Cloud Segmentation using Fully Convolutional Neural Networks

Ricardo C. Davila
e-mail: ricardo.davila.1@us.af.mil

March 17, 2021

Contents

1 Abstract

Cloud detection is a fundamental requirement for meteorological or weather satellite imagery, where the presence or absence of clouds plays key roles from weather forecast initialization to supporting military operations. Existing techniques are based on per-pixel classication using computationally intensive classification algorithms placing high demands on computer processing requirements. An alternate technique, such as the one described in this paper, involves leveraging recent advances in machine learning using a deep fully convolutional neural (FCN) network to semantically segment cloud layers based on cloud phase output. This study shows promising initial steps using an FCN to segment clouds based on operational geostationary weather satellite images and true class derived cloud phases. The experimental results show that the proposed method can effectively detect cloud phases in complex scenes for a given month and selected three hour period.

2 Introduction

Since the 1960s, atmospheric satellite data has been assimilated into numerical weather forecast prediction models to improve operational weather nowcasts and forecasts [1]. Applying machine learning techniques to improve combined satellite and weather prediction has been ongoing since the late 1990s [2]; however, instances of using atmospheric weather models along with satellite atmospheric measurement data to predict potential satellite observations are not as numerous. This study proposal outlines a quick study to investigate using Convolutional Neural Networks (CNN) to properly segment cloud phases based on pre-selected and differenced infrared satellite images. If successful, this study will serve as phase I of a multi-staged effort to eventually combine operational weather prediction model output and satellite images to forecast high, mid and low-level cloud decks in 6, 12, and 18 hour prediction periods.

The primary imagery data source for this study is the Geostationary Operational Environmental Satellite (GOES-R). The GOES-R series is the US's advanced fleet of operational geostationary weather satellites. The GOES-R series significantly improves the detection and observation of environmental phenomena that directly affect public safety, protection of property and our nation's economic health and prosperity. GOES-R produces over 35+ operational terrestrial and space weather products for full, continental US and smaller scale mesoscale images with update rates of 10 minutes for large scale products to once per minute for smaller but key mesoscale image data collections. There are many excellent review articles that cover GOES-R capabilities, instruments and mission operations [3, 4, 5, 6, 7].

3 Related Work

Semantic segmentation is a deep learning algorithm that associates a label or category with every pixel in an image and is a key focus area in computer vision. It is used to recognize a collection of pixels that form distinct categories. Typical segmentation applications include, but are not limited

to, autonomous driving, human-machine interaction, computational photography and image search engines [8]. Additionally, convolutional neural networks have been applied to contrail detection [9, 10], LandSat cloud images [11], European Meteosat data [12], Russian satellite Electro-L [13] and ground based cloud detection classification [14]. While this study focuses on segmenting weather cloud or phase types, it differs from these previous studies due to its use of a fully convolutional neural network and using a differenced long wave infrared image as the input.

4 Dataset

The dataset used for this effort uses only GOES-R weather satellite measurements. The GOES-R data were collected over various areas of the eastern continental United States and restricted to this satellite mission's mesoscale regions. These regions are 1000 km by 1000 km and are usually designated by the National Weather Service as high-interest regions where forecasts indicate a high probability of severe weather occurring.

4.1 Time Period

In order to limit computer processing demands and determining the feasibility of using FCNs for cloud segmentation, GOES-R imagery were limited to a three-hour Zulu time period, $0000 - 0300$ Universal Time Coordinated (UTC) or Zulu time for the month of February 2021. The primary requirement was using enough images to establish suitable training, validation and test datasets.

4.2 GOES-R Weather Satellite Cloud Parameters

The truth data or 'true class' cloud segmentation are taken directly from the GOES-R derived cloud phase product. The cloud phase values are detailed in Table 1. The GOES-R cloud phase algorithm uses several satellite imagery channels combined with parameterized based formulas to determine cloud phase values for a given image mesoscale location.

Table 1: GOES-R Cloud Phase Values

Phase	**Cloud Type**
0	Clear
1	Liquid Water
2	Super Cooled Water
3	Mixed
4	Ice

Figure 1 shows cloud phases for a GOES-R mesoscale image collected on 1 February 2021 at 0259 UTC. Similar images are collected every minute for designated mesoscale regions.

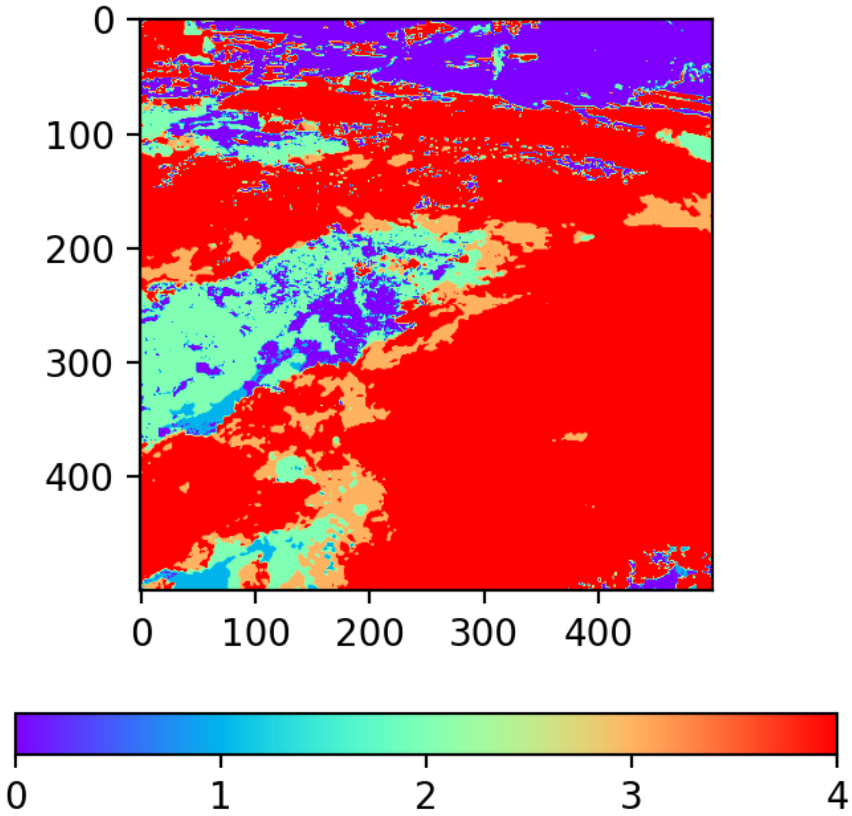

Figure 1: GOES-R cloud phase for mesoscale region collected on 1 February 2021 at 0259 UTC

4.3 GOES-R Weather Satellite Emissive Channels

The GOES-R Advanced Baseline Imager (ABI) is the primary sensor for imaging Earth's weather, ocean and environment. This study uses 2 channels (***rad_ch13*** and ***rad_ch15***) out of a total sixteen reflective and emissive channels available in the ABI system. The two GOES-R ABI bands and bandwidth specifications are listed in Table 2. Figures 3 and 2 show example images for GOES-R channels 13 and 15, respectively. These two images were collected at the same time as that for Figure 1, 1 February 2021 at 0259 UTC.

