DETECTION OF CLOUDS IN MEDIUM-RESOLUTION SATELLITE IMAGERY USING DEEP CONVOLUTIONAL NEURAL NETS

Amit Hasan [1], Chandi Witharana [1], Mahendra R. Udawalpola [1], Anna K. Liljedahl [2]

[1] Dept. of Natural Resources and the Environment, University of Connecticut, Storrs, Connecticut, USA - (amit.hasan, mahendra.udawalpola, chandi.witharana)@uconn.edu
[2] Woodwell Climate Research Center, Falmouth, Massachusetts, USA - aliljedahl@woodwellclimate.org

KEY WORDS: Cloud detection, Semantic segmentation, Deep learning, Satellite imagery, U-Net.

ABSTRACT:

Cloud detection is an inextricable pre-processing step in remote sensing image analysis workflows. Most of the traditional rule-based and machine-learning-based algorithms utilize low-level features of the clouds and classify individual cloud pixels based on their spectral signatures. Cloud detection using such approaches can be challenging due to a multitude of factors including harsh lighting conditions, the presence of thin clouds, the context of surrounding pixels, and complex spatial patterns. In recent studies, deep convolutional neural networks (CNNs) have shown outstanding results in the computer vision domain. These methods are practiced for better capturing the texture, shape as well as context of images. In this study, we propose a deep learning CNN approach to detect cloud pixels from medium-resolution satellite imagery. The proposed CNN accounts for both the low-level features, such as color and texture information as well as high-level features extracted from successive convolutions of the input image. We prepared a cloud-pixel dataset of approximately 7273 randomly sampled 320 by 320 pixels image patches taken from a total of 121 Landsat-8 (30m) and Sentinel-2 (20m) image scenes. These satellite images come with cloud masks. From the available data channels, only blue, green, red, and NIR bands are fed into the model. The CNN model was trained on 5300 image patches and validated on 1973 independent image patches. As the final output from our model, we extract a binary mask of cloud pixels and non-cloud pixels. The results are benchmarked against established cloud detection methods using standard accuracy metrics.

1. INTRODUCTION

Cloud detection in satellite imagery is very important in many remote sensing applications (Pugazhenthi & Kumar, 2020). Being complex in shape, clouds are very difficult to detect from satellite imagery (Shrivastava, 2013). Researchers have already developed several methods to detect cloud pixels from satellite imagery, such as, Rule-based cloud detection- Fmask (Zhu & Woodcock, 2012), Machine learning approaches- Bag-of-words and SVM (Yuan & Hu, 2015), SVM classification (Bai et al., 2016; Ishida et al., 2018). However, current methods primarily rely on per pixel-based classification algorithms, thus mainly focusing on the spectral characteristics or the statistics of pixel values. This leads to misclassifications of pixels with similar spectral signatures, for example, highly reflective man-made structures, sand in deserts, and snow/ice. The spatial patterns are often ignored, or solely used in a simple post-processing step, mainly due to the lack of efficient methods for including them in the analysis (Jeppesen et al., 2019).

