The Classification of Tropical Storm Systems in Infrared Geostationary Weather Satellite Images Using Transfer Learning

Jacob Senior-Williams, Frank Hogervorst, Erwin Platen, Arie Kuijt, Jacobus Onderwaater, Roope Tervo, Viju O. John, and Arata Okuyama

***Abstract*—The work performed in this study evaluated the application of generalized pretrained object detection models for the identification and classification of tropical storm (TS) systems through transfer learning. While the majority of literature focuses on developing bespoke models for this application, these typically require significantly more training data, compute resources, and time to train the models due to the large number of parameters the model has to tune to achieve similar results. These models also required additional preprocessing steps, such as extracting the storm from the image, and used a limited number of classes to describe the intensity of the storms. The approach presented here used considerably less data than the majority of other work (2–10x fewer input images) and a larger number of classes. The accuracies of the produced models trained on four different experimental datasets (varying the amount of data and number of classes) through this approach were 75%, 82%, 69%, and 89%. Overall, the models produced promising results, performing approximately equal to the bespoke models with scope to improve the performance of the model.**

***Index Terms*—Machine learning (ML), remote sensing, transfer learning, tropical storm (TS).**

I. Introduction

UNDERSTANDING the evolution of tropical storms (TS) in a changing climate is important for climate adaptation as these storms are some of the most extreme weather events and have a huge socio-economic impact. Globally, there are about 86 TS per year based on the data for the last 43 years (1980–2022) with an interannual variability of nine storms. During this period, the maximum number of storms occurred in 2020 with 104 storms and the minimum in 2010 with 68 storms. The satellite data archive that spans over more than four decades allows us to produce a long-term TS dataset including classification of storms and their intensity, which has applications, for example, in investigating climate trends such as the rapid intensification of TSs in a warming climate [1].

The objective of this work was to create a system that could accurately label TS contained in historical data of geostationary weather satellites without external data sources, which may be incomplete. The system could then process all historical data from any satellite and determine if any TS are present in any image and, if so, estimate the intensity. To minimize the requirements for the model to be put into a production environment, the aim was to use an approach that required the minimal amount of preprocessing of the data, computational power, and data possible. In the end, this system would enable climate researchers to have a more complete set of image data of TS so that they could perform their research more effectively. It would also provide some insight into the TS data that is potentially missing from the dataset containing the evolution and track data for historical TS.

The work performed investigated the use of pretrained object detection machine learning (ML) models for the classification of TS by their intensity as recorded in historical data from geostationary weather satellites. If the models developed in this work performed with similar or better accuracy than those of other approaches in the literature, it could allow these very valuable models to be produced with significantly lower barriers to entry in terms of the data, compute resources, and compute time required to train them, which would enable faster iterations to improve the current models and allow more people to be able to train.

A. Background

Throughout the literature, there were generally two approaches taken. The first was to use the images and classify the TS into individual classes based upon one of the established TS intensity scales, and the second was to attempt to infer the wind speed of the TS by approaching the problem as a regression task. Due to the utilization of transfer learning implemented in this research, only the literature that used classification was investigated for the literature review. This is due to the fact that pretrained models used for transfer learning require the new task to also be a classification problem.

For the work performed using classification, Jiang and Tao [2] achieved a classification accuracy of 97.12% and precision

Manuscript received 25 September 2023; revised 22 November 2023 and 22 January 2024; accepted 30 January 2024. Date of publication 14 February 2024; date of current version 28 February 2024. This work was supported by EUMETSAT and in part by JMA. *(Corresponding author: Jacob Senior-Williams.)*

Jacob Senior-Williams, Frank Hogervorst, Erwin Platen, and Arie Kuijt are with the Science and Technology B.V, 2616 LR Delft, The Netherlands (e-mail: jacob.senior@stcorp.nl).

Jacobus Onderwaater, Roope Tervo, and Viju O. John are with the EUMETSAT, 64295 Darmstadt, Germany.

Arata Okuyama is with the Japan Meteorological Agency, Tokyo 105-8431, Japan.

Digital Object Identifier 10.1109/JSTARS.2024.3365852

© 2024 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

of 97.00% using a bespoke model predicting six classes. 13 200 cloud images were used and cropped to a resolution of 512×512 pixels and containing only the TS. Zhou et al. [3] trained a model predicting four classes achieving a 92.35% accuracy, with approximately 3500 training samples and 600 test samples. While Gardoll and Boucher [4] achieved over 99% accuracy with their model, it was only performing binary classification (TS or no TS), which limits the usefulness of the model output and used 28 521 images. Zhang et al. [5] predicted three classes and their novel model architecture achieved approximately 80% accuracy. Their dataset contained 19 451 TS images, which was increased to 36 957 after a number of data augmentation processes were applied. Wang et al. [6] also utilized data augmentation to increase their initial dataset from 6690 images to 22 746. The authors also used a bespoke model containing predicting eight classes and achieved an accuracy of 89% for their final system architecture. This was composed of eight separate models, all performing binary classification for one specific storm class and the model with the highest confidence was used as the storm classification. Kar et al. [7] used only three storm classes and their approach yielded a model with 84% accuracy, although the number of images used was not specified.

What sets apart the research described in this article from the discussed pieces is that all of them focus on developing a novel model architecture for a specific task, requiring significantly more computing resources, time, and data to train the model. In the models developed in the literature typically used fewer classes for the TS intensity than the approach described in this article, and they generally require the TS in the images to be cropped such that the image solely contains the TS to be classified, rather using the whole image. Using the whole image is advantageous because if the model were to be applied in a near real time setting, the location data for TS may not exist yet, which would complicate the inference process as the system may not be able to identify the TS in the image. Another advantage is that fewer preprocessing steps yielding a pipeline is more efficient and would result in less time to get a prediction from an image.

B. Novel Contributions

1) The application of off-the-shelf pretrained models on the infrared channels of geostationary weather satellites.
2) Demonstrating minimal preprocessing of the data was required when compared to methods employed in the literature.
3) The use of the Dvorak intensity scale, while all other pieces of literature used either RMSC Tokyo or Saffir-Simpson.
4) The implementation of the F1+rmse model performance metric, which accounts for the fact that the classes are not discrete, unlike classes in most classification tasks.

II. Materials and Methods

A. Satellite Data

For this work, 1803 images from the JMA's Geostationary Meteorological Satellites (GMS) 1–4 were used, containing 4526 TS. These satellites were positioned at 140° East longitude in geosynchronous orbit with an altitude of approximately 36 000 km above the equator in line with Japan and in operation from 1977 until 2000 [8].