The measured radiance data from these two emissive channels were collected every 1 minute and have radiance units of $\left(mW/(m^2 \cdot sr \cdot cm^{-1})\right)$. These images also coincide with the respective cloud phase image as shown in Figure 1. It is important that a corresponding cloud phase image be collected at the same time as the channel 13 and 15 images. These sets of images become the training pair for the FCN model. Because of the nature of blackbody radiation in our atmosphere,

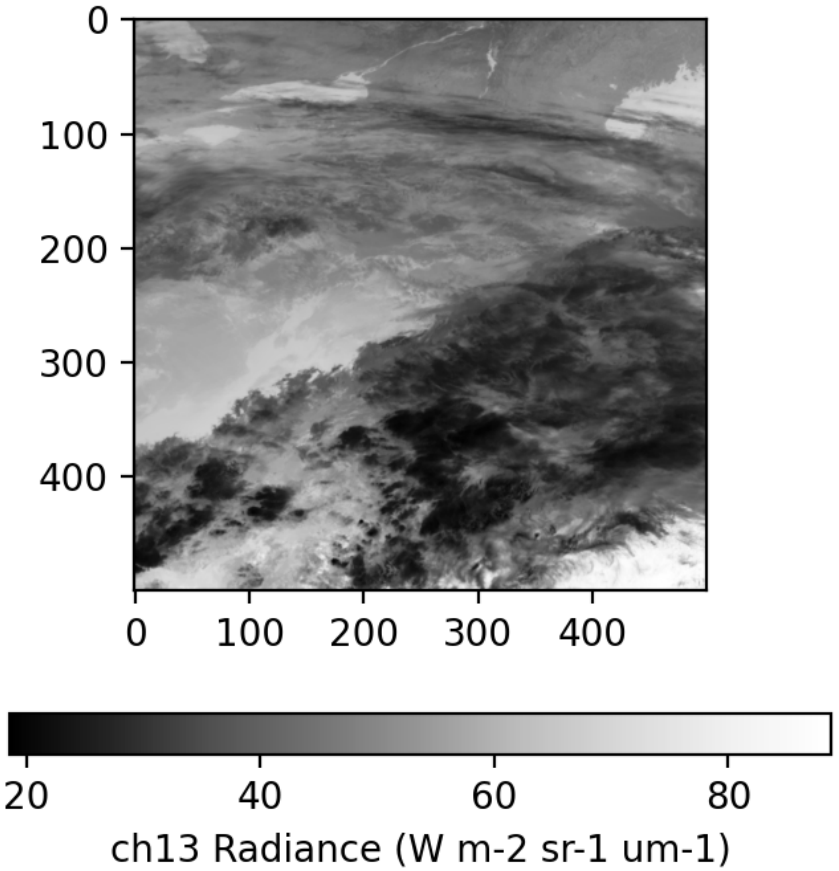

Figure 2: GOES-R infrared imagery for channel 13 collected on 1 February 2021 at 0259 UTC

typical radiance values collected at the ABI sensors increase from ABI channel 13 to channel 15 (i.e., $rad\_ch13 < rad\_ch15$).

Table 2: GOES-R ABI Bands and Bandwidths

ABI Band	**Central Wavelength** (μm)	**Wavelength Range** (μm)	**Spatial Resolution** (km)
13	10.35	10.1 to 10.6	2
15	12.3	11.8 to 12.8	2

We work with the difference between channels 15 and 13 as described in previous contrail detection studies outlined in [9, 10]. A key difference between these previous contrail detection studies and this effort is previous efforts used the differences in brightness temperatures between

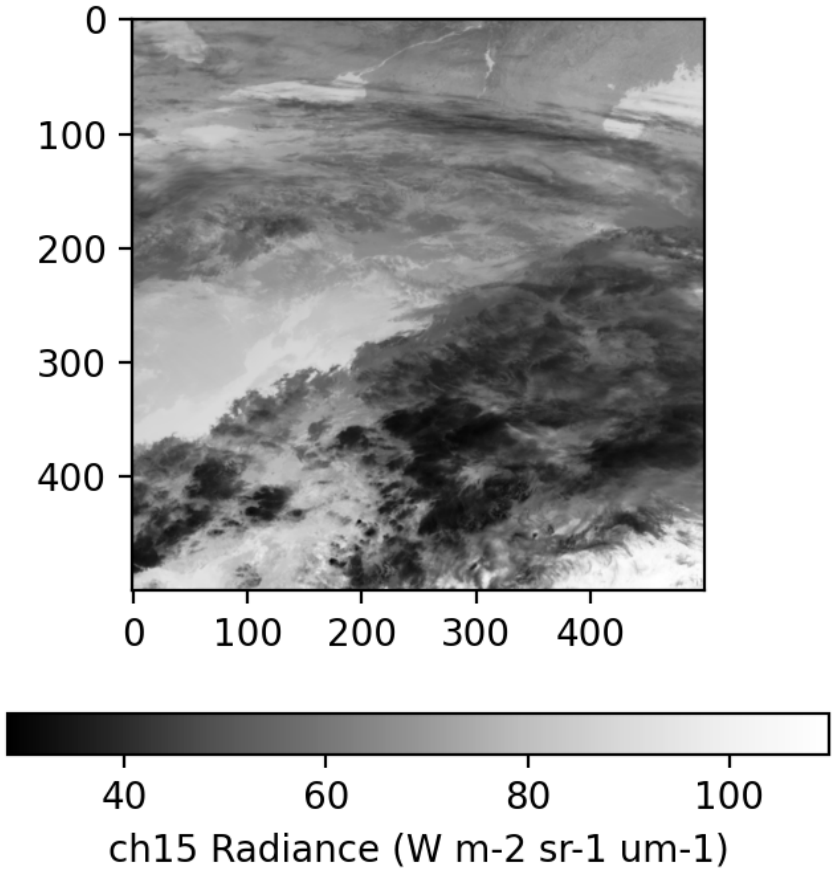

Figure 3: GOES-R infrared imagery for channel 15 collected on 1 February 2021 at 0259 UTC

the two channels while this study works only with the difference in radiance values. Figure 4 shows the normalized radiance value that resulted from the difference between Figures 3 and 2.