Owing to their superior performances in computer vision tasks such as everyday image understanding, medical image analysis, deep learning (DL) algorithms have radically been adopted in remote sensing image analysis. Several DL-based (DL) approaches, such as convolutional neural nets (CNNs) (Li et al., 2019; Mateo-Garcia et al., 2017; F. Xie et al., 2017; Zhan et al., 2017) haven secured a wider attention in recent years; however, utilization of sophisticated DL architectures is in operational context yet at exploratory phases. A plethora of DL CNN architectures have developed and tested in automated image analysis tasks, including classification (VGG16 (Simonyan & Zisserman, 2015), InceptionV3 (Szegedy et al., 2016), ResNet50 (He et al., 2016), Xception (Chollet, 2017), InceptionResNetV2 (Szegedy et al., 2017), ResNeXt50 (S. Xie et al., 2017)), detection (R-CNN (Girshick et al., 2014), R-FCN (Dai et al., 2016), SSD (Liu et al., 2016)), semantic segmentation (ParseNet (Liu et al., 2015), U-Net (Ronneberger et al., 2015), PSPNet (Zhao et al., 2017)), and semantic instance segmentation (SOLOv2 (Wang et al., 2020), Mask R-CNN (He et al., 2017), UPSNet (Xiong et al., 2019), DeepLabv2 (Chen et al., 2017)). Typically, each DLCNN model has its own pros and cons with respect to performances and computational needs. In most instances, these algorithms are application dependent, thus, require various adaptation strategies such as re-training based a new set of training samples, tuning of hyper parameters, modification of the architecture, and inclusion of additional data inputs. Among other contenders, the U-Net architecture (Ronneberger et al., 2015) is one of the widely used DLCNN based image segmentation algorithms. This is one of the simplified DL architectures hence outperforms, both computationally and accuracy-wise, other state-of-the-art image segmentation algorithms (Soni et al., 2020). In addition to spectral properties, clouds can have different and distinct characteristics (e.g. shape attributes, background separation, shadow, density attributes) that can prudently be mined in automated classification process (Mahajan & Fataniya, 2019). It is evident that we can visually differentiate clouds as bright feature in standard RGB given that cloud density is not very thin. Other than RGB channels, near infrared (NIR), visible-infrared (VIR), thermal infrared (T-IR) bands exhibit significant responses to cloudy regions (Jan et al., 2019). The overarching of our study is to explore the possibility of modifying the generic U-Net architecture to classify cloud pixels from moderate resolution satellite images. Through modifications, we aim to reduce the number of trainable parameters by decreasing the number of convolutional layers. However, the modified architecture is yet capable enough to extract contextual information from images.

Depending on sensors characteristics, satellite imagery is acquired at multiple spatial resolutions and spectral specifications. Moderate resolution satellite sensors, such as Landsat-8 and Sentinel-2 record imagery at 30m, and 10m resolutions, respectively, whereas very high spatial resolution

commercial satellite sensors such as WorldView-2 acquire imagery at 0.5m resolution. A limited number of sensors own distinct spectral bands (e.g., band 9 (cirrus) of Landsat-8) of which wavelengths are sensitive to clouds. This luxury is not available with a majority of sensors which have limited spectral ranges. Most cases spectral resolution is confined to visible and NIR range. Thus, in our model development process, we purposely focused only on blue, green, red, and NIR channels of Landsat-8 and Sentinel-2. By doing this we aimed to understand how feasible and transferable a DLCNN model is when classifying cloud pixels only based on visible and NIR channels. In our systematic experiment, we utilized candidate scenes acquired by Landsat-8 and Sentinel-2 sensors to train the modified version of U-Net model. We evaluated the model performances based on standard accuracy metrics.

2. METHODS

2.1 Study Area

We centred the analysis on satellite image scenes acquired by chose Landsat-8 and Sentinel-2 sensors. Both sensors provide cloud masks. Image scenes were chosen randomly and representing different biomes (Figure 1). We downloaded a total of 121 satellite images- 60 of Landsat-8 (30m) and 61 of Sentinel-2 (20m) from the USGS earth explorer. We did not use 10m resolution images from Sentinel-2 due to the absence of 10m cloud masks. Distribution of the selected image scenes is shown in Figure 2. Of the available multispectral channels, we selected only red, green, blue and NIR bands for model development. There are two reasons for only relying on four bands; firstly, these bands are commonly available in almost any multispectral satellite imagery (including commercial satellite imagery), thus it will ensure that our model can also be utilized with satellites other than Landsat-8 and Sentinel-2, and secondly, these bands show significant response for cloud pixels (Yao et al., 2022).