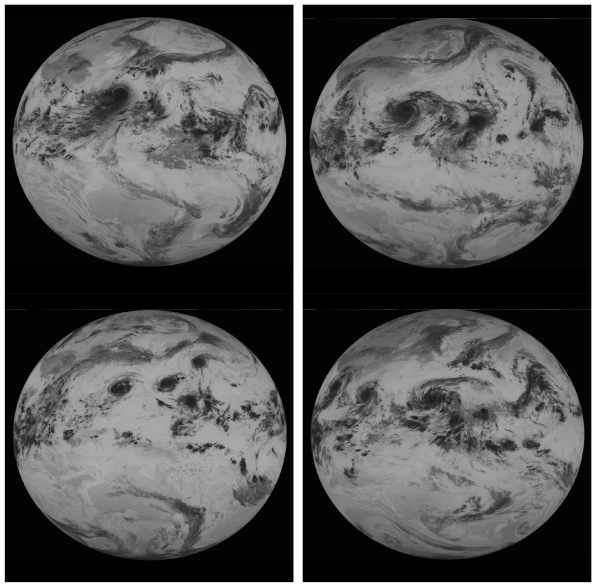

Fig. 1. Examples of images used as input data. Each image is from a different GMS Satellite and shows the consistency of the data across the different generations of satellite.

The data used consist of images taken every three hours for the three same days of each month (15th, 16th, and 17th) from 1979 until 1995. The images used were all from the infrared sensor, and as such, are single channel grey-scale images, as seen in Fig. 1.

B. IBTrACS

To identify the TS in the images and determine their intensity, the International Best Track Archive for Climate Stewardship (IBTrACS) was used [9]. This is a tabular data file containing the most complete global collection of tropical cyclones available. The data include the TS names, coordinates, wind speeds, and pressures every three hours for storms dating back to 1842. The data are generated by combining TS track datasets from various agencies across the world.

C. Intensity Scales

There are a number of different intensity scales used worldwide for classifying the intensity of TS. The one most often used Western Hemisphere is the Saffir–Simpson Hurricane Wind Scale (SSHWS), developed by the National Hurricane Center (NHC). This scale is typically used for TS in the Atlantic and the Central and Eastern Pacific. For the Western Pacific, the RSMC Tokyo scale is used. There are separate scales for the Northern and South Western Indian Ocean, and one for Australia and Fiji. The main issue with these disparate scales is that they all use different wind speeds for different TS severity levels, and what could be a Typhoon on the RSMC Tokyo scale could range from a Category 1 to Category 3 on the SSHWS as can be seen in

T-Number	1-min Winds (knots)	(mph)	(km/h)	Category (SSHWS)	Min. Pressure (millibars) Atlantic	NW Pacific	Category (RSMC Tokyo)
1.0 – 1.5	25	29	45	below TD	----	----	Tropical Depression
2.0	30	35	55	TD	1009	1000	
2.5	35	40	65	TS	1005	998	Tropical Storm
3.0	45	52	83	TS	1000	991	
3.5	55	63	102	TS-Cat 1	994	984	Severe Tropical
4.0	65	75	120	Cat 1	987	976	Typhoon
4.5	77	89	143	Cat 1–Cat 2	979	966	
5.0	90	104	167	Cat 2–Cat 3	970	954	Very Strong Typhoon
5.5	102	117	189	Cat 3	960	941	
6.0	115	132	213	Cat 4	948	927	Violent Typhoon
6.5	127	146	235	Cat 4	935	915	
7.0	140	161	260	Cat 5	921	898	
7.5	155	178	287	Cat 5	906	879	
8.0	170	196	315	Cat 5	890	858	
8.5†	185	213	343	Cat 5	873	841	

Fig. 2. Modified DCI scale from [11] to include RSMC Tokyo scale, showing the relationship between wind speed, aAir pressure, and two different TS classification systems.

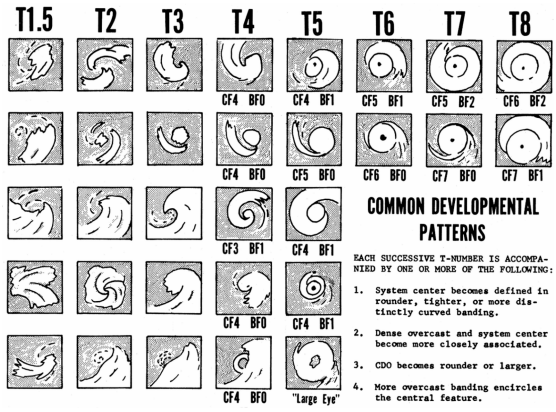

Fig. 3. Common developmental patterns diagram showing typical TS shape for different DCI T-numbers [12]. The cloud structures were used to aid in the creation of the DCI scale.

Fig. 2. The scale selected in this project was the Dvorak Current Intensity (DCI) for three reasons. First, it has a higher intensity scale resolution (increasing from 1–8 in 0.5 steps) than other hurricane intensity scales, therefore has the highest probability that the classification would translate into other distinct intensity levels on multiple other scales. Second, a storm's intensity level can be inferred from the DCI using wind speed or minimum pressure, therefore, if one of the pieces of data is missing from IBTrACS, the other can be used instead, where other scales only use wind speed to derive the storm's intensity class. Lastly, the DCI was developed specifically using visible and infrared satellite images [10]. Its classes also take into account the TS structure based on the common development patterns chart in Fig. 3. This was expected to increase the likelihood that the model would be able to classify the storms with greater accuracy.

D. Machine Learning

Over the past decade, ML techniques, such as deep learning, have repeatedly proven to excel at image-based tasks, particularly identifying and classifying objects within images. One powerful technique that has been developed is called transfer learning. This method significantly reduces the data, compute resources, and time required to train a deep neural network. The model has already been "pretrained" on hundreds of thousands, if not millions, of RGB images containing a variety of objects such as various animals, modes of transport, and every day objects. The layers of the model have then learned to extract a wide variety of features from images and can classify a wide range of different objects within images. Transfer Learning takes this model, locks the vast majority of the weights and only requires the final few layers to be tuned to transform the very generalized pretrained model to a specialized one for the desired task.