5 Approach/Methodology

5.1 Fully Convolutional Neural Network (CNN) and Semantic Segmentation

A primary goal of this study is to segment high clouds (class 4 or ice phase in Table 1) based on a radiometrically differenced GOES-R channel 15 and 13 image as depicted in Figure 4. We want to determine whether or not a convolutional neural network can be trained to identify the mixed and ice clouds, phases 3 and 4, respectively in Table 1. These phases are indicative of high clouds over lower layers (mixed) and very high clouds, such as cirrus clouds, which are composed mainly of ice. A secondary, but high interest, goal is to determine how a the CNN classifies contrails.

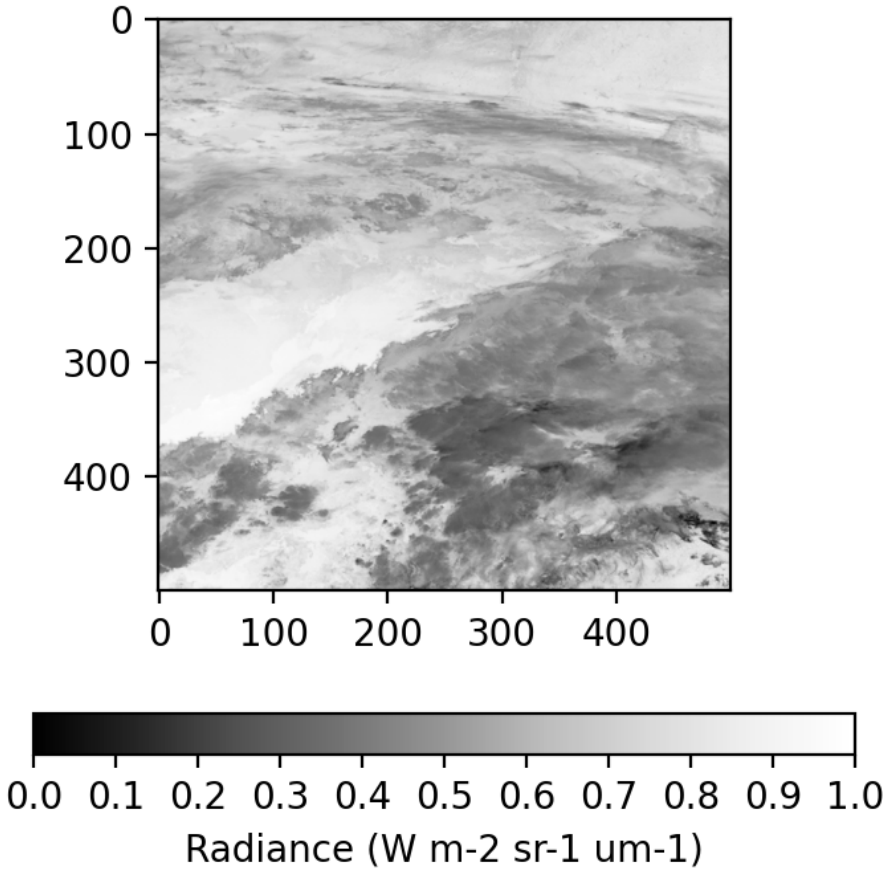

Figure 4: Normalized radiance difference between GOES-R channels 15 and 13 collected on 1 February 2021 at 0259 UTC

We use Fully Convolutional Neural Networks (FCNs) to semantically segment, or classify every pixel in an image according to the class of the object to which it belongs [15, 16, 17]. While clouds can be categorized by features as seen from an observer looking from the ground-to-sky or sky-to-sky (i.e., co-altitude), these same features can not be easily identified in the same manner from an overhead sensor. Using an overhead sensor, such as a satellite detecting clouds from above, the key observed feature is cloud tops. Without another data source to determine vertical features of the cloud, it is difficult to classify whether or not the cloud is a thinly layered or solid like those clouds that make up a large thunderstorm system. The overhead sensor dictates a detection methodology to classify every pixel in an image to every pixel in a true class cloud segmented phase image. We use the normalized difference between channels 13 and 15 as the input training pair along with the associated segmented cloud phases to train the neural network.

The GOES-R algorithm to classify the cloud phase depicted in Figure 1 uses an involved algo-

rithm based on several GOES-R ABI channels and historical imagery collected by a NASA funded and operated cloud sensing sensor platform. To perform a FCN and semantic segmentation for this study, we followed the procedure outlined below:

(1) We collected 450 sample time periods of GOES-R infrared (channels 13 and 15) and cloud phase output. Each sample consisted of a time correlated channel 13, 15, cloud phase and channel 15/13 differenced image.

(2) Next, we normalized the channel 15/13 differenced image based on mean and standard deviation for a given day/time.

(3) We constructed labeled phase data sets for a given day/time by comparing the cloud phase data to the normalized channel 13/15 image. This will result in 5 separate labeled images (per Table 1) for each normalized differenced image such as the one shown in Figure 4. A breakout example is shown below in Figure 5