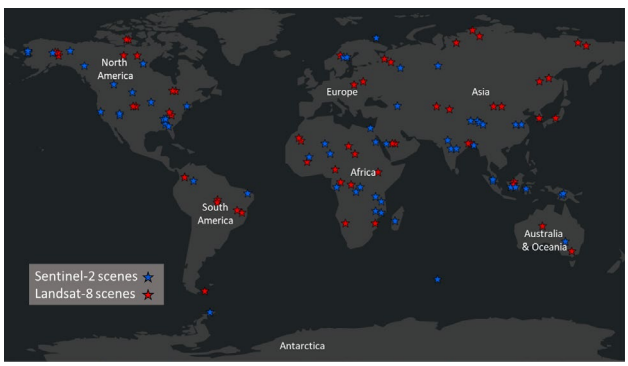

Figure 1. Distribution of the satellite image scenes used in this study.

Downloaded image scenes were tiled for training and prediction purposes. Figure 2 shows the general workflow.

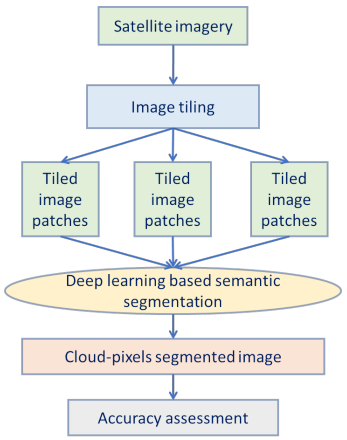

Figure 2. Simplified block diagram of the study.

2.2 Data Processing

Array size of the Landsat-8 scenes is approximately 7000 by 7000 pixels whereas for the Sentinel-2 scenes, it is around 5000 by 5000 pixels. Due to memory limitations, smaller tile sizes are preferred for deep learning models. We tiled the large image scenes into a total of 7273 image tiles of 320 by 320 pixels (Figure 3). We randomly selected 5300 image tiles for training the DL model and kept the rest of the image tiles for the validation and testing purposes. We utilized 30m QA_PIXEL band in Landsat-8 and 20m SCL band in Sentinel-2 image scenes to create cloud masks. Table 1 summarizes the image scenes and tiles from different sensors.

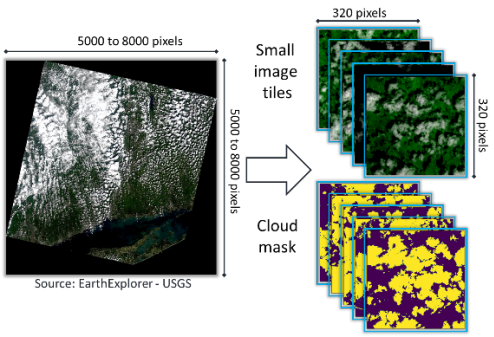

Figure 3. Tiling of the satellite image scenes and cloud mask generation.

Table 1: Summary of satellite image scenes and number of image tiles from Landsat-8 and Sentinel-2.

	Sentinel-2	Landsat-8	Total
Scenes	61	60	121
Image tiles	1799	5474	7273
Tiles for training	1300	4000	5300
Tiles for validation	499	1474	1973

2.3 Model Preparation

We developed and trained a modified version of the U-Net model to detect cloud pixels using medium resolution satellite imagery. The U-Net architecture is an encoder-decoder based deep learning model and was originally utilized for bio-medical image segmentation (Ronneberger et al., 2015). U-Net concatenates the encoder (blue blocks in Figure 4) feature maps to up-sampled feature maps from the decoder (red blocks in Figure 4) at every stage to form a ladder-like structure. A simplified block diagram of the modified U-Net architecture is shown in Figure 4.

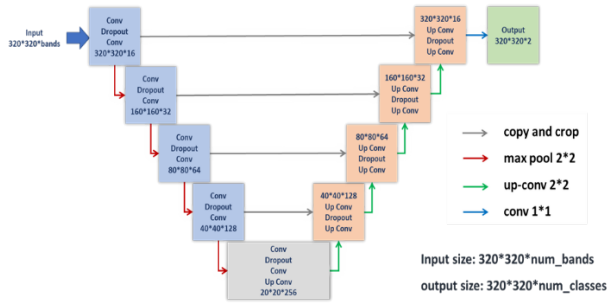

Figure 4. Simplified block diagram of the modified U-Net model.