One of the significant advantages of using transfer learning is the ability to train a variety of different models with little effort. Three different models were trained for each of the datasets discussed in the next section. The models used for this work were FasterRCNN Resnet50 V1, FasterRCNN Inception V2, and SSD Resnet50 V1. The selected models were chosen for a number of reasons. Although the first model has the lowest mean average precision (mAP) for a model of its input resolution, and one of the oldest pretrained models available, it is relatively lighter in terms of computational requirements compared to the other models. Therefore, it was faster to train and establish a baseline model performance. Object detection models need to perform two tasks. The first is to identify areas of interest within an image and the second is to classify these areas of interest into discrete categories. A model with a region proposal network (RPN), performs this task in two steps, whereas other architectures like single shot detectors (SSD), do this in one step directly using a feature map. The second model was the same architecture as the first but used a different classifier (Inception). The final model was an SSD model, to evaluate the performance difference between the two architectures for this task. The models were pretrained on the Microsoft Common Objects In Context (COCO) dataset, achieving a mAP of 31.0, 38.7, and 38.3, respectively [13]. While other models are available to use at [13], some of which potentially could perform better than the three selected, but they were too large to load into the memory of the GPU used for this work.

E. Experimental Design

1) Data Preparation: The initial images were reshaped and down-sampled from their native resolution of 2451×6686 pixels to 1024×1024 pixels. This is because pretrained models have a fixed input size corresponding to the original training data that were used. This was the only processing that was performed on the images.

The labels that identify the storms in the images were prepared into individual XML files for each image in a specific format called PASCAL VOC. This format contains meta-data, both about the image (filename, path, directory) and the objects contained in the image itself (object class, object location within the image given by bounding box). 4,526 TS were found in the IBTrACS dataset. For each TS, the center pixel of the system was acquired and then approximate bounding boxes were created, 50 pixels in each direction from the center point (i.e., if the center

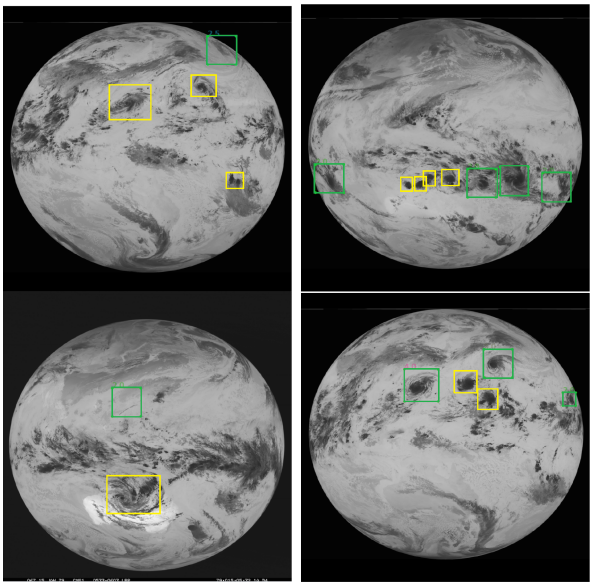

Fig. 4. Images with possible storms missing in IBTrACS. Green boxes denote the TS found in IBTrACS, Yellow boxes denote possible TS that were not recorded in IBTrACS.

pixel was pixel 200, 200, the bounding box top left corner was 150, 150 and the bottom right was 250, 250).

After the files were generated, the data were validated to ensure that the labels appeared to be correct. This was performed manually using the open source software, LabelImg [14]. Some minor modifications were made to some bounding boxes to better encapsulate the TS structure within it. During this process, it became apparent that IBTrACS is the most complete database of storm systems publicly available, but far from perfect. There were numerous examples of potential TS present in the images, with no data present in IBTrACS. Examples can be seen in Fig. 4.

Similarly, there were bounding boxes that appeared to contain no TS; however, the labels were present in multiple sequential images. The authors believe this could be attributed to a combination of the angle of the TS to the sensor (as they typically occurred toward the extremity of the Earth disk) and relatively light cloud structures. No labels were added or removed from the dataset.

2) Experiments: Four experiments were conducted during work. The first used all data and the full range of DCI levels. The second reduced the number of DCI levels to whole integers, rather than steps of 0.5. TS with a 0.5 intensity level were combined into the lower whole integer class (i.e., a 3.5 TS became a 3.0 TS). The purpose of this experiment was to evaluate whether the accuracy of the model increases with broader categories, as the visual difference between TS of close intensity (e.g., 3.0 and 3.5) may not be significant enough to reliably differentiate between them.

The third and fourth experiments had the same two groups previously described, but the storms of intensity below 3.0 were discarded from the dataset. These low-intensity TS account for roughly half of the storm systems in the dataset, but are not posing significant danger to human life. The potential consequence of the model performance is that it may correctly identify a TS

TABLE I
TABLE SHOWING THE NUMBER OF SAMPLES CONTAINED IN EACH OF THE CLASSES FOR EACH EXPERIMENTAL DATASET

TS intensity	All data 0.5 steps	All data 1.0 steps	Reduced data 0.5 steps	Reduced data 1.0 steps
1.0	771	771	0	0
2.0	774	1146	0	0
2.5	672	0	0	0
3.0	837	1193	837	1193
3.5	356	0	356	0
4.0	510	853	510	853
4.5	343	0	343	0
5.0	97	184	97	184
5.5	87	0	87	0
6.0	47	71	47	71
6.5	24	0	24	0
7.0	8	8	8	8
7.5	0	0	0	0
8.0	0	0	0	0
Total	4526	4526	2309	2309

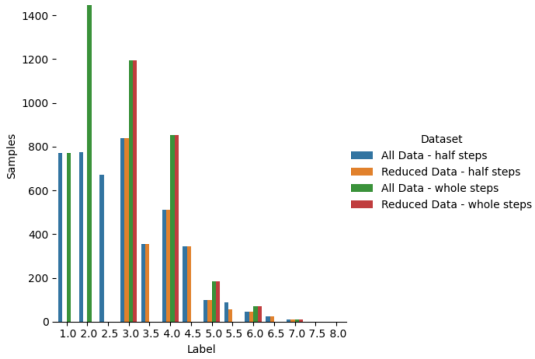

Fig. 5. Graph showing the number of samples contained in each of the classes for each experimental dataset. The imbalance in the number of samples in each class is clearly visible with many more lower intensity TS present compared to higher intensity TS.

of intensity 2.0/2.5, but misclassify it as a 3.0 TS. A correctly identified TS would be treated as an error and lead to a lower accuracy. The number of TS per class and dataset can be seen in Table I.

For the datasets using all of the storm systems, 2649 images were used, containing 4526 TS. In the datasets with TS of intensity greater than 3.0, 1803 images were used, which contained 2309 TS. The distribution of the classes for the datasets can be seen in Fig. 5.

