Evaluation of U-Net Backbones for Cloud Segmentation in Satellite Images

Laura G. M. Arakaki[1], Leandro H. F. P. Silva[1,2], Matheus V. Silva[1], Bruno M. Melo[2], André R. Backes[3], Mauricio C. Escarpinati[2] and João Fernando Mari[1]

[1]*Instituto de Ciências Exatas e Tecnológicas, Universidade Federal de Viçosa, Rio Paranaíba, Brazil*

[2]*School of Computer Science, Federal University of Uberlândia, Uberlândia, Brazil*

[3]*Department of Computing, Federal University of São Carlos, São Carlos-SP, Brazil*

Keywords: Cloud Segmentation, U-Net, Cloud-38, Convolutional Neural Networks, Remote Sensing.

Abstract: Remote sensing images are an important resource for obtaining information for different types of applications. The occlusion of regions of interest by clouds is a common problem in this type of image. Thus, the objective of this work is to evaluate methods based on convolutional neural networks (CNNs) for cloud segmentation in satellite images. We compared three segmentation models, all of them based on the U-Net architecture with different backbones. The first considered backbone is simpler and consists of three contraction blocks followed by three expansion blocks. The second model has a backbone based on the VGG-16 CNN and the third one on the ResNet-18. The methods were tested using the Cloud-38 dataset, composed of 8400 satellite images in the training set and 9201 in the test set. The model considering the simplest backbone was trained from scratch, while the models with backbones based on VGG-16 and ResNet-18 were trained using fine-tuning on pre-trained models with ImageNet. The results demonstrate that the tested models can segment the clouds in the images satisfactorily, reaching up to 97% accuracy on the validation set and 95% on the test set.

1 INTRODUCTION

After World War II, the United States and the Union of Soviet Socialist Republics (USSR) disputed world hegemony in different aspects. One aspect of the dispute was in space exploration through the launch of artificial satellites and manned missions (Siddiqi, 2000; Whitfield, 1996). From the period of the conflict to the present, the use of satellites has become increasingly recurrent for various applications, among which we can mention: urban planning, climate monitoring, environmental preservation, and precision agriculture (PA) (Francis et al., 2019).

In general, satellites provide imaging of an area of interest through sensors for different decision-making purposes. In addition, the satellites can have different sensors attached, which will generate multi and hyperspectral images, to provide the most varied type of analysis of the imaged area. On the other hand, the images captured by satellites may present noise that will consequently influence the aforementioned analyses. Among these noises, we can highlight the presence of clouds, shadows, fog, and snow, for example. Thus, the task of identifying and eventually removing such noise for a precise analysis of the satellite image is necessary (Ikeno et al., 2021; Meraner et al., 2020).

In this sense, techniques based on image processing can be used for the task in question, where those based on deep learning stand out. Specifically for the identification task, which can be understood as a segmentation, we highlight the U-Net, which is a deep neural network proposed initially for the segmentation of medical images and also used in other applications (Ronneberger et al., 2015; Eppenhof et al., 2019; Silva et al., 2022).

Recent works have shown that the combination of different backbones can improve the performance of classification networks (e.g., U-Net) in some situations (Zhang et al., 2020). The possibility that backbones are pre-trained could justify this improvement. In addition, it is important to highlight that the training of an algorithm based on deep learning is empirical and will take into account, for example, the variety of hyperparameter configurations and the dataset where the model will be applied for the best fit.

Thus, this work aims to evaluate the U-Net capacity, in three different backbone configurations, in the task of segmenting clouds in multispectral images ob-

Arakaki, L., Silva, L., Silva, M., Melo, B., Backes, A., Escarpinati, M. and Mari, J.
Evaluation of U-Net Backbones for Cloud Segmentation in Satellite Images.
DOI: 10.5220/0011787200003417
In *Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2023) - Volume 4: VISAPP*, pages 452-458
ISBN: 978-989-758-634-7; ISSN: 2184-4321

tained by satellites.

2 THEORICAL BASES

In this section, we present: (i) the theoretical foundation for this work, and (ii) related works describing the state of the art for the cloud detection problem.

2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are one of four categories of deep learning methods, along with constrained Boltzman machines, autoencoders, and sparsecoding. A CNN consists of three types of neuron layers: (i) convolutional layers, (ii) pooling layers, and (iii) fully connected layers (Goodfellow et al., 2016; Guo et al., 2016).