The modified U-Net model consists of 10 smaller blocks. The blocks on the left side reduce the image dimensionality in the x and y-dimension, collects information and stacks all the extracted feature in the z-dimension. Once the dimension is reduced to 20 pixels by 20 pixels, the red blocks on the right side increase the image dimension in the x and y dimension until the dimension becomes equal to the input x and y dimensions. In the red blocks, while moving from bottom to top layers, the number of z dimension gets reduced and in the final output layer the dimension is reduced to the same height and width as the input image. Here we have two classes, one for the cloud object, the other for the background which is a default class in almost any image segmentation networks.

The modified U-Net model that we proposed consists of 19 convolutional or deconvolutional layers, whereas the original U-Net model has 23 layers. The convolution operations used in the modified U-Net model are padded convolutions but in the original U-Net, there is spatial reduction between subsequent convolutional layers. Padding improves performance by keeping information at the borders (Islam et al., 2021). Overall, the modified U-Net model has a smaller number of parameters compared to the original version. Thus, the modified U-Net takes less time to train and to infer.

2.4 Model Training

We trained the proposed U-Net model up to 300 epochs in a local machine with Intel(R) Core (TM) i9 CPU with NVIDIA GeForce RTX 2070 SUPER with 8GB of GPU memory. The training time increases based on the number of training samples, thus while training with both Landsat-8 and Sentinel-2 image tiles, average training time per epoch was 71 seconds. While training the model, we used a learning rate of 0.0001 and categorical cross-entropy as the loss function. The training parameters are listed in Table 2.

Table 2: Model parameters and machine specifications.

Training parameters		
Model specifications	Learning rate	0.0001
	Epochs	300
	Loss	Categorical cross-entropy
Machine specifications	CPU	Intel Core i9-10900
	RAM	128 GB
	GPU	NVIDIA RTX-2070 super
	GPU memory	8GB GDDR6
Average Training time	Landsat-8 data	59 s/epoch
	Sentinel-2 data	16 s/epoch
	Combined	71 s/epoch

As seen in the loss graph (Figure 5) the validation loss decreases up to 150 epochs and then fluctuates around the same values. Figure 6 shows the validation accuracy for the training process on the combined data reaches the plateau between epochs 150 to 200.

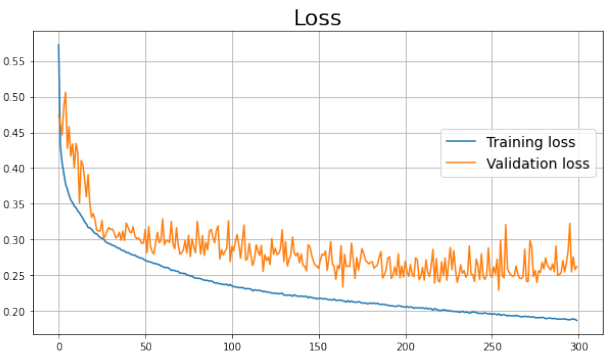

Figure 5. Loss graph for the training process on the combined data.

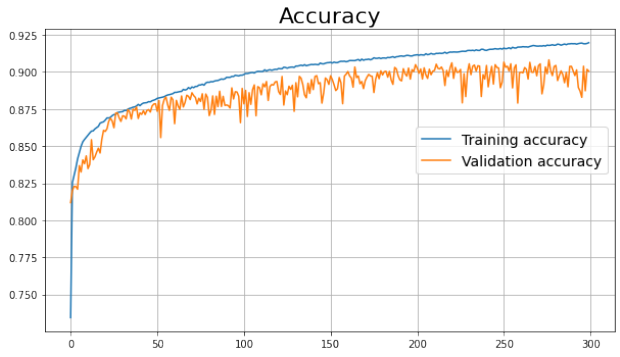

Figure 6. Accuracy graph for the training process on the combined data.