The most notable aspect of Fig. 5 is that there is a significant imbalance toward the TS of weak intensity. Logically, this is unsurprising, as TS of T-number 2.0 are far more common than TS with T-number 6.0. The issue this presented is that an imbalanced dataset can lead to ML models becoming biased in their predictions toward labels with more samples in the training data. In a structured data problem, such as numerical data in an excel file, the solution is to evenly sample the data such that it becomes balanced. However, this is not the common solution for an object detection problem as it is more difficult due to single images containing multiple, different TS with different intensities. Instead, a specific loss function must be used such as focal loss, which is a dynamically scaled cross entropy loss. This down-weights the loss associated with individual classes as the

confidence in predicting the classes increases. This allows the model to learn each class equally, regardless of imbalances [15]. This was used for all models in this work. Finally, the datasets were partitioned into three subsets: training, validation, and test sets with an 80:10:10 split, respectively.

F. Model Performance Evaluation

To determine the model performance, accuracy, precision, recall, and F1 score were used during training. These are defined by the following:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN}.$$

One challenge with using these metrics for this task is their treatment of all classes as discrete and unrelated. However, in this context, the classes are closely related, where the distance between classes corresponds to a linear scale of different intensities of a TS. Classes nearby each other (e.g., 2.5, 3.0, 3.5) are more similar than those further apart (1.0, 3.0, 5.0). The significance of misclassifying a TS as a neighboring class is less than for classes far apart.

This is, in some respects, a regression problem as the classes are a continuous number range with discrete increases. Therefore, rmse was used to evaluate the quality of the different model predictions and is defined in the following :

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\left(\frac{d_i - f_i}{\sigma_i}\right)^2}.$$

RMSE is a commonly used metric for regression problems as the distance between the true and predicted value is taken into account, with larger errors contributing more than smaller errors [16]. For other approaches that implemented classification ([2], [3], [4], [5], [6], [7]) and used TS intensity scales with phrases for the class labels (e.g., Typhoon, Very Strong Typhoon, Violent Typhoon), this type of analysis was not possible.

For the rmse to be calculated, the false positives and false negatives needed to be accounted for. Initially, they were included as belonging to class "0.0" for the true labels and predictions, respectively. False negatives remained in the dataset with the same values, the true value of the storm missed would be used and 0.0 would be used for the predicted value. This reflected the fact that the stronger the TS missed and would result in a greater increase in rmse, as the more severe the intensity of the TS not detected, the greater and more significant it is that the model missed that TS. Due to the previously discussed concerns regarding the completeness of IBTrACS, false positives were treated as misclassifications of one class lower, such that they are still represented in the rmse, but their impact on the rmse is not substantial. As a higher rmse represents less quality

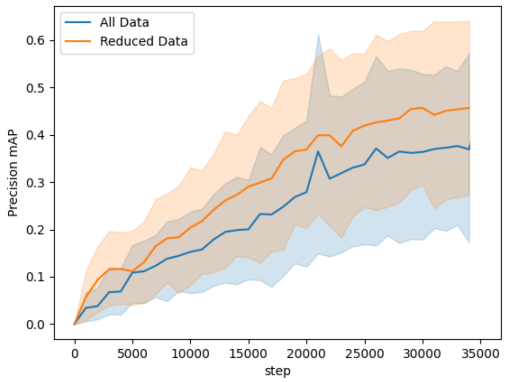

Fig. 6. Average and range performance of all models on validation datasets with all TS (all data) and without lower intensity TS (reduced data). Models without the lower intensity TS performed better than those with all TS.

predictions, a method was devised to combine rmse and F1 score by multiplying the F1 score by the 1 – rmse. The higher this value is, the higher quality the model outputs are. Formally, it was calculated using the following:

$$F1\&RMSE = F1 * (1 - RMSE).$$

This model performance metric would not have been possible to implement without using the DCI scale as other intensity scales are qualitative scales rather than numerical.

G. Transfer Learning Strategy and Model Training

The following procedure was used to perform transfer learning:

1) install Tensorflow object detection API;
2) annotate dataset as described in experimental design;
3) split datasets into training, validation, and test datasets;
4) transform datasets into TF records for data pipeline optimization;
5) download and configure pretrained model;
6) model training and evaluation.

A more complete description of the process can be found at [17]. The models were trained on a computer running Windows 10 with an Intel i7-9700 K CPU @ 3.60 GHz, 64 GB RAM, and an NVIDIA RTX 2070 Super GPU. The models were trained until the performance of the model on the validation data began to plateau to avoid overfitting, which was typically around 30 000 training steps.

III. Results

The results of the model training on validation data are as follows. Fig. 6 shows the average performance of all models on each amount of data (all data or only storms of intensity 3.0 and above) with the line, and the shaded area represents the spread of the model performance. Fig. 7 shows performance of models with different labels (half or whole intensity steps). These graphs show that, overall, the reduced dataset and whole intensity steps do perform better than all of the data and half intensity steps on average, there is a significant overlap between the range of performance for all models. While it is logical that less data and fewer classes would make the classification task easier for

TABLE II
TABLE SHOWING THE PERFORMANCE METRICS OF THE FRCNN INCEPTION MODEL ON THE VALIDATION DATA FOR EACH OF THE EXPERIMENTAL DATASETS

Dataset	Accuracy	Precision	Recall	F1 Score	RMSE	F1 & RMSE
All Data, 0.5 Steps	0.75	0.74	0.75	0.74	0.460	0.410
All Data, 1.0 Steps	0.82	0.81	0.82	0.81	0.522	0.397
Reduced Data, 0.5 Steps	0.69	0.67	0.69	0.67	0.918	0.059
Reduced Data, 1.0 Steps	0.89	0.86	0.89	0.88	0.601	0.363

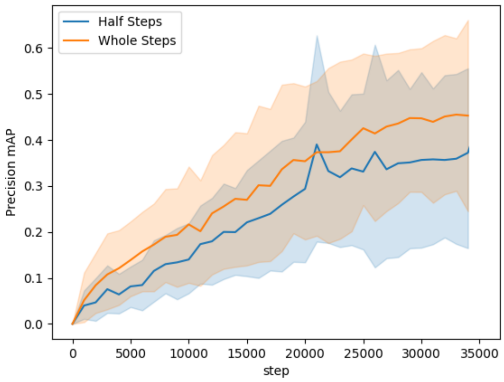

Fig. 7. Average and range performance of all model on validation datasets with label intensity increases of 0.5 (half steps) and 1.0 (whole steps). Models using labels with 1.0 intensity increases outperformed the models using labels with 0.5 intensity increases.

