	418.1170797
9/2/2019 11:00	515.9654429
9/2/2019 11:30	792.4065898
9/2/2019 12:00	695.8677429
9/2/2019 12:30	539.5973083
9/2/2019 13:00	544.9462209
9/2/2019 14:00	608.6760813
9/2/2019 14:30	270.598938
9/2/2019 15:00	441.7680985
9/2/2019 15:30	243.4523438
9/2/2019 16:00	108.7634776

Table 5 Tirupati Ground Data Measurements

Forecasted Measurements

Block Matching Forecasted Results

Date	Forecast_GHI_kutch

1/9/2019 8:00	247.256722
1/9/2019 8:30	387.4644155
1/9/2019 9:00	506.6102529
1/9/2019 9:30	615.4785955
1/9/2019 10:00	707.0084114
1/9/2019 10:30	779.5063669
1/9/2019 11:00	838.3781037
1/9/2019 11:30	871.940393
1/9/2019 12:00	876.2853844
1/9/2019 12:30	867.6546951
1/9/2019 13:00	848.3229569
1/9/2019 13:30	816.6028688
1/9/2019 14:00	759.8330211
1/9/2019 14:30	689.5190268
1/9/2019 15:00	609.143756
1/9/2019 15:30	538.0915941
1/9/2019 16:00	451.5802348
2/9/2019 8:00	200.7932438
2/9/2019 8:30	324.7910647
2/9/2019 9:00	417.2419352
2/9/2019 9:30	492.6768645
2/9/2019 10:00	546.4539219
2/9/2019 10:30	607.3697445
2/9/2019 11:00	657.0365382
2/9/2019 11:30	696.4154722
2/9/2019 12:00	732.2536786
2/9/2019 12:30	747.8680724
2/9/2019 13:00	761.7134624
2/9/2019 14:00	723.3668162
2/9/2019 14:30	681.3561649
2/9/2019 15:00	619.3714312
2/9/2019 15:30	557.0906808
2/9/2019 16:00	471.9965977

Table 6 Kutch Block Matching Forecasted Measurements

Date	Forecast_GHI_Jaisalmer
9/1/2021 8:00	269.8440531
9/1/2021 8:30	350.167833

9/1/2021 9:00	425.981862
9/1/2021 9:30	495.5232993
9/1/2021 10:00	557.03852
9/1/2021 10:30	609.1912095
9/1/2021 11:00	650.8487521
9/1/2021 11:30	681.1373106
9/1/2021 12:00	699.3656595
9/1/2021 12:30	705.2330719
9/1/2021 13:00	698.5692438
9/1/2021 13:30	679.5040821
9/1/2021 14:00	648.455039
9/1/2021 14:30	606.2520556
9/1/2021 15:00	553.4637032
9/1/2021 15:30	491.3791804
9/1/2021 16:00	421.3487289
9/2/2021 8:00	267.6224132
9/2/2021 8:30	347.8982538
9/2/2021 9:00	423.7318841
9/2/2021 9:30	493.2199539
9/2/2021 10:00	554.7177168
9/2/2021 10:30	606.8655534
9/2/2021 11:00	648.4464035
9/2/2021 11:30	678.6610265
9/2/2021 12:00	696.7906436
9/2/2021 12:30	702.5342862
9/2/2021 13:00	695.7166739
9/2/2021 13:30	676.4930796
9/2/2021 15:00	549.9064038
9/2/2021 15:30	487.6539227
9/2/2021 16:00	417.5252898

Table 7 Jaisalmer Block Matching Forecasted Measurements

Date	Forecast_GHI_tirupati
1/9/2019 8:00	179.8587645
1/9/2019 8:30	215.0227065
1/9/2019 9:00	307.3489285
1/9/2019 9:30	328.5974199
1/9/2019 10:00	420.4730611

1/9/2019 10:30	433.1343259
1/9/2019 11:00	501.3826802
1/9/2019 11:30	498.733188
1/9/2019 12:00	548.58825
1/9/2019 12:30	536.5679352
1/9/2019 13:00	568.3400422
1/9/2019 13:30	540.2958334
1/9/2019 14:00	550.9195447
1/9/2019 14:30	494.6258012
1/9/2019 15:00	470.4727893
1/9/2019 15:30	398.3204645
1/9/2019 16:00	349.1923035
2/9/2019 8:00	351.5138306
2/9/2019 8:30	463.2979441
2/9/2019 9:00	578.5488009
2/9/2019 9:30	688.263661
2/9/2019 10:00	746.3009791
2/9/2019 10:30	820.709362
2/9/2019 11:00	860.614656
2/9/2019 11:30	884.7113526
2/9/2019 12:00	892.4908462
2/9/2019 12:30	911.9705304
2/9/2019 13:00	880.6839697
2/9/2019 14:00	810.9224819
2/9/2019 14:30	732.5832278
2/9/2019 15:00	669.6470932
2/9/2019 15:30	565.6881833
2/9/2019 16:00	472.7568893