2.5 Accuracy Assessment

We conducted a multi-step accuracy assessment for the outputs. The outputs are in the form of class names and binary masks. We considered the output pixels having the same values as the validation cloud masks as correctly predicted. Figure 7 shows the confusion matrix and defines the terms such as, true positive, true negative, false positive, and false negative which are used in the model evaluation metrics.

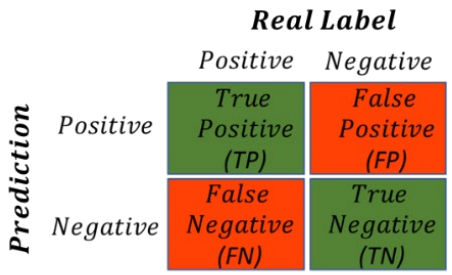

Figure 7. Confusion matrix.

We calculated accuracy, precision, recall and F1-score for each of for each of the images as well as for the whole validation dataset using equations 1, 2, 3, and 4.

$$Accuracy = \frac{true\ positive + true\ negative}{total\ predicted} \quad (1)$$

$$Precision = \frac{true\ positive}{true\ positive + false\ positive} \quad (2)$$

$$Recall = \frac{true\ positive}{true\ positive + false\ negative} \quad (3)$$

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4)$$

3. RESULTS AND DISCUSSION

We evaluated the trained model based on the validation dataset which consists of 1973 image tiles randomly selected from 121 Landsat-8 and Sentinel-2 image scenes.

3.1 Evaluation

After the model training step was completed, we calculated accuracy, precision, recall, F1-score depending on which dataset was used to train the model and which dataset was used to evaluate the model. Table 1 shows the mean accuracy is high if the model was trained and evaluated on only Landsat-8 dataset. However, for evaluating on the combined dataset, the highest accuracy of 87.27% was achieved when the model was trained on the combined dataset. The model trained only on one satellite data performs poorly on the other type of satellite data (Tables 3, 4, 5, 6). All the evaluation metrics such as mean accuracy, precision, recall, and F1-score have higher values when trained and validated on the similar types of datasets (Tables 3, 4, 5, 6).

Table 3: Mean accuracy values on different types of datasets.

Mean Accuracy (%)		Evaluated on		
		Landsat-8	Sentinel-2	Combined
Trained on	Landsat-8	89.8	60.21	82.32
	Sentinel-2	65.12	84.84	70.1
	Combined	88.99	82.2	87.27

Table 4: Mean precision values on different types of datasets.

Mean Precision (%)		Evaluated on		
		Landsat-8	Sentinel-2	Combined
Trained on	Landsat-8	90.28	74.02	86.63
	Sentinel-2	77.95	86.24	80.11
	Combined	89.45	84.15	88.11

Table 5: Mean recall values on different types of datasets.

Mean Recall (%)		Evaluated on		
		Landsat-8	Sentinel-2	Combined
Trained on	Landsat-8	89.82	61.55	82.67
	Sentinel-2	63.87	84.77	69.16
	Combined	88.99	82.47	87.34

Table 6: Mean F1-scores on different types of datasets.

Mean F1-score (%)		Evaluated on		
		Landsat-8	Sentinel-2	Combined
Trained on	Landsat-8	89.19	54.82	81.52
	Sentinel-2	59.09	83.5	65.49
	Combined	88.3	80.7	86.38

Our accuracy budget reveals that if we train the model with only one type of satellite image tiles, that particular model performs well only on that type of satellite image tiles and performs poorly on the other type of satellite image tiles. Adding different types of satellite image tiles makes the model robust and thus the model trained on combined data performs well on almost all types of satellite image tiles. In this study, we could not implement other methods such as Fmask, DeepLab, DCN, MSCFF on our dataset, however, the results from those models (Table 7) as reported by other researchers (Li et al., 2019), are similar to our results on different datasets.

Table 7: Accuracy scores from other studies based on Landsat-8 scenes (Li et al., 2019).

Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Fmask	89.59	85.8	93.01	89.3
DeepLab	87.72	91.26	81.37	86
DCN	92.37	95.96	87.27	91.4
MSCFF	94.96	95.05	93.93	94.5

Sample results along with the original cloud mask are shown in Figure 8. The yellow pixels show the cloudy pixels. Visually these results look promising, and the cloud pixels seem to be labelled correctly.

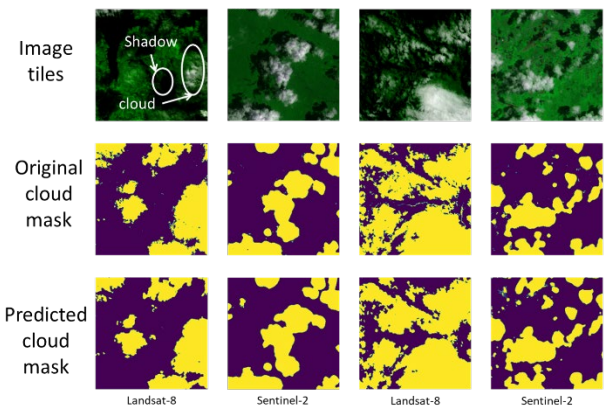

Figure 8. Samples of predicted cloud masks compared to original cloud masks.

We randomly selected some samples with lower accuracy values. Figure 9 shows some sample prediction where the accuracy values are lower than the average. In the visual inspection, it looks like there are some issues with the original cloud masks and most of these results are from the Sentinel-2 cloud masks.

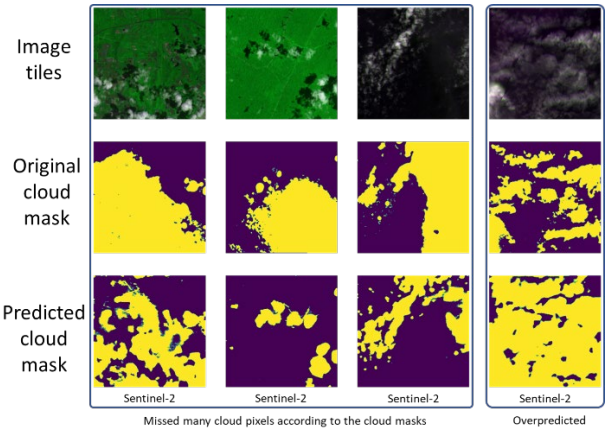

Figure 9. Some randomly selected predicted cloud masks with lower accuracy.

3.2 Challenges

Based on visual inspections, the predicted cloud mask seems to be consistent with provided cloud mask. However, there might be some incorrect labels in the provided cloud masks. As Figure 10 shows, in some cases roads are marked as clouds in some of the Sentinel-2 cloud masks. Sometimes, rivers are marked as clouds in the provided cloud masks. Thus, these types of issues on the training samples might cause poor performance of the model on some image tiles.