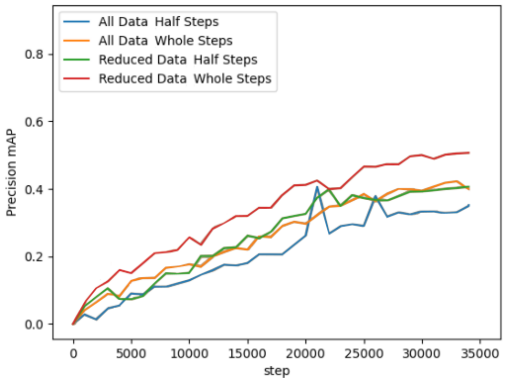

Fig. 8. Average performance of all models on each of the different experimental validation datasets. The models using the simplest dataset (no low intensity TS and labels with 1.0 intensity increases) performed the best.

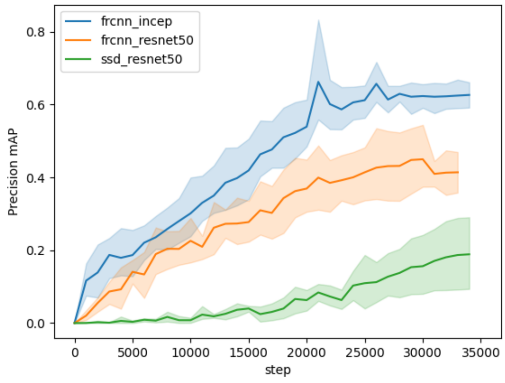

Fig. 9. Average and range of performance for different models across all four validation datasets. FRCNN inception model outpeformed both of the other models with the worst FRCNN inception model still performing better than the next best.

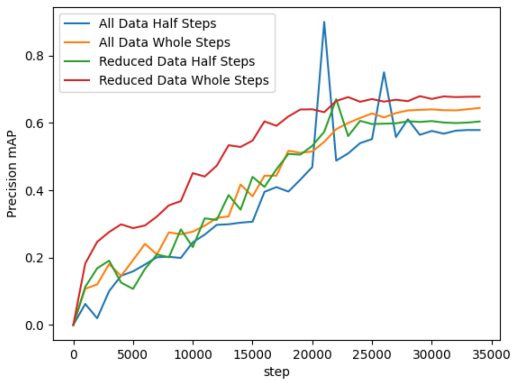

Fig. 10. Performance of the FRCNN inception model on each of the different experiment validation datasets. The model trained on the simplest dataset (no low intensity TS and 1.0 label increases) did perform the best, however there is not a large difference between all the models.

the model to learn, it is clear that there was a large variation between different models. Fig. 8 shows the amalgamated model performance on the different experiment datasets. From this figure, it is clear that the reduced dataset with whole intensity step labels is the best-performing dataset overall. The reduced dataset with half-intensity step labels and whole dataset with whole intensity step labels produce similar results. The most difficult dataset for the model to learn was the whole dataset with half intensity step labels being the experiment with the overall worst produced models.

From Fig. 9, it is clear that the FRCNN inception model is the best performing, FRCNN ResNet 50 is the second best, and SSD ResNet50 is the worst, with the lowest final mAP and most variation between datasets. Taking this into account, the remainder of the result evaluation will only consider the FRCNN inception network, as even on the most difficult dataset, it outperformed the next best model's performance on the easiest dataset.

Each model was evaluated on the test data, producing the precision, recall and F1 scores as seen in Table II. The scores provided are the weighted average, which takes into account the imbalance of the labels in the datasets. Subsequently, the rmse and F1 and rmse combined score were calculated.

The confusion matrices (see Figs. 11–14) were generated from the results of the FRCNN inception model predictions on the test data. As specified earlier, the 0.0 row and columns represent false positives and false negatives in the actual and predicted axis, respectively.

Finally, Table III shows the performance of the models trained in this work compared to those found in the literature.

TABLE III
TABLE COMPARING THE MODELS IN THE LITERATURE, WITH THE NUMBER OF IMAGES, NUMBER OF LABELS USED, AND ACCURACY, PRECISION, AND RECALL WHERE AVAILABLE

Model	#Images	#Labels	Accuracy (%)	Precision	Recall
Jiang and Tao [5]	13 200	6	97.12	97.13	–
Zhou, Xiang and Huang [17]	4 100	4	92.35	–	91.04
Gardoll and Boucher [4]	28 521	2	>99.00	–	–
Zhang et al. [16]	36 957	3	80.49	77.00	74.00
Zhang et al. [16]	36 957	5	–	60.00	62.00
Wang et al. [14]	22 746	8	89.00	89.00	89.00
Kar, Kumar and Banerjee [6]	Unknown	3	84.00	–	–
FRCNN Inception - All Data, 0.5 Steps	2 649	14	75.00	0.74	0.75
FRCNN Inception - All Data, 1.0 Steps	2 649	7	82.00	0.81	0.82
FRCNN Inception - Reduced Data, 0.5 Steps	1 803	14	69.00	0.67	0.69
FRCNN Inception - Reduced Data, 1.0 Steps	1 803	7	89.00	0.86	0.88

Bold Entries are models that were produced in this work.

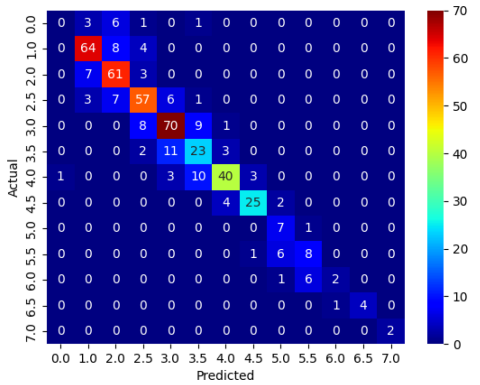

Fig. 11. Confusion matrix of FRCNN inception model trained on all data with 0.5 intensity step labels. Values are the classifications produced by the model (predicted) compared to the corresponding label in the validation dataset (actual).

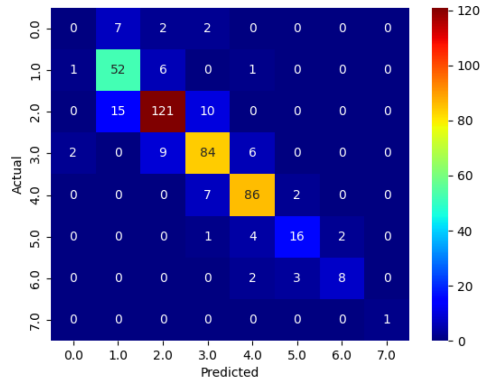

Fig. 12. Confusion matrix of FRCNN inception model trained on all data with 1.0 intensity step labels. Values are the classifications produced by the model (predicted) compared to the corresponding label in the validation dataset (actual).