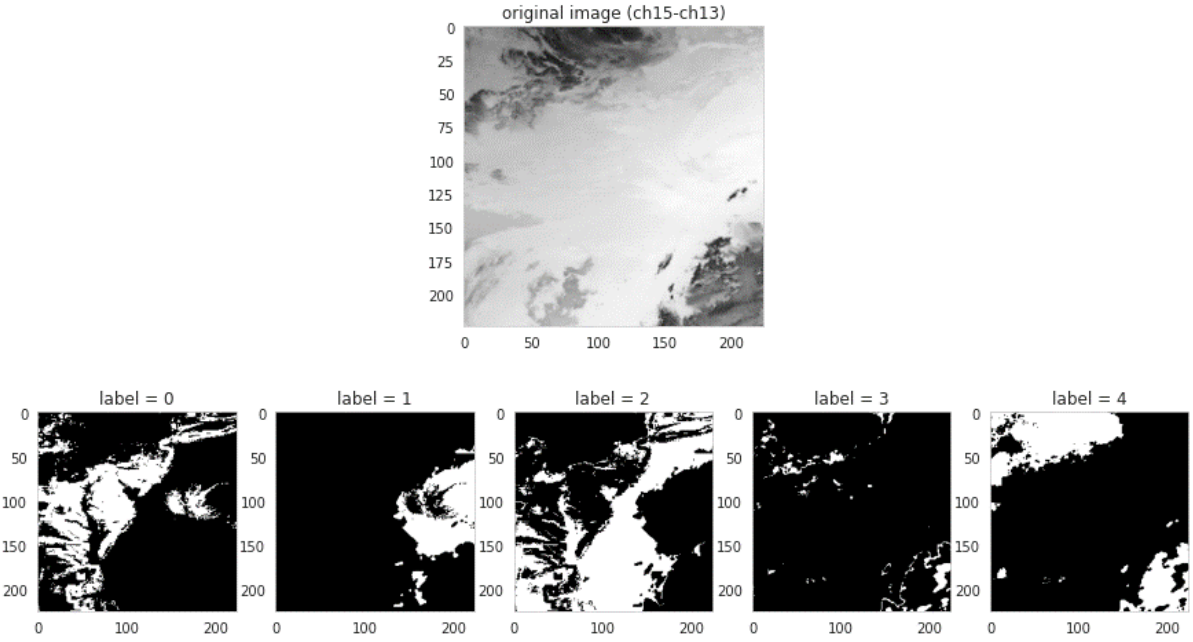

Figure 5: Image segmented by label 0 (clear sky), label 1 (liquid water), label 2 (super colled water), label 3 (mixed) and label 4 (ice)

(4) In order to minimize computer processing time, the final training, validation and testing images were cropped to less than half the size of the original images. Each image is cropped down to a 224×224 image array.

(5) A combined VGG-16, a convolutional neural network that has achieved a 92.7% test accuracy in ImageNet, and fully convolutional neural network (FCN8) was used to perform the semantic segmentation to classify every pixel. The high-level FCN8 architecture is shown in Figure 6. It follows closely the VGG-16 neural network; however, this FCN discards the final classifier layer and convert all fully connected layers to convolutions. The FCN appends a 1×1 convolution with channel dimension the same as the number of segmentation classes, 5 for this study since 5 is the

number of cloud phase categories. The methodology follows closely the pattern outlined in the referenced article [15].

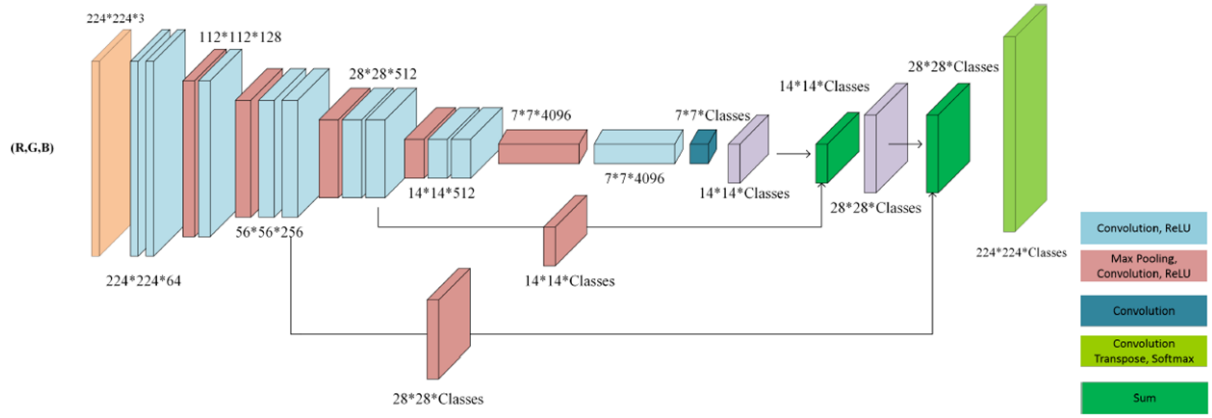

Figure 6: Fully convolutional neural network architecture (FCN-8) [18]

5.2 Experiment Settings

The neural network was developed on an HP ENVY x360 laptop with a Windows 10 operating system using Python version 3.7.10. All the image label development was created via the Spyder interactive development environment (IDE) while the neural network was developed by Jupyter Notebook with the combined Keras API version 2.4.3 and Tensorflow version 2.4.1. Due to the intensive image demands of this project, the Jupyter Notebook was upload to Google Colab to utilize Colab's Graphics Processing Unit (GPU) capability. This reduced processing time for each epoch from approximately 10 minutes using the standard laptop processing unit to approximately 12 seconds using Google Colab's GPUs.

While no changes were made to the neural network architecture, three different models were created for three optimizers. The optimizer code, along with the hyper-parameters, is show below:

```
from keras import optimizers
es = EarlyStopping(monitor='val_loss',patience=15)

(1) sgd = optimizers.SGD(lr=0.01, decay=0.0016, momentum=0.9, nesterov=True)
(2) adam = optimizers.Adam(learning_rate=0.0001)
(3) adam = optimizers.Adam(learning_rate=0.0001,epsilon=1e-08)

model.compile(loss='categorical_crossentropy',
optimizer=(1) or (2) or (3),
metrics=['accuracy'])
```

For the rest of this study, options (1) Stochastic gradient descent (sgd), (2) adam and (3) adam

will be referenced as (a) SGD, (b) Adam and (c) Adam2, respectively. To prevent over fitting, early stopping was implemented with a patience of 15 and validation loss as the monitor.

Of the 450 images, 80% were kept in the training set and the remaining 20% were used in the validation set. The accuracy and loss model evaluations were recorded for each optimizer model per the Python code below:

```
# Compute and display the loss and accuracy of the trained model on the test set.
test_loss, test_accuracy = model.evaluate(x=X_test, y=y_test, verbose=False)
print('accuracy: {acc:0.4f}'.format(acc=test_accuracy))
print('loss: {loss:0.4f}'.format(loss=test_loss))
```

Twelve images were used in a separate test dataset. The confusion matrix and classification report from Python's sklearn were used to help quantify how well the model classified the twelve test images.

```
from sklearn.metrics import confusion_matrix
y_true = segtest.flatten()
y_pred = seg.flatten()
print(confusion_matrix(y_true, y_pred,labels=[0,1,2,3,4]))
target_names = ['class 0', 'class 1', 'class 2','class 3', 'class 4']
print(classification_report(y_true, y_pred, target_names=target_names))
```

6 Results

Figure 7 shows the overall accuracy and loss model evaluations for each optimizer model and the number of epochs required to achieve the respective predictions. Epochs were set to 500; however, with early stopping, the number of epochs for optimizer (a) SGD, (b) Adam and (c) Adam2 were 205, 120 and 117, respectively. The stochastic gradient decent optimizer displayed several spikes in the accuracy and loss graphs compared to the relatively smooth graphs for the two variants of the Adam optimizer. The highest accuracy was recorded for Adam with a learning rate of $0:001$.

