# **Process Overview for Cloud Detection Using ResNet and Three-Frame Image Analysis**

### Summary of the Process:

1. **View Images and Define Intervals**:
   * **Script**: view\_images\_h\_s.py
   * **Purpose**: Inspect images, identify cloud intervals, and record them in a CSV file.
   * **Steps**:
     + Load NetCDF files and extract variables such as radiance, ISS latitude, and longitude.
     + Display images with adjustable parameters (e.g., time step, color range).
     + Use interactive features like sliders for changing the time step and color range.
     + Identify cloud intervals and note them for inclusion in a CSV file.
2. **Generate Training Data**:
   * **Script**: create\_training\_images\_3f.py
   * **Purpose**: Prepare training images from raw NetCDF files using the defined intervals.
   * **Steps**:
     + Read NetCDF files.
     + Extract and normalize radiance data.
     + Save consecutive frames as three-channel images (previous, current, next).
     + Label images based on intervals defined in the CSV file.
3. **Train the Model**:
   * **Script**: resnet\_training\_3f.py
   * **Purpose**: Train a ResNet model on the generated training images.
   * **Steps**:
     + Load image data for training and validation.
     + Preprocess the images.
     + Define the ResNet model architecture.
     + Compile and train the model.
     + Save the trained model.
4. **Prepare Prediction Data**:
   * **Script**: generate\_images\_to\_predict\_3f.py
   * **Purpose**: Generate prediction images from NetCDF files.
   * **Steps**:
     + Read NetCDF files.
     + Extract and normalize radiance data.
     + Save consecutive frames as three-channel images.
5. **Run Predictions**:
   * **Script**: predict\_all\_orbits\_3f.py
   * **Purpose**: Perform predictions and save results.
   * **Steps**:
     + Load the pre-trained model.
     + Preprocess images for prediction.
     + Predict cloud presence and calculate prediction probabilities.
     + Calculate running averages and filtered binary predictions.
     + Save predictions to CSV files, including running averages and filtered binary predictions.
6. **Add Predictions to NetCDF**:
   * **Script**: add\_mlcloud\_to\_nc\_files\_3f.py
   * **Purpose**: Add the MLCloud variable to NetCDF files based on the predictions.
   * **Steps**:
     + Read the predictions from the CSV files.
     + Extend MLCloud values to handle edge frames.
     + Read the original NetCDF files and create new ones with the MLCloud variable.
     + Save the new NetCDF files.
7. **Plot Results**:
   * **Script**: plot\_mlcloud\_from\_nc\_files\_3f\_v2.py
   * **Purpose**: Visualize the MLCloud predictions and compute confusion matrices.
   * **Steps**:
     + Extract orbit number from filenames.
     + Create binary arrays based on cloud intervals.
     + Plot MLCloud values and cloud intervals.
     + Calculate and plot confusion matrices with percentages.

### Additional Notes:

* **Three-Frame Images**: The "3f" in the script names indicates that three frames (center, 5 before, 5 after) were used to create a three-layer image.
* **Training and Prediction Orbits**: Orbits 2-60 were used for training, and predictions were applied to orbits 70, 85, 90, 105, and 130.
* **Filtering**: A filter was applied to set continuous frames with fewer than 30 consecutive '1's to '0' (the shortest interval in the CSV file was 31).
* **Accuracy Examination**: Orbit 90 was specifically examined where the accuracy was lower than the rest.
* **Current Work**: Currently working on predicting all orbits of November and writing 'MLCloud' to create new NetCDF files.

### Below are the plots and accuracy of orbits 70, 85, 90, 105, and 130:

Orbit 105:

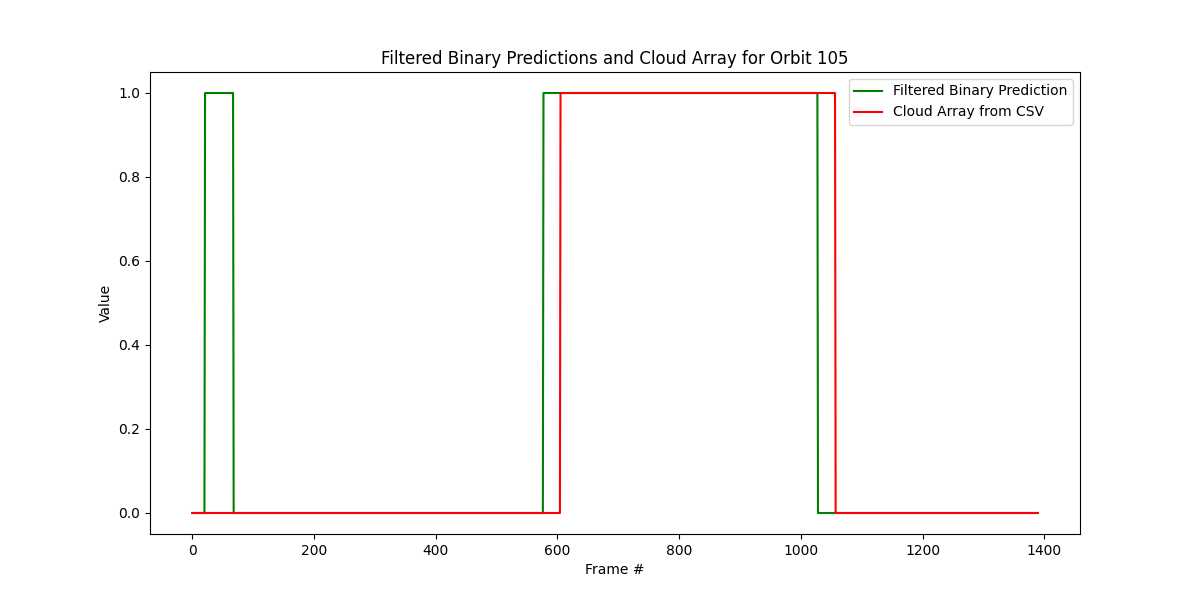
Accuracy: 92.52%

True Negatives: 863 (62.09%)

False Positives: 75 (5.40%)

False Negatives: 29 (2.09%)

True Positives: 423 (30.43%)



Orbit 120:

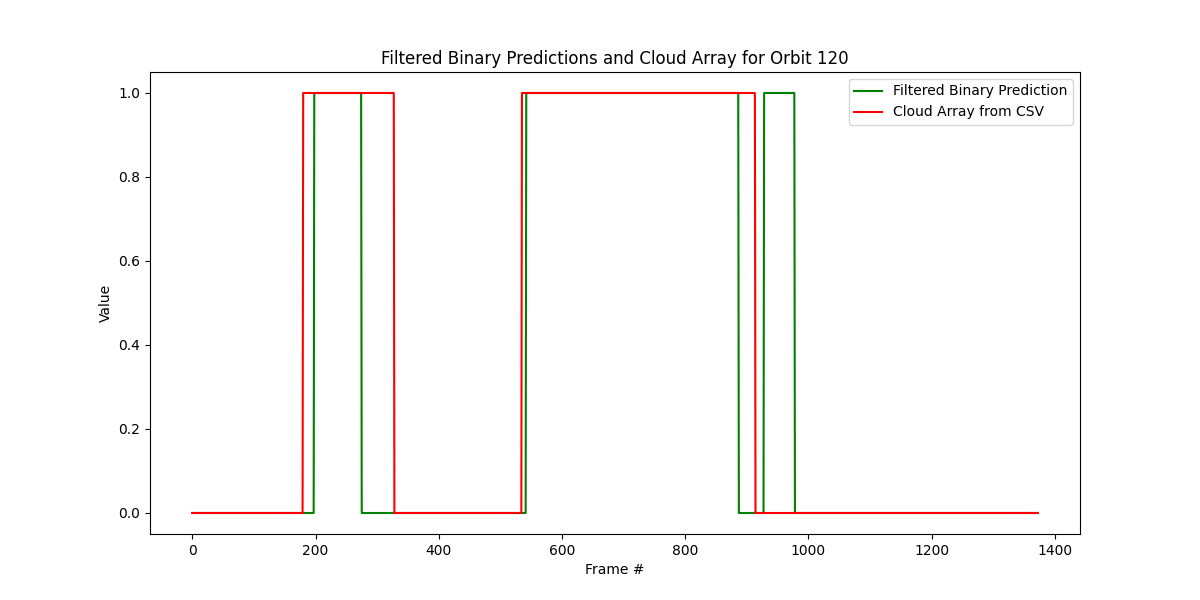
Accuracy: 88.71%

True Negatives: 796 (57.98%)

False Positives: 50 (3.64%)

False Negatives: 105 (7.65%)

True Positives: 422 (30.74%)



Orbit 135:

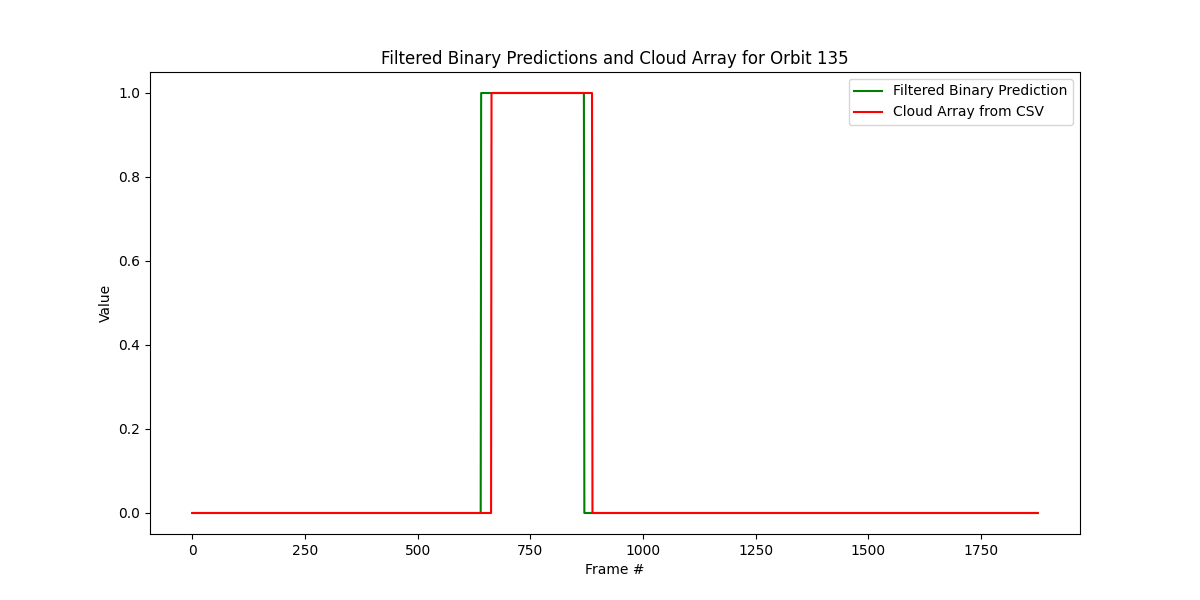
Accuracy: 97.82%

True Negatives: 1630 (86.84%)

False Positives: 23 (1.23%)

False Negatives: 18 (0.96%)

True Positives: 206 (10.97%)



Orbit 75:

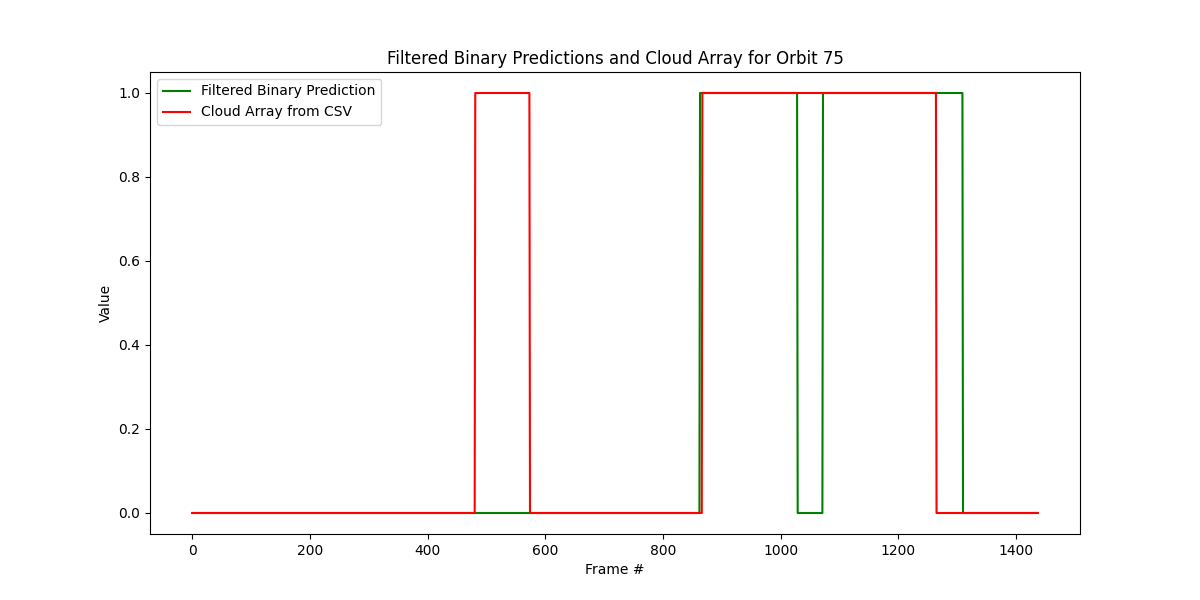
Accuracy: 87.13%

True Negatives: 898 (62.45%)