Table 8 Tirupati Block Matching Forecasted Measurements

Optical Flow Forecasted Results

Date	Forecast_GHI_kutch
1/9/2019 8:00	408
1/9/2019 8:30	414.5459998
1/9/2019 9:00	560.8718991
1/9/2019 9:30	630.7039981
1/9/2019 10:00	718.3147195
1/9/2019 10:30	777.888987
1/9/2019 11:00	811.1114527

1/9/2019 11:30	877.1070381
1/9/2019 12:00	812.9543629
1/9/2019 12:30	887.6022793
1/9/2019 13:00	981.09849
1/9/2019 13:30	955.3205018
1/9/2019 14:00	925.6213294
1/9/2019 14:30	874.5478955
1/9/2019 15:00	798.6607313
1/9/2019 15:30	713.6125863
1/9/2019 16:00	557.0154837
2/9/2019 8:00	242.5027488
2/9/2019 8:30	261.8346245
2/9/2019 9:00	347.4355586
2/9/2019 9:30	226.8518525
2/9/2019 10:00	555.9780569
2/9/2019 10:30	832.6616698
2/9/2019 11:00	899.2898516
2/9/2019 11:30	931.8055772
2/9/2019 12:00	961.9664734
2/9/2019 12:30	341.4713223
2/9/2019 13:00	196.1967545
2/9/2019 14:00	368.5714083
2/9/2019 14:30	567.0955602
2/9/2019 15:00	392.9557417
2/9/2019 15:30	544.0965186
2/9/2019 16:00	389.2368362

Table 9 Kutch Optical Flow Forecasted Measurements

Date	Forecast_GHI_Jaisalmer
1/9/2019 8:00	369
1/9/2019 8:30	478.2036222
1/9/2019 9:00	591.5001098
1/9/2019 9:30	673.2994053
1/9/2019 10:00	762.1372297
1/9/2019 10:30	830.6387371
1/9/2019 11:00	868.7195297
1/9/2019 11:30	928.6678557
1/9/2019 12:00	850.5446132
1/9/2019 12:30	822.2748794
1/9/2019 13:00	844.5839492
1/9/2019 13:30	898.790087
1/9/2019 14:00	735.9748764
1/9/2019 14:30	760.1118621
1/9/2019 15:00	632.342295
1/9/2019 15:30	528.1861218
1/9/2019 16:00	570.4190435
2/9/2019 8:00	365.7518859
2/9/2019 8:30	455.9735342
2/9/2019 9:00	562.2212227
2/9/2019 9:30	631.5820405
2/9/2019 10:00	726.874571
2/9/2019 10:30	740.0172384
2/9/2019 11:00	738.4953469
2/9/2019 11:30	742.0431494

2/9/2019 12:00	988.3484919
2/9/2019 12:30	921.9453356
2/9/2019 13:00	433.6951036
2/9/2019 14:00	612.3559474
2/9/2019 14:30	648.3508078
2/9/2019 15:00	776.6516369
2/9/2019 15:30	680.707157
2/9/2019 16:00	586.3495671

Table 10 Jaisalmer Optical Flow Forecasted Measurements

Date	Forecast_GHI_tirupati
1/9/2019 8:00	236.5094905
1/9/2019 8:30	368.4297927
1/9/2019 9:00	449.1143056
1/9/2019 9:30	374.6917433
1/9/2019 10:00	484.0370678
1/9/2019 10:30	702.6791773
1/9/2019 11:00	608.4187532
1/9/2019 11:30	591.0301759
1/9/2019 12:00	637.2641112
1/9/2019 12:30	633.2101868
1/9/2019 13:00	745.4196798
1/9/2019 13:30	582.4850985
1/9/2019 14:00	425.22403
1/9/2019 14:30	473.644561
1/9/2019 15:00	337.8373672
1/9/2019 15:30	281.9561
1/9/2019 16:00	330.943326
2/9/2019 8:00	379.9398078
2/9/2019 8:30	442.1522487
2/9/2019 9:00	575.5924171
2/9/2019 9:30	544.1148257
2/9/2019 10:00	704.283118
2/9/2019 10:30	688.0754106
2/9/2019 11:00	823.8158843
2/9/2019 11:30	837.9940584
2/9/2019 12:00	857.5929179
2/9/2019 12:30	940.0980773
2/9/2019 13:00	839.1596766

2/9/2019 14:00	595.0752764
2/9/2019 14:30	488.2838121
2/9/2019 15:00	509.2817652
2/9/2019 15:30	501.7774631
2/9/2019 16:00	311.4589615