6.1 Validation Dataset

Figure 8 shows examples of the differences between the predicted and true classes for the validation data set. The validation dataset is used to estimate prediction error for model selection; for this study, it served as a consistent comparison between the three optimized datasets. Visually, the predicted class for each optimized model is more 'blocked' than the true class; however, overall the predicted class does depict the trends between the different cloud phases. A confusion matrix and associated comparison report was not performed for the validation data set results.

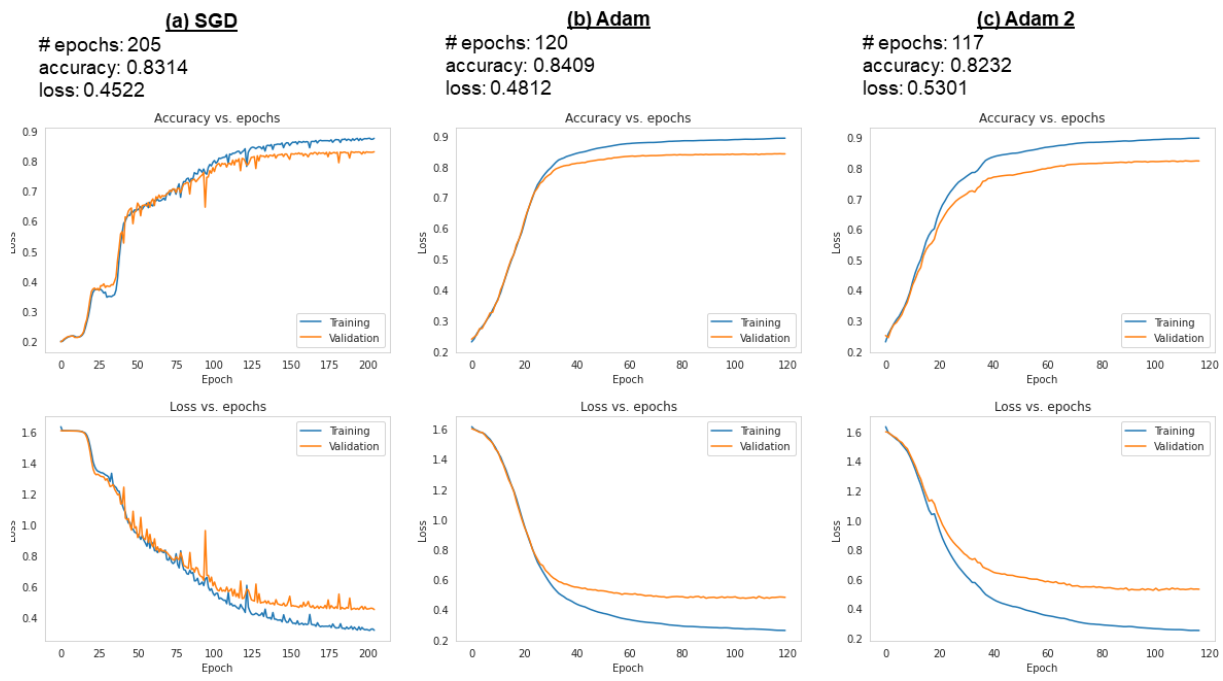

Figure 7: Accuracy and loss statistics and respective plots versus Epochs for (a) SGD, (b) Adam and (c) Adam 2

6.2 Test Dataset

Table 3 contains a list of 12 images that make up the test dataset for the full convolutional neural network. This data is used as an unbiased evaluation of a final model fit on the training dataset. This data set is sorted from the highest to lowest average of the three accuracy values for each optimizer based model predictions. Accuracy is defined as the fraction of correct predictions over the number of samples, where the number of samples for this case is the total pixel count for each image, $224 \times 224 = 50176$.

Since the training dataset consisted of images collected between 0000 to 0300 UTC using GOES-16 imagery, the highest accuracies, in general, from the test data set were primarily GOES-16 images collected between 0000 and 0300 UTC. Further explanation of selected test data samples will be discussed further in the discussion section.

Figure 9 shows the output for a GOES-16 mesoscale region centered over the state of Kentucky collected on 1 October 2020 at 0900 UTC for the three optimized models. There is a mismatch between the colors for the GOES phase map and the true class display from the Python plot code. For the true class output, red is class 0/clear sky, blue is class 1/liquid water, green is class 2/super cooled water, yellow is class 3/mixed, and purple is class 4/ice clouds. While all the training data were collected using GOES-16, a satellite with a footprint covering the Eastern United States and the Atlantic, the UTC time is 6 hours later and the data were collected almost 5 months earlier than the February 2021 training data set. The overall accuracies for the three models is a near or above 80%, but these results are biased by the clear sky (or phase/class 0) category. For this