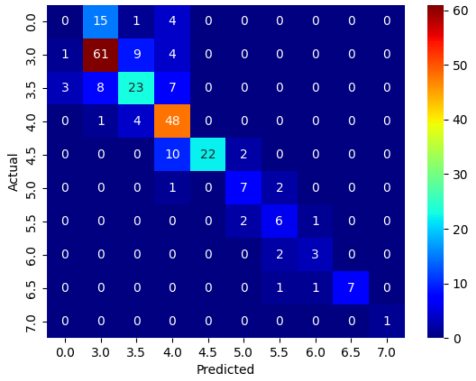

Fig. 13. Confusion matrix of FRCNN inception model trained on reduced data with 0.5 intensity step labels. Values are the classifications produced by the model (predicted) compared to the corresponding label in the validation dataset (actual). Classifications in the top row (0.0) likely to be false positives of lower intensity TS that were excluded from the training data.

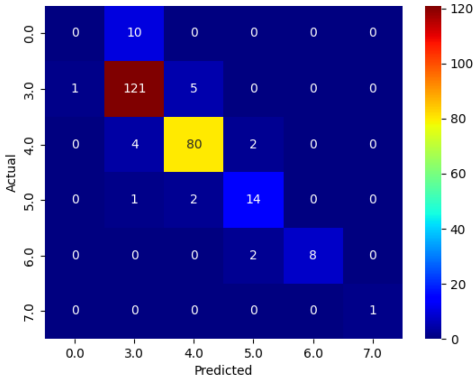

Fig. 14. Confusion matrix of FRCNN inception model trained on reduced data with 1.0 intensity step labels. Values are the classifications produced by the model (predicted) compared to the corresponding label in the validation dataset (actual). Classifications in the top row (0.0) likely to be false positives of lower intensity TS that were excluded from the training data.

IV. DISCUSSION

Before taking rmse into account, as was expected from Fig. 10, the dataset that led to the model with the best classification performance on the test set was the one with only the more intense storms and whole intensity steps. The second-best performing dataset contained all data and whole intensity steps, which also is logical as having fewer classes, with a bigger visual difference between them is likely to lead to a better performing model as its task is easier. Interestingly, the third best model was trained on all data and half-intensity steps, where it may have been expected to be the model trained on reduced data and half-intensity steps.

RMSE was used as a method to evaluate the quality of the predictions of the models, and these results suggest that the model trained on all of the data and half integer steps actually produced the highest quality results. This is likely due to the fact that the misclassifications were less severe as the majority

of them were in neighboring classes (± 0.5 away from the true class) rather than 1.0 in the models trained using whole integer steps. The model trained on the reduced data and half integer steps had the most false positives and false negatives (20 and 4, respectively), which likely contributed to it attaining the highest rmse.

One interesting, albeit expected, result of removing the lower intensity storms was an increase in the number of false positives produced by the models despite having roughly half the number of data samples. This is visible in Figs. 13 and 14. The majority of the false positives for models trained with the reduced datasets were TS of intensity 3.0, which are likely to be TS of intensity 2.0 or 2.5 that were in images that also contained TS of intensity 3.0 or greater. The model did technically detect valid TS, however, as they are outside of the range of TS intensities used for the model they are false positives.

When comparing the confusion matrices of models trained on half-intensity steps to whole intensity steps, the half-step trained models made fewer misclassifications of comparable severity than that of the model using whole intensity steps. Therefore, while there are more misclassifications of storms using the half-step intensity increases compared to the same sized datasets with the different labels, the severity and potential consequences of these misclassifications are much less.

The most surprising difference in model performance was the disparity between the FRCNN Inception V2 model and the SSD ResNet 50 model [13], they both reportedly have very similar performance on the standard test dataset used a mAP of 38.7 and 38.3, respectively. As described in the ML section, there is a fundamental difference between how these two models identify different areas of interest within an image. The FRCNN models a two stage approach where region proposal network first proposes different regions and types of objects within those regions, and then a separate component performs the classification and bounding box calculations for those regions. Models, such as SSD, use a feature map to do this in one single pass. The difference in performance of these models could be explained by the fact that the data used to train these models were 3 channel RGB images. As the feature maps of 3 channel images have significantly more variation than single channel images, and the feature map used by the SSD models when applied to single channel black and white images may not be varied enough for the model to be able to identify the TS in the images accurately in a single pass. As the FRCNN models have a dedicated component specifically for this task, it is better able to interpret the image and determine where the storms are located to classify them. This would also explain why FRCNN ResNet 50 outperformed the SSD ResNet 50 model, despite having a significantly worse mAP on the test dataset (31.0 compared to 38.3).

Table III shows the difference between the models found in the literature and the models produced in this research. The difference in amount of data used to train the models ranged from 1.55x to 13.95x when the FRCNN Inception models were trained on all data and between 2.27x and 20.49x for the FRCNN inception models trained on the reduced data. While only the model in [6] was trained with more classes than any of the models produced in this research (8 labels versus 7 labels), the model accuracy was identical and Wang et al. [6] trained eight separate binary classification models to achieve this performance and needed 10x more data. While the best performing model in the literature did outperform the models produced via transfer learning, they also required the TS to be extracted from the whole satellite image such that the input images to the model only contained the storm to be classified. This may not be a feasible approach if the models were to be used in a production environment as the information to determine where a storm is may not exist for this task to be performed.

With only eight samples available for the strongest storm in the dataset (T-number 7.0), there were concerns regarding whether the focal loss function would be able to deal with such a significant imbalance. This class is accounts for only 0.17% of the whole dataset. While the model was able to successfully classify these storms in the validation data, a sample size of only eight means that it is difficult to determine if this is truly representative of model performance.

For ML models, the quality of the datasets used is the most important contributor to model performance. There are a few ways that the quality of these datasets could be improved.