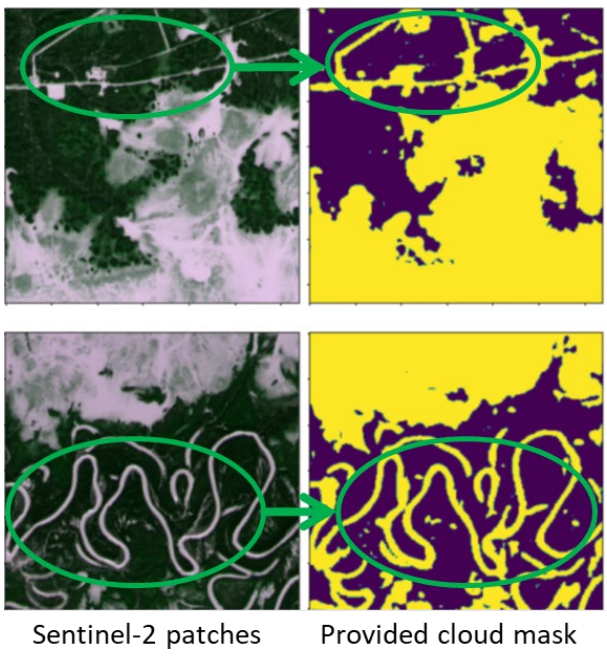

Figure 10. Probable issues with the original cloud masks

4. CONCLUSION

In this study, we propose a new approach for cloud detection from medium resolution satellite imagery using a deep learning-based image segmentation algorithm named U-Net. The core principle is to utilize contextual information in the image rather than using traditional cloud detection algorithms based on pixel values. Our cloud detection results showed that the proposed pipeline performed well on combined Landsat-8 and Sentinel-2 dataset. Our proposed method is applicable in a variety of use cases and can be repurposed with different types of satellite imagery. A shortcoming of the method is that we need to rely on provided cloud masks from different sources. Our future research will address this issue and reduce the dependency on training samples by means of image augmentations on manually inspected training samples.

ACKNOWLEDGEMENTS

This work is funded by the U.S. National Science Foundation, NSF, Navigating the New Arctic Program (Navigating the new Arctic tundra through big data, artificial intelligence, and cyberinfrastructure; Award #s 1927872, 1927723, 1927729, 1927720 & 1927920).

REFERENCES

Bai, T., Li, D., Sun, K., Chen, Y., & Li, W. (2016). Cloud detection for high-resolution satellite imagery using machine learning and multi-feature fusion. *Remote Sensing*, *8*(9), 715. https://doi.org/10.3390/rs8090715

Chen, L.-C., Papandreou, G., Schroff, F., & Adam, H. (2017). *Rethinking Atrous Convolution for Semantic Image Segmentation*. https://doi.org/10.48550/arxiv.1706.05587

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-Janua*, 1800–1807. https://doi.org/10.1109/CVPR.2017.195

Dai, J., Li, Y., He, K., & Sun, J. (2016). R-FCN: Object detection via region-based fully convolutional networks. *Advances in Neural Information Processing Systems*, 379–387. https://github.com/weiliu89/caffe/tree/ssd

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 580–587. https://doi.org/10.1109/CVPR.2014.81

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2017). Mask R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, *2017-Octob*, 2980–2988. https://doi.org/10.1109/ICCV.2017.322

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 770–778. https://doi.org/10.1109/CVPR.2016.90

Ishida, H., Oishi, Y., Morita, K., Moriwaki, K., & Nakajima, T. Y. (2018). Development of a support vector machine based cloud detection method for MODIS with the adjustability to various conditions. *Remote Sensing of Environment*, *205*, 390–407. https://doi.org/10.1016/j.rse.2017.11.003

Islam, M. A., Kowal, M., Jia, S., Derpanis, K. G., & Bruce, N. D. B. (2021). *Position, Padding and Predictions: A Deeper Look at Position Information in CNNs*. http://arxiv.org/abs/2101.12322

Jan, M., David, T., Riesland, W., Eshelman, L. M., Nakagawa, W., Martin, J. A. S., Tauc, J., Riesland, D. W., Shaw, J. A., & Tauc, M. J. (2019). Simulations and experimental results of cloud thermodynamic phase classification with three SWIR spectral bands. *Https://Doi.Org/10.1117/1.JRS.13.034526*, *13*(3), 34526. https://doi.org/10.1117/1.JRS.13.034526

Jeppesen, J. H., Jacobsen, R. H., Inceoglu, F., & Toftegaard, T. S. (2019). A cloud detection algorithm for satellite imagery based on deep learning. *Remote Sensing of Environment*, *229*, 247–259. https://doi.org/10.1016/j.rse.2019.03.039

Li, Z., Shen, H., Cheng, Q., Liu, Y., You, S., & He, Z. (2019). Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors. *ISPRS Journal of Photogrammetry and Remote Sensing*, *150*, 197–212. https://doi.org/10.1016/j.isprsjprs.2019.02.017