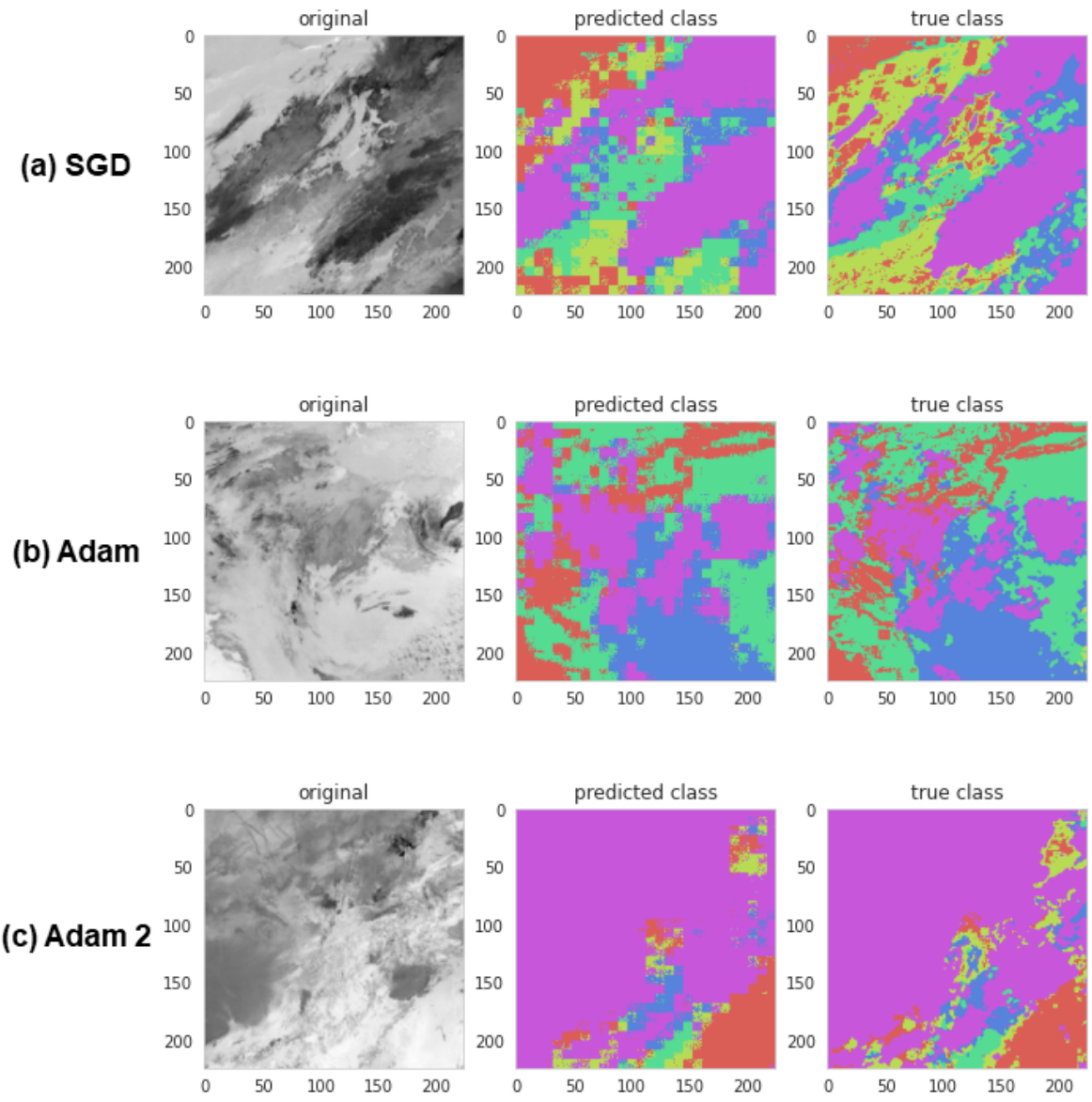

Figure 8: Sample graphic prediction versus true class comparisons for (a) SGD, (b) Adam and (c) Adam 2

sample, the clear sky pixels make up over 85% of the phase image. A visual comparison of the predicted and true classes shows a clear mismatch between the non-phase 0 prediction and true class outputs.

Figure 10 shows the output for a GOES-16 mesoscale region centered over the state of New Jersey collected on 10 February 2021 at 0900 UTC. The average accuracy of the three models was near 70%. This sample's highest precision score was for the class 4/ice clouds with an average precision of 86% for the three models. Precision here is defined as the ratio of the $tp/(tp + fp)$ where tp is the number of true positives and fp the number of false positives. The number of pixels

Table 3: Test Model Results

				accuracy		
GOES	location	date	time (UTC)	SGD	Adam	Adam 2
16	KY	11 Feb 21	0209	0.94	0.94	0.94
16	KY	01 Oct 20	0900	0.79	0.88	0.86
16	NJ	11 Feb 21	0208	0.80	0.81	0.79
17	AK	01 Oct 20	0900	0.77	0.74	0.74
17	NV	01 Oct 20	0900	0.64	0.78	0.75
16	NJ	10 Feb 21	0900	0.72	0.67	0.71
17	AK	11 Feb 21	0208	0.71	0.60	0.61
16	KY	10 Feb 21	0900	0.62	0.61	0.61
17	AK	10 Feb 21	0200	0.63	0.57	0.59
17	NV	11 Feb 21	0209	0.52	0.54	0.54
17	NV	10 Feb 21	0200	0.39	0.27	0.48
16	TX	01 Oct 20	0900	0.21	0.27	0.31

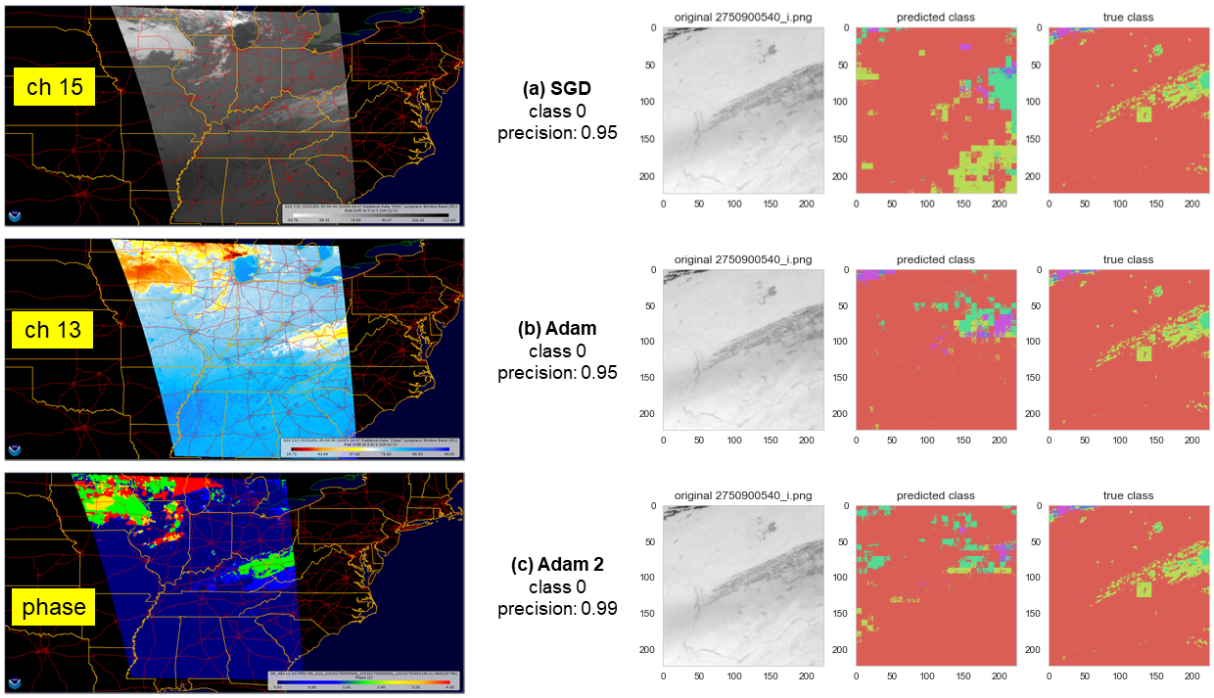

Figure 9: Predictive versus true class predictions for GOES-16 mesoscale image taken on 1 October 2020 at 0900 UTC over Kentucky for (a) SGD, (b) Adam and (c) Adam 2

that make up the ice clouds is above 75% of the image. This sample was collected during February 2021, but at 0900 UTC, outside the $0000 - 0300$ UTC time period. A visual comparison of the predicted and true classes shows a clear mismatch between the non-phase 4 prediction and true class outputs. This sample also indicates that inputting data outside of the $0000 - 0300$ into these three models is not advisable. The model output indicates enough radiometric differences exist outside the $0000 - 0300$ UTC window to influence the segmentation prediction.