False Positives: 49 (3.41%)

False Negatives: 136 (9.46%)

True Positives: 355 (24.69%)



Orbit 90:

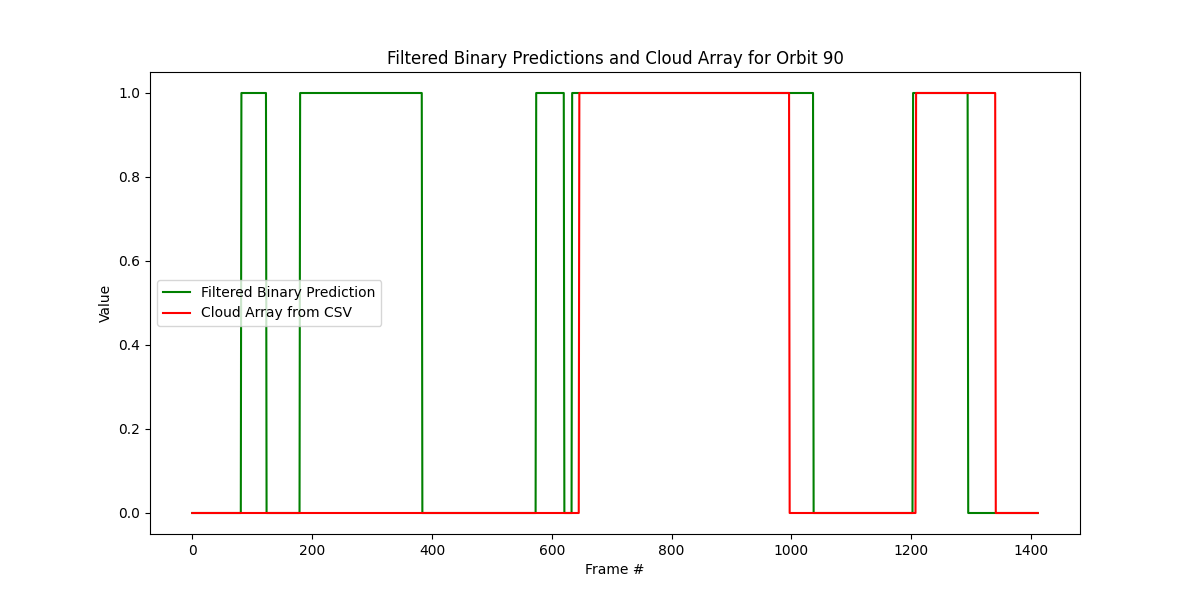
Accuracy: 71.95%

True Negatives: 578 (40.93%)

False Positives: 350 (24.79%)

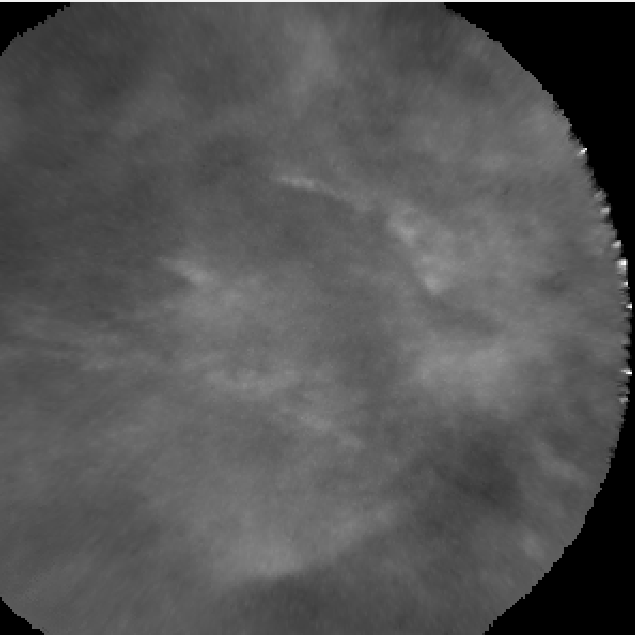
False Negatives: 46 (3.26%)

True Positives: 438 (31.02%)



Examine orbit 90 at the ‘green blocks’ where the ML model identified clouds but manually labelling did not.

Around 100: Looks like there are clouds here that were missed by us.



At 200-400: It looks pretty ambiguous, but more on the cloud side than on the clear side.