1) With access to more image data from more complete GMS satellite datasets, not just the 15th, 16th, and 17th of each month, and from other geostationary weather satellites), it would be possible to greatly improve the balance of the dataset labels.
2) While this research demonstrated that it is possible to attain good results with a limited dataset, increasing the size of the data will likely increase the robustness of the model, as a greater variety of TS could be included. This would also alleviate a potential issue with current dataset which is that due to the fact that there many images that contain the same storm systems at different times with little change in its structure or shape and this could artificially enhance the model performance.
3) Limit the area of interest that TS can be labeled in to the center of the Earth disk, excluding the areas are toward the extremities. This is because toward the edge of the Earth disk, the angle between the TS and the sensor is relatively shallow and meaning that much of the distinguishing structure of the TS is lost. By limiting the storms in the image to those that are in the center of the frame, it is ensured that the TS structure remains clearly visible in the image. This should not be implemented as an image preprocessing step by cropping the image, instead when the labels are generated, TS that are positioned outside a certain area would be excluded.
4) Incorporate data from additional satellites covering different areas of the planet. This would not only increase the size of the dataset, but would also include the TS excluded in the previous suggestion while also better preserving the distinguishing structure of these storms compared to the current implementation which should increase the model's ability to classify them.
5) To reduce the number of false positives produced by the models. As discussed in Section II-E, the data source used to extract the storms from the images (IBTrACS) is not

perfect, and there were a number of what appeared to be TS systems that no data was found for. It is difficult to determine whether these cloud systems were actually TS or not. It is feasible that lower intensity storms that formed and decayed solely over water were not recorded. One potential solution with access to more data would be to train a model on more recent data that there is a higher degree of certainty about its completeness and then apply the model to older data.

6) Of the models available in [13], generally, the higher the input resolution of images, the higher the mAP. This is not surprising as high resolution images contain more information within them and this could be a relatively easy method to increase performance.

Regarding improvements to the model, in Fig. 10 there is a clear and significant spike in mAP at approximately 21 000–22 000 steps and another smaller spike at approximately 25 000 steps for the model train on all data and half-intensity steps. At these points the model outperforms all other models. After training had completed, this model checkpoint had been overwritten and therefore this weight configuration was not usable. Implementing a method to always retain the best performing model checkpoint would eliminate this from occurring again. Finally, the last method that could result in better performance is to use a model with a higher base mAP. As the inference is not time critical, the response time of the model is not significantly important and therefore a model that takes longer to perform inference on each image but is more accurate would be worthwhile to investigate.

Regarding the novel contributions of this study, the outcomes were as follows.

1) Despite the limited dataset size and significant reduction in information in each image due to only having a single channel compared to 3 in the training data, the model was able to detect and classify storms successfully.
2) Using the image of the whole Earth disk, the model was still able to correctly detect and classify the TS. This method not only reduces the computational requirements to prepare the data, but also shows the location of the TS in situ on the Earth, which would give users of this model better context as to the TS's location.
3) The use of the Dvorak scale not only enabled us to have greater resolution in the topical storm classes, but also allowed for easier conversion between other intensity scales and for the F1+RMSE metric to be developed.
4) The implementation of the F1+rmse model performance metric resulted in a different model being identified as the best-performing model when compared to using the F1 score as the performance metric.

One approach that was originally considered for this study was the development of a model that utilizes the inherent temporal aspect of the data. The potential advantage of this method would be that incorporating the previous state of the TS over a number of timesteps may enhance the predictive power of the model. The authors decided that this is best suited to a piece of future work as it would directly oppose the main benefits of the work presented here.

V. Conclusion

The research performed in this study aimed to investigate the implementation of generalized pretrained object detection models to the task of the identification and classification of TS systems. The model classification results were generally comparable with results found in the literature, however, the models in this study used more classes (7 or 14 classes versus typically 5 or 6). Instead of developing a bespoke model for the task as per the majority literature reviewed, three pretrained generalized models were used to determine their ability to classify the TS in the images. This approach is significantly faster to develop a working model, required substantially less data (between 2x and 10x), less compute resources and less time to train the models.

Four different datasets were used to train the models to evaluate the impact that reducing the number of TS (by excluding lower intensity storms) and reducing the number of classes had on the model performance. While the model with the highest precision, recall, and F1 score was trained on the dataset without lower intensity storms and reduced number of classes, when taking into account the quality of the predictions as measured by rmse, it was only the third best model. This is because the majority of misclassifications for each model were in neighboring classes, and the classes increased in whole integer steps, rather than increases of 0.5, therefore, any misclassifications were more severe than the models with half-integer steps, which contributed to the higher rmse. When the rmse and F1 score for the models were combined, the best-performing model was the one trained on all of the data and the 0.5 intensity step increases.

Overall, the models performed well considering the data used were significantly different from the data the original models were trained on. While bespoke models have been shown to perform better than the models produced in this study, they also have additional requirements for data preprocessing and when used for classification, the number of classes are generally fewer than the models produced here, meaning their output provides less information. Bespoke models also require significantly more data, compute requirements to train to produce a model of high quality and required the input images to be cropped such that they only contained the TS, which is unlikely to be a valid approach if these models were to be implemented in a real world environment. With additional images and higher quality labels through some of the processes discussed, it may be possible to increase the performance of these models to a comparable level of the bespoke models while still requiring less data than other methods.

To summarize, this study demonstrated that it is possible to produce ML models that are comparable to the best performing bespoke models in other literature, while requiring substantially less training data and less expert knowledge in ML to implement, making this approach significantly more accessible. This result, combined with multiple potential methods to improve the performance of these model, for classifying TS in decades of archived data from all geostationary weather satellites. This could not only be a valuable resource for climate scientists researching

the genesis and decay of TS, but it may also improve the current most complete dataset regarding the evolution and track of TS systems.

ACKNOWLEDGMENT

The author would like to thank the members of the colleagues at S&T for their valuable input.