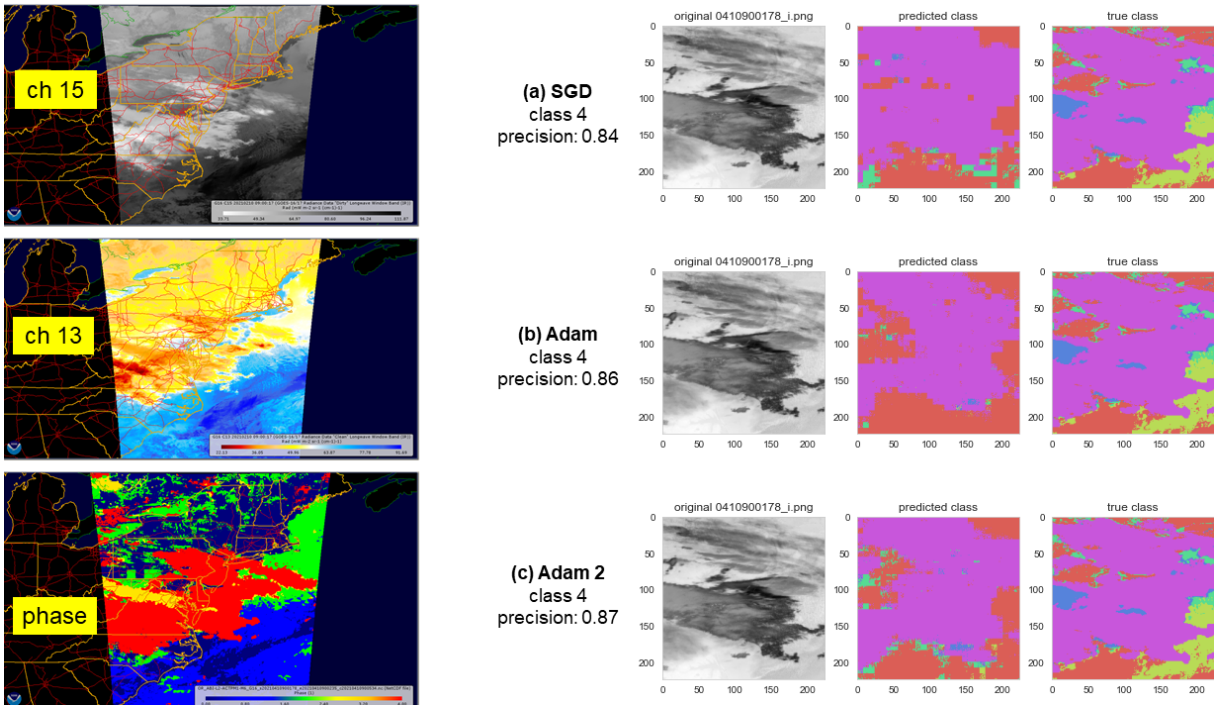

Figure 10: Predictive versus true class predictions for GOES-16 mesoscale image taken on 10 February 2021 over New Jersey at 0900 UTC for (a) SGD, (b) Adam and (c) Adam 2

Figure 11 shows the output for a GOES-16 mesoscale region centered over the state of Texas collected on 10 February 2021 at 0900 UTC. The average accuracy of the three models was near 26%, the lowest of all the samples that make up the test dataset. Like the first sample in Figure 9, this sample was also collected on 1 October 2020 at 0900 UTC. The true class shows the majority of the same had clear skies or class 0 category. A visual comparison of the predicted and true classes shows poor comparisons between all the respective class or phase category. Like the previous sample in Figure 10, inputting data outside of $0000 - 0300$ into these three models is not advisable. While the models clearly differentiate the land, coast and sea, predicting overcast skies over the ocean with clear skies is incorrect. This is another indication that seasonal changes are not captured in the one month February 2021 training set and possible lack of enough training images containing cloud phases over land, coast and sea/ocean.

7 Conclusions and Future Work

This study applied semantic segmentation to a modified GOES-16 image created from two infrared images collected and segmented according to a GOES-16 derived cloud phase product. An already developed and tested fully convolutional neural network (FCN) was used to perform the segmentation and achieved validation dataset accuracies above 80% for three different optimizer FCNs. These results were achieved using a training dataset containing GOES-16 satellite data collected during February 2021 during $0000 - 0300$ UTC for Eastern United States.