Table 11 Tirupati Optical Flow Forecasted Measurements

6. Conclusion and Future Work

- From the error metrics results, Average Root mean square error is 40%. As per the research papers, the minimum error percentage reported for the forecasting models was 30%
- The RMSE is high when compared to the error percentage of the prior research paper work but such high values due to the fact that we have used satellite images for our work.
- The RMSE values for tirupati are very high as compared to the RMSE for Jaisalmer and Kutch. Since tirupati has a tropical weather near the coastline it has more cloudiness as compared to the arid regions of Jaisalmer and Kutch.
- From the above we could say that our model is accurate for clear sky conditions and not accurate for cloudy conditions because the model is unable to detect the clouds.
- There are many problems in the INSAT-3D satellite images. The satellite resolution is very low as compared to other geostationary satellites launched and maintained by other space agencies like NASA and ESA.
- Some timestamps data are missing which not only makes the sequential coding difficult but also increases the forecasting horizon which reduces the models efficiency.
- There are errors in many INSAT-3d satellite data, for example for the month of september the data for the timestamps 11 and 11:30 had errors in them and this error was prevalent for the entire month.
- To increase the efficiency of the model, deep learning algorithms can be used for cloud detection and finding the CMVS associated with those clouds.
- Further to this, our model can be evaluated with the NASA's Landsat data and can be optimized even better based on the results obtained from the error metrics.

.

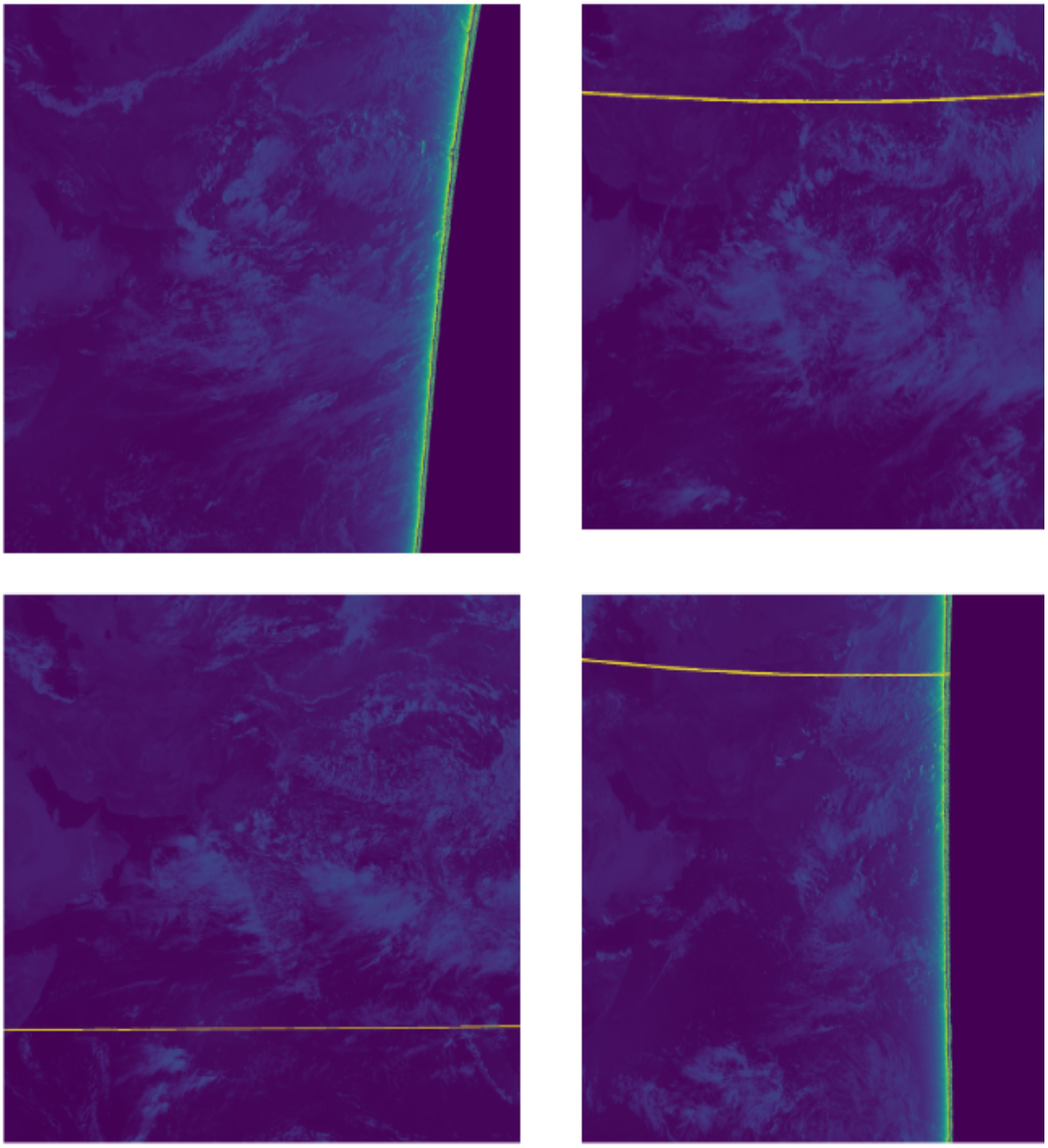


Figure 26 Sample Error Images in the Satellite Data

As can be seen on the above given images there are lines and uneven shadings in these satellite images. There were many such images and we strongly believe that these erroneous images hampered the models accuracy.