REFERENCES

[1] K. Bhatia et al., "A potential explanation for the global increase in tropical cyclone rapid intensification," *Nature Commun.*, vol. 13, no. 1, 2022, Art. no. 6626, doi: 10.1038/s41467-022-34321-6.
[2] S. Jiang and L. Tao, "Classification and estimation of typhoon intensity from geostationary meteorological satellite images based on deep learning," *Atmosphere*, vol. 13, no. 7, 2022, Art. no. 1113. [Online]. Available: https://www.mdpi.com/2073-4433/13/7/1113
[3] J. Zhou, J. Xiang, and S. Huang, "Classification and prediction of typhoon levels by satellite cloud pictures through GC–LSTM deep learning model," *Sensors*, vol. 20, no. 18, 2020, Art. no. 5132. [Online]. Available: https://www.mdpi.com/1424-8220/20/18/5132
[4] S. Gardoll and O. Boucher, "Classification of tropical cyclone containing images using a convolutional neural network: Performance and sensitivity to the learning dataset," *EGUsphere*, vol. 2022, pp. 1–29, 2022. [Online]. Available: https://egusphere.copernicus.org/preprints/egusphere-2022-147/
[5] C.-J. Zhang, X.-J. Wang, L.-M. Ma, and X.-Q. Lu, "Tropical cyclone intensity classification and estimation using infrared satellite images with deep learning," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 2070–2086, 2021.
[6] C. Wang, G. Zheng, X. Li, Q. Xu, B. Liu, and J. Zhang, "Tropical cyclone intensity estimation from geostationary satellite imagery using deep convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–16, 2022.
[7] C. Kar, A. Kumar, and S. Banerjee, "Tropical cyclone intensity detection by geometric features of cyclone images and multilayer perceptron," *SN Appl. Sci.*, vol. 1, no. 9, 2019, Art. no. 1099.
[8] "Satellite: Himawari-1 (GMS-1) observing systems capability analysis and review tool," 2023. Accessed: Feb. 22, 2023. [Online]. Available: https://space.oscar.wmo.int/satelliteprogrammes/view/himawari_1st_generation_gms
[9] K. R. Knapp, M. C. Kruk, D. H. Levinson, H. J. Diamond, and C. J. Neumann, "The international best track archive for climate stewardship (ibtracs): Unifying tropical cyclone best track data," *Bull. Amer. Meteorological Soc.*, vol. 91, no. 3, pp. 363–376, 2010.
[10] C. Velden et al., "The dvorak tropical cyclone intensity estimation technique: A satellite-based method that has endured for over 30 years," *Bull. Amer. Meteorological Soc.*, vol. 87, no. 9, pp. 1195–1210, 2006.
[11] "Dvorak current intensity chart wikipedia, wikimedia foundation, 29 Jul. 2019," 2023. Accessed: Feb. 22, 2023. [Online]. Available: https://en.wikipedia.org/wiki/Dvorak_technique#Details_of_the_method
[12] "Dvorak common developmental patterns diagram wikipedia, wikimedia foundation, 29 Jul. 2019," 2023. Accessed: Feb. 22, 2023. [Online]. Available: https://en.wikipedia.org/wiki/Dvorak_technique#/media/File:DvorakCDP1973.png
[13] "Tensorflow model zoo github.com," 2023. Accessed: Feb. 22, 2023. [Online]. Available: https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/tf2_detection_zoo.md
[14] "LabelImg github.com," 2023. Accessed: Feb. 24, 2023. [Online]. Available: https://github.com/heartexlabs/labelImg
[15] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2999–3007, doi: 10.1109/ICCV.2017.324.
[16] "What is root mean square error (RMSE) kaggle.com," 2023. Accessed: Feb. 22, 2023. [Online]. Available: https://www.kaggle.com/general/215997
[17] J. Varghese, "Creating a custom object detector using transfer learning," Sep. 2020. [Online]. Available: https://medium.com/swlh/creating-your-own-custom-object-detector-using-transfer-learning-f26918697889

Jacob Senior-Williams received the M.Sc. degree in computational intelligence and robotics from the University of Sheffield, Sheffield, U.K., in 2018.

Having worked as an AI Engineer with the Advanced Manufacturing Research Centre in Sheffield, he relocated to The Netherlands in August 2020 and is currently employed with S[&]T, Delft, The Netherlands, where he is a Senior Data Scientist focusing on developing a wide range of applications that utilize remote sensing data.

Frank Hogervorst received the M.Sc. degree in aerospace engineering from the Technical University of Delft, Delft, The Netherlands, in 2018.

He has worked on various space applications gathering insights from remote sensed data using AI. He has collaborated with environment inspection agencies to make their way of working more efficient using space data. Currently, he is working on detecting anomalous data from historical meteorological data for improved climate analysis and investigates air pollution in urban areas.

Erwin Platen received the Ph.D. degree in astrophysics from the University of Groningen, Groningen, The Netherlands, in 2009.

The Ph.D. study involved the structural analysis of empty regions within the spatial distribution of matter within the Universe. Here, he developed novel void-finding algorithms and nonlinear reconstruction methods to be able to compare the observed spatial distribution with the simulated spatial distribution. Currently, he is employed as a Scientific Software Engineer with S[&]T, Delft, The Netherlands, where he is working on (science) data-processing projects, related mostly to the reprocessing of archived satellite data, quality-control monitoring of satellite data-processing, and on the testing and integration of the Level-2 Processors for the Sentinel-5 Instrument.

Arie Kuijt received the M.Sc. degree in applied physics from the Technical University of Delft (TUD), Delft, The Netherlands, in 1993, and the Graduation degree in pattern recognition and image processing.

He is currently a Senior Project Manager with the Dutch company Science and Technology (S[&]T), Delft, The Netherlands, and leads various projects such as Earth observation projects, service projects for ESA Data, Innovation, and Science Cluster (DISC) programs as well as high-tech and defence projects.

Jacobus Onderwaater received the M.S. degree in experimental physics from Utrecht University, Utrecht, The Netherlands, in 2012, and the Ph.D. degree in high energy physics from TU Darmstadt, Darmstadt, Germany, in 2016.

Currently serving as the Copernicus Reprocessing Coordinator with EUMETSAT, Darmstadt, Germany, he oversees the production of improved data records for Sentinel missions. In addition, he actively participates in diverse data record initiatives, encompassing collaboration on satellite data with partner agencies.

Roope Tervo received the Ph.D. degree in machine learning from Aalto University, Espoo, Finland, in 2021 with a title "Machine Learning-Based Weather Impact Forecasting."

He is currently working with EUMETSAT, Darmstadt, Germany, as European Weather Cloud (EWC) Service Coordinator and AI/ML expert.

Viju O. John received the M.S. degree in physics and in atmospheric science from the Cochin University of Science and Technology, Kochi, India, in 1998 and 2000, respectively, and the Ph.D. degree in physics from the University of Bremen, Bremen, Germany, in 2005.

He is currently a Climate Product Expert with the European Organization for the Exploitation of Meteorological Satellites (EUMETSAT), Darmstadt, Germany. His research interests include the remote sensing of the Earth's atmosphere, intercalibration of satellite sensors, evaluation climate models using satellite observations, and generation of climate data records of essential climate variables tailored for climate monitoring and for assimilation in climate reanalyses.

Arata Okuyama received the B.S. and M.S. degrees in applied physics from Hokkaido University, Sapporo, Japan, in 1997 and 1999, respectively.

He worked on on-orbit calibration of passive microwave radiometers and visible/infrared imagers. He is currently a Scientific Officer for the Japan Meteorological Agency (JMA), Tokyo, Japan, involved with preparation project of the next JMA geostationary meteorological satellite, Himawari-10.