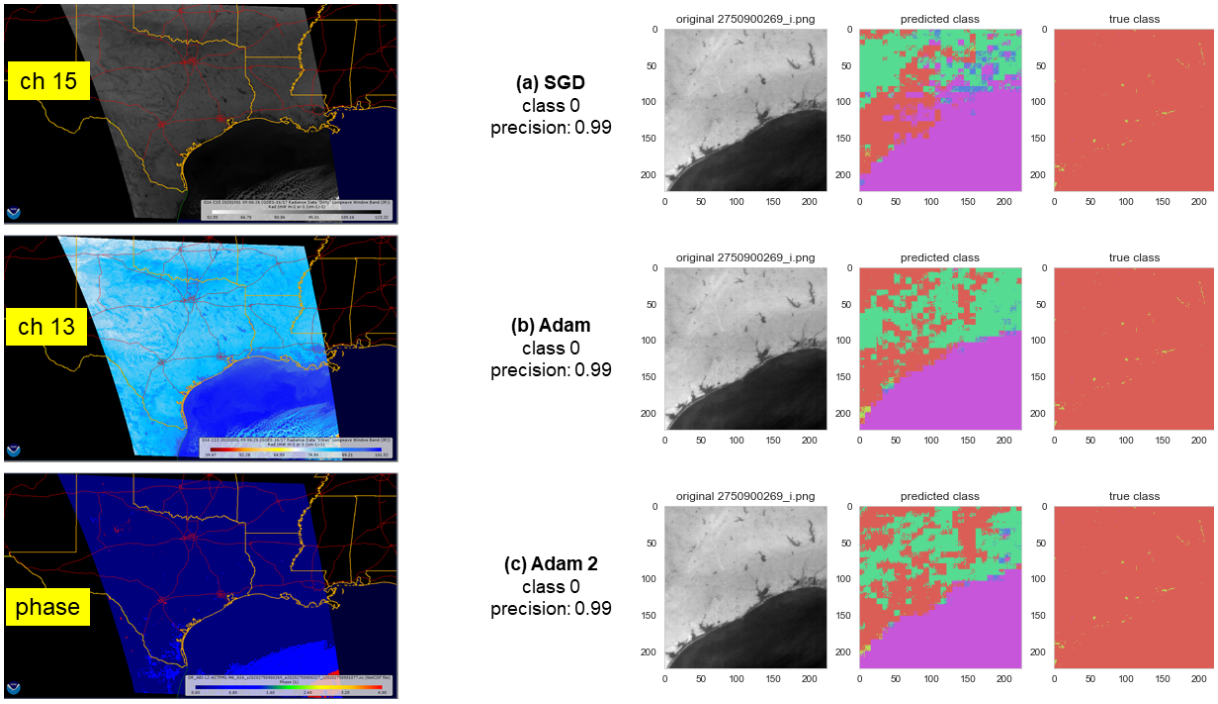

Figure 11: Predictive versus true class predictions for GOES-16 mesoscale image taken on 1 October 2020 at 0900 UTC over Texas for (a) SGD, (b) Adam and (c) Adam 2

While the results show promise, there were several incorrect hypothesis and prediction misclassifications highlighted after comparing FCN predictions to true class cloud phase segmentations. The key conclusions along with future work are listed below:

(1) The FCN can properly segment cloud phases and achieve a relatively high accuracy percentage. For GOES-16 satellite sample images, extrapolation beyond the collected time period and season can lead to misclassifications and degraded prediction accuracies.

(2) Contrail segmentation is not possible using the GOES-16 segmented cloud phases. A fully convolutional neural network along with labeled contrails may be required to properly classify contrail cloud formations.

(3) Proper seasonal segmentation may require data collects over 4 months. For the northern hemisphere, continental United States: spring (March through May), summer (June through August), fall (September through November) and winter (December through February). To capture western regions of United States, at least half of the training images should be GOES-17 satellite images and respective phase product.

The cloud phase segmentation results show enough promise to expand this study to include numerical weather predictions. With a neural network trained using cloud phase segmentations integrated with numerical weather parameters that capture the cloud phase dynamics, it is possible to use this integrated neural network to predict high cloud predictions at least 6 to 12 hours in advance.

References

[1] J. R. Eyre, S. J. English, and M. Forsythe, "Assimilation of satellite data in numerical weather prediction. Part I: The early years," *Quarterly Journal of the Royal Meteorological Society*, vol. 146, no. 726, pp. 49–68, 2020.

[2] R. L. Bankert, M. Hadjimichael, A. P. Kuciauskas, K. L. Richardson, J. Turk, and J. D. Hawkins, "Automating the estimation of various meteorological parameters using satellite data and machine learning techniques," *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2, no. C, pp. 708–710, 2002.

[3] M. M. Gunshor, T. J. Schmit, D. Pogorzala, S. Lindstrom, and J. P. Nelson, "Geostationary Operational Environmental Satellite-R series advanced baseline imagery artifacts," *Journal of Applied Remote Sensing*, vol. 14, no. 03, pp. 1–19, 2020.

[4] S. Kalluri, C. Alcala, J. Carr, P. Griffith, W. Lebair, D. Lindsey, R. Race, X. Wu, and S. Zierk, "From photons to pixels: Processing data from the Advanced Baseline Imager," *Remote Sensing*, vol. 10, no. 2, 2018.

[5] T. J. Schmit, P. Griffith, M. M. Gunshor, J. M. Daniels, S. J. Goodman, and W. J. Lebair, "A closer look at the ABI on the goes-r series," *Bulletin of the American Meteorological Society*, vol. 98, pp. 681–698, apr 2017.

[6] T. J. Schmit, S. S. Lindstrom, J. J. Gerth, and M. M. Gunshor, "Applications of the 16 spectral bands on the Advanced Baseline Imager (ABI).," *Journal of Operational Meteorology*, vol. 06, no. 04, pp. 33–46, 2018.

[7] T. J. Schmit, J. Li, S. J. Lee, Z. Li, R. Dworak, Y. K. Lee, M. Bowlan, J. Gerth, G. D. Martin, W. Straka, K. C. Baggett, and L. Cronce, "Legacy Atmospheric Profiles and Derived Products From GOES-16: Validation and Applications," *Earth and Space Science*, vol. 6, no. 9, pp. 1730–1748, 2019.

[8] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, "A survey on deep learning techniques for image and video semantic segmentation," *Applied Soft Computing Journal*, vol. 70, pp. 41–65, 2018.

[9] D. Zhang, C. Zhang, R. Cromley, D. Travis, and D. Civco, "An object-based method for contrail detection in AVHRR satellite images," *GIScience and Remote Sensing*, vol. 49, no. 3, pp. 412–427, 2012.

[10] G. Zhang, J. Zhang, and J. Shang, "Contrail recognition with convolutional neural network and contrail parameterizations evaluation," *Scientific Online Letters on the Atmosphere*, vol. 14, pp. 132–137, 2018.

[11] K. Aouaidjia and I. Boukerch, "Multi-scale convolutional neural networks for cloud segmentation," no. September 2020, p. 12, 2020.

[12] J. Drönner, N. Korfhage, S. Egli, M. Mühling, B. Thies, J. Bendix, B. Freisleben, and B. Seeger, "Fast cloud segmentation using convolutional neural networks," *Remote Sensing*, vol. 10, no. 11, pp. 1–24, 2018.

[13] V. D. Bloshchinskiy, M. O. Kuchma, A. I. Andreev, and A. A. Sorokin, "Snow and cloud detection using a convolutional neural network and low-resolution data from the Electro-L No. 2 Satellite," *Journal of Applied Remote Sensing*, vol. 14, no. 03, p. 1, 2020.

[14] J. Zhang, P. Liu, F. Zhang, and Q. Song, "CloudNet: Ground-Based Cloud Classification With Deep Convolutional Neural Network," *Geophysical Research Letters*, vol. 45, no. 16, pp. 8665–8672, 2018.

[15] D. Gupta, "A Beginner's guide to Deep Learning based Semantic Segmentation using Keras," *June 06, 2019*, pp. 19–21, 2019.

[16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June, pp. 431–440, IEEE, jun 2015.

[17] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[18] S. Piramanayagam, E. Saber, W. Schwartzkopf, and F. W. Koehler, "Supervised classification of multisensor remotely sensed images using a deep learning framework," *Remote Sensing*, vol. 10, no. 9, pp. 1–25, 2018.