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9905 LNCS*, 21–37. https://doi.org/10.1007/978-3-319-46448-0_2

Liu, W., Rabinovich, A., & Berg, A. C. (2015). *ParseNet: Looking Wider to See Better*. https://github.com/weiliu89/caffe/tree/fcn

Mahajan, S., & Fataniya, B. (2019). Cloud detection methodologies: variants and development—a review. *Complex & Intelligent Systems 2019 6:2*, *6*(2), 251–261. https://doi.org/10.1007/S40747-019-00128-0

Mateo-Garcia, G., Gomez-Chova, L., & Camps-Valls, G. (2017). Convolutional neural networks for multispectral image cloud masking. *International Geoscience and Remote Sensing Symposium (IGARSS)*, *2017-July*, 2255–2258. https://doi.org/10.1109/IGARSS.2017.8127438

Pugazhenthi, A., & Kumar, L. S. (2020). Automatic cloud segmentation from INSAT-3D satellite image via IKM and IFCM clustering. *IET Image Processing*, *14*(7), 1273–1280. https://doi.org/10.1049/iet-ipr.2018.5271

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 234–241.

Shrivastava, P. (2013). Cloud detection with MATLAB. *Journal of Engineering Science and Technology Review*, *6*(1), 68–71. https://doi.org/10.25103/jestr.061.13

Simonyan, K., & Zisserman, A. (2015, September 4). Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR*

2015 - Conference Track Proceedings. https://doi.org/10.48550/arxiv.1409.1556

Szegedy, C., Ioffe, S., Vanhoucke, V., & Alemi, A. A. (2017). Inception-v4, inception-ResNet and the impact of residual connections on learning. *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, 4278–4284. https://doi.org/10.48550/arxiv.1602.07261

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2016-Decem*, 2818–2826. https://doi.org/10.1109/CVPR.2016.308

Wang, X., Zhang, R., Kong, T., Li, L., & Shen, C. (2020). SOLOv2: Dynamic and fast instance segmentation. *Advances in Neural Information Processing Systems*, *2020-Decem*. https://doi.org/10.48550/arxiv.2003.10152

Xie, F., Shi, M., Shi, Z., Yin, J., & Zhao, D. (2017). Multilevel cloud detection in remote sensing images based on deep learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *10*(8), 3631–3640. https://doi.org/10.1109/JSTARS.2017.2686488

Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-Janua*, 5987–5995. https://doi.org/10.1109/CVPR.2017.634

Xiong, Y., Liao, R., Zhao, H., Hu, R., Bai, M., Yumer, E., & Urtasun, R. (2019). Upsnet: A unified panoptic segmentation network. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, *2019-June*, 8810–8818. https://doi.org/10.1109/CVPR.2019.00902

Yao, X., Guo, Q., Li, A., & Shi, L. (2022). Optical remote sensing cloud detection based on random forest only using the visible light and near-infrared image bands. *European Journal of Remote Sensing*, *55*(1), 150–167. https://doi.org/10.1080/22797254.2021.2025433

Yuan, Y., & Hu, X. (2015). Bag-of-Words and Object-Based Classification for Cloud Extraction from Satellite Imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *8*(8), 4197–4205. https://doi.org/10.1109/JSTARS.2015.2431676

Zhan, Y., Wang, J., Shi, J., Cheng, G., Yao, L., & Sun, W. (2017). Distinguishing Cloud and Snow in Satellite Images via Deep Convolutional Network. *IEEE Geoscience and Remote Sensing Letters*, *14*(10), 1785–1789. https://doi.org/10.1109/LGRS.2017.2735801

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid scene parsing network. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, *2017-Janua*, 6230–6239. https://doi.org/10.1109/CVPR.2017.660

Zhu, Z., & Woodcock, C. E. (2012). Object-based cloud and cloud shadow detection in Landsat imagery. *Remote Sensing of Environment*, *118*, 83–94. https://doi.org/10.1016/j.rse.2011.10.028