

# Mapping Greenhouse Gas Emissions Using Satellite and Model-Based Datasets

## Introduction

Objective: Develop a system to map and monitor GHG emissions using satellite data and model-based datasets.

GHG Types: Focus on key gases like Carbon Dioxide (CO<sub>2</sub>), Methane (CH<sub>4</sub>), and Nitrous Oxide (N<sub>2</sub>O).

Purpose: Climate modeling, policy support, environmental management.

# **Step 1 – Defining the Scope**

Geographical Scope: Global, national, or regional levels.

Temporal Scope: Select time intervals such as daily, weekly, or monthly.

Target Greenhouse Gas Types: CO<sub>2</sub>, CH<sub>4</sub>, N<sub>2</sub>O, among others.

Purpose: Applications may include scientific research, policy advisement, and strategies for emission reduction.

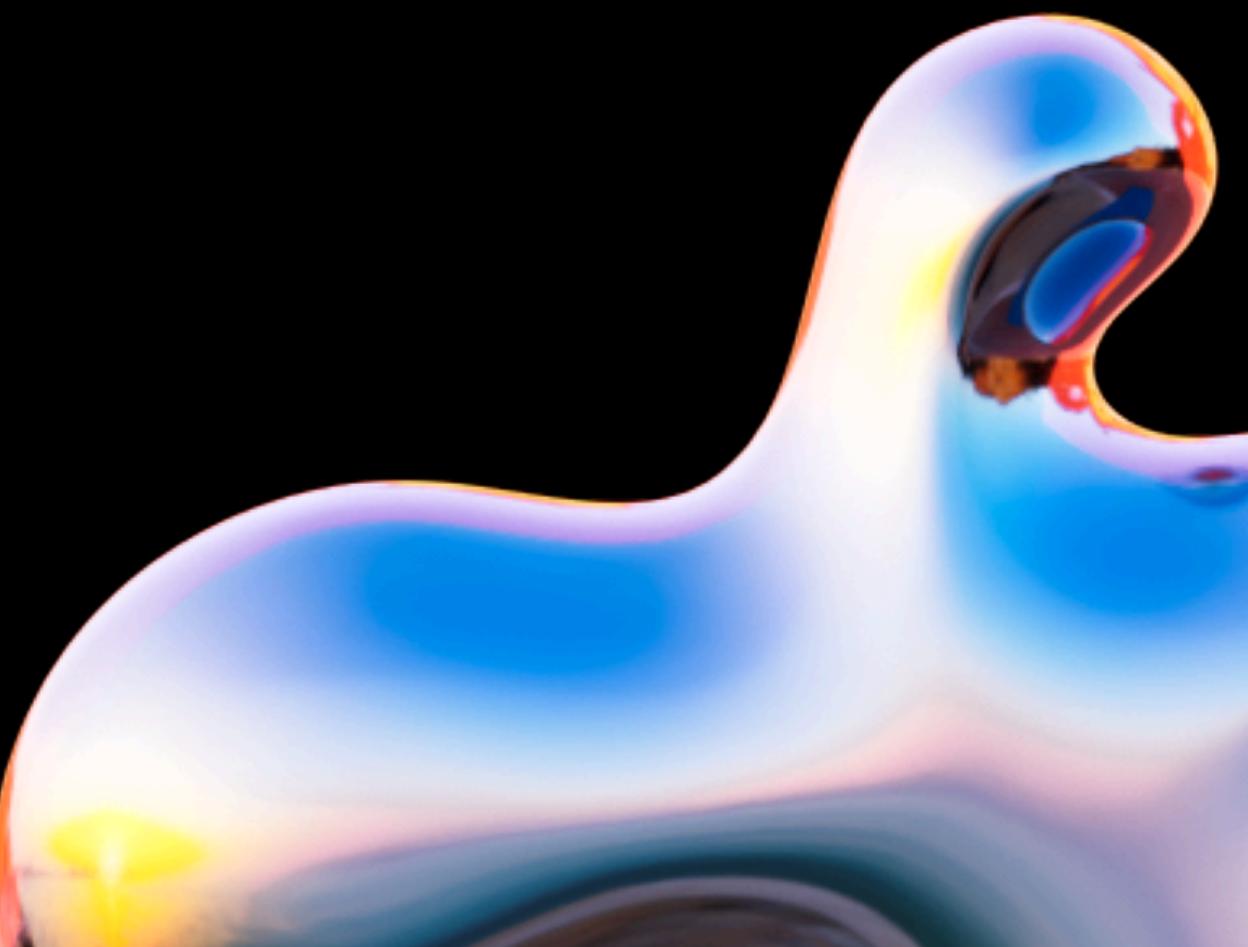
# **Step 2: Identifying Data Sources**

Satellite Datasets:

- NASA's OCO-2 (CO<sub>2</sub> measurements)
- Sentinel-5P (CH<sub>4</sub> and trace gas data)
- GHGSat (industrial emission tracking)

Model-Based Datasets:

- EDGAR (global greenhouse gas emissions)
- CarbonTracker (CO<sub>2</sub> estimations)
- TROPOMI (methane and CO<sub>2</sub> monitoring)





## **Step 3: Data Preprocessing**

- Data Access: Obtain satellite and model data in netCDF and HDF5 formats.
- Coordinate Alignment: Make sure all data adheres to a uniform spatial projection, particularly WGS84.
- Temporal Alignment: Align the timeframes of satellite and model data either daily or monthly.
- Gap Filling: Address any gaps in satellite data resulting from issues like cloud cover.

## **Step 4: Data Integration and Fusion**

- Fusion of Satellite and Model Data: Integrate datasets to enhance the accuracy of emissions mapping.
- Gridding: Resample the data into uniform grids, such as  $1^\circ \times 1^\circ$ .
- Statistical Methods: Employ Kalman filtering or analogous techniques to minimize uncertainty.

# **Step 5:Analysis and Visualization**

- GHG Concentration Mapping: Utilize GIS tools such as ArcGIS, QGIS, and Google Earth Engine.
- Emission Hotspot Identification: Employ machine learning techniques to pinpoint areas with high emissions.
- Trend Analysis: Illustrate temporal trends and seasonal patterns effectively.

# **Step 6: Validation and Calibration**

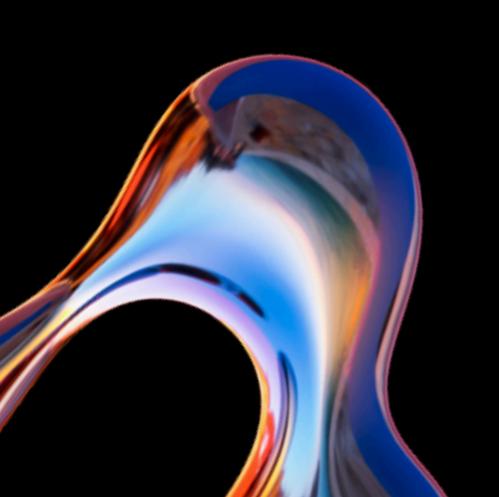
- Ground-Based Validation: Compare satellite data with ground observations, such as those from the NOAA network.
- Error Analysis: Evaluate uncertainties arising from sensor limitations and atmospheric conditions.
- Recalibration: Refine models by incorporating insights from error analysis and ground-truth data.

# **Step 7: Reporting and Visualization Tools**

- Dashboards: Develop interactive tools for stakeholders utilizing platforms such as Power BI, Tableau, or custom applications.
- Automated Reporting: Implement regular updates on greenhouse gas (GHG) trends, providing insights through monthly or quarterly reports.

# **Step 8: Forecasting and Prediction**

- Predictive Modeling: Utilize machine learning or statistical models to forecast future emissions.
- Scenario Analysis: Simulate the impact of various policies on emissions, such as carbon pricing.



# **Step 9: Collaboration and Data Sharing**

- Stakeholder Engagement: Involvement of key entities such as governments, researchers, and international organizations (e.g., IPCC).
- Open Data Platforms: Utilization of platforms like Zenodo and OpenGHGMap for public data sharing and collaborative efforts.

# **Step 10: Tools and Platforms**

- GIS Software: ArcGIS, QGIS
- Programming Languages: Python (including libraries such as rasterio, pandas, and netCDF4), R
- Cloud Platforms: Google Earth Engine, Amazon Web Services (AWS), Microsoft Azure
- Databases: PostgreSQL with PostGIS for the management of spatial data

# Conclusions

- Key Takeaway: Combining satellite and model-based data enhances GHG emissions mapping, aiding climate policy and environmental management.
- Future Steps: Focus on implementation, scaling the system, and collaborating with global stakeholders.
- This outline clearly presents each step in the system development process for a presentation.

# Thanks