7. References

[1] W. Xing, G. Zhang, and S. Poslad, “Estimation of global horizontal irradiance in China using a deep learning method,” International Journal of Remote Sensing, vol. 42, no. 10, pp. 3899–3917, May 2021, doi: 10.1080/01431161.2021.1887539.

[2] R. A. Rajagukguk, R. Kamil, and H.-J. Lee, “A Deep Learning Model to Forecast Solar Irradiance Using a Sky Camera,” Applied Sciences, vol. 11, no. 11, p. 5049, May 2021, doi: 10.3390/app11115049.

[3] D. Pattanaik, S. Mishra, G. P. Khuntia, R. Dash, and S. C. Swain, “An innovative learning approach for solar power forecasting using genetic algorithm and artificial neural network,” Open Engineering, vol. 10, no. 1, pp. 630–641, Jul. 2020, doi: 10.1515/eng-2020-0073.

[4] S. Mahajan and B. Fataniya, “Cloud detection methodologies: variants and development—a review,” Complex Intell. Syst., vol. 6, no. 2, pp. 251–261, Jul. 2020, doi: 10.1007/s40747-019-00128-0.

[5] V. Kallio-Myers, A. Riihelä, P. Lahtinen, and A. Lindfors, “Global horizontal irradiance forecast for Finland based on geostationary weather satellite data,” Solar Energy, vol. 198, pp. 68–80, Mar. 2020, doi: 10.1016/j.solener.2020.01.008.

[6] B. Benamrou, M. Ouardouz, I. Allaouzi, and M. Ben Ahmed, “A Proposed Model to Forecast Hourly Global Solar Irradiation Based on Satellite Derived Data, Deep Learning and Machine Learning Approaches,” J. Ecol. Eng., vol. 21, no. 4, pp. 26–38, May 2020, doi: 10.12911/22998993/119795.

[7] P. A. G. M. Amarasinghe, N. S. Abeygunawardana, T. N. Jayasekara, E. A. J. P. Edirisinghe, S. K. Abeygunawardane, and Department of Electrical Engineering, University of Moratuwa, Sri Lanka, “Ensemble models for solar power forecasting—a weather classification approach,” AIMS Energy, vol. 8, no. 2, pp. 252–271, 2020, doi: 10.3934/energy.2020.2.252.

[8] E. T. Velasco and I. B. Salbidegoitia, “New methodology for solar irradiance calculation using Meteosat satellite imagery,” Casablanca, Morocco, 2019, p. 190016. doi: 10.1063/1.5117713.

[9] J. H. Jeppesen, R. H. Jacobsen, F. Inceoglu, and T. S. Toftegaard, “A cloud detection algorithm for satellite imagery based on deep learning,” Remote Sensing of Environment, vol. 229, pp. 247–259, Aug. 2019, doi: 10.1016/j.rse.2019.03.039.

[10] B. Ameen, H. Balzter, C. Jarvis, and J. Wheeler, “Modelling Hourly Global Horizontal Irradiance from Satellite-Derived Datasets and Climate Variables as New Inputs with Artificial Neural Networks,” Energies, vol. 12, no. 1, p. 148, Jan. 2019, doi: 10.3390/en12010148.

[11] S. Mohajerani, T. A. Krammer, and P. Saeedi, “Cloud Detection Algorithm for Remote Sensing Images Using Fully Convolutional Neural Networks,” arXiv:1810.05782 [cs], Oct. 2018, Accessed: Jul. 29, 2021. [Online]. Available: http://arxiv.org/abs/1810.05782

[12] B. Sivaneasan, C. Y. Yu, and K. P. Goh, “Solar Forecasting using ANN with Fuzzy Logic Pre-processing,” Energy Procedia, vol. 143, pp. 727–732, Dec. 2017, doi: 10.1016/j.egypro.2017.12.753.

[13] Data Source: *https://www.isro.gov.in/Spacecraft/insat-3d*

[14] Data Source: *https://niwe.res.in/*

[15] Link for the code: *https://github.com/kapasi1234/solar-forecasting-using-INSAT-3D-satelite-images*