The layers of a CNN are structured hierarchically, with convolutional layers interspersed with pooling layers and fully connected layers at the top of the hierarchy. The sequence of convolutional and pooling layers is responsible for automatically extracting features that can be used to classify these images. The initial layers are responsible for learning the simplest features that are combined in subsequent layers, allowing the learning of increasingly complex features as one approaches the end of the network. The fully connected layers are responsible for classifying the images considering the features learned throughout the architecture (LeCun et al., 2015; Ponti et al., 2017).

2.1.1 U-Net

U-Net is a neural network architecture developed for image segmentation tasks. The U-Net consists of an encoder and decoder structure based on a Convolutional Neural Network. U-Net can be divided into two parts: a contract block and a expand block (Long et al., 2015).

The first part is composed of a sequence of convolution layers with pooling layers. In this block, features are extracted from the input images (convolution layer) and the size of the images is reduced (pooling layer) allowing feature extraction in multi-resolution. Each convolution layer is followed by a batch normalization and a non-linear activation function. The type of pooling used is max-pooling with $stride = 2$, which reduces the output resolution by half (Ronneberger et al., 2015).

The second part performs the expansion in levels, making interconnections between images and equivalent scales. The U-Net architecture has connections between the output of a contraction block and the input of the corresponding expansion block (Ronneberger et al., 2015).

2.1.2 Backbones

A backbone is an element of the network architecture that defines how the layers are organized in the encoder part and thus also determines how the encoder should be constructed. Several backbones can be implemented when using U-Net as architecture, such as VGG, ResNet, Inception, and others (Wang et al., 2020).

Backbones can be used to improve the performance of a CNN due to the ability to extract features in an optimized way, including enabling the use of pre-trained backbones (Ciaparrone et al., 2020).

2.2 Related Works

Mohajerani and Saeedi (2019a) proposed an approach based on a fully connected network to perform the cloud segmentation, known as Cloud-Net. This approach is composed of convolutional blocks and simple layers. Each convolutional block contains addition, concatenation, and copy layers. After each convolution, a ReLu activation function was applied. The authors compared the approach presented with those presented by Zhu et al. (2015) and Mohajerani et al. (2018a). For the evaluation, they considered Jaccard, Precision, Recall, Specificity, and Accuracy indices. The results show that Cloud-Net improved all indexes, except for Recall where the approach proposed by Zhu et al. (2015) performed better.

The work of Francis et al. (2019) also reinforces the importance of correctly detecting clouds in satellite images for better use of these images in different contexts. In this sense, the authors presented a fully convolutional approach inspired by U-Net for cloud segmentation. The approach presented is called CloudFCN and has become state-of-the-art for the problem in question. In general terms, CloudFCN will merge the shallowest layers with their content visible to the deepest layers. The authors demonstrate the effectiveness of CloudFCN with several experiments performed on images obtained by the Carbonite-2 and Landsat 8 satellites.

The cloud detection problem is complex and therefore sensitive to failures, mainly to deal with smaller clouds with sparse distribution (called thin clouds). In this sense, Li et al. (2022) presents the GCDB-UNet, an approach based on the U-Net for cloud segmentation, especially those that are more difficult to be detected. In general terms, GCDB-UNet has two layers for extracting specialized fea-

tures in thin clouds. The authors evaluated the approach through experiments conducted on datasets from three different satellites (Landsat 8, SPARCS, and MODIS). The GCDB-UNet demonstrated robustness and superior performance when compared to other methods.

3 MATERIAL AND METHODS

This section presents the dataset, experimental setup, and computational resources used to conduct this study.

3.1 Image Dataset

The dataset used in the experiments was the 38-Cloud[1], which is composed of 8,400 patches for training and 9,201 patches for testing. Each patch has a size of 384×384 pixels extracted from 38 scenes (of which 18 are for training and 20 for tests) obtained by the Landsat 8 satellite. Each patch is composed of four channels: Red, Green, Blue, and Near-Infrared (NIR). Each patch of the training set comes with a binary reference image (ground-truth) in which the clouds present in the image were manually delineated. The ground-truth of the test set is only provided for the complete scenes and thus the evaluation measures for the test set are computed only for this scenario. Figure 1 shows three examples of scenes (in pseudo-colors) and their respective ground truths. Figure 2 shows three examples of patches used in the experiments extracted from the scenes (Mohajerani et al., 2018b; Mohajerani and Saeedi, 2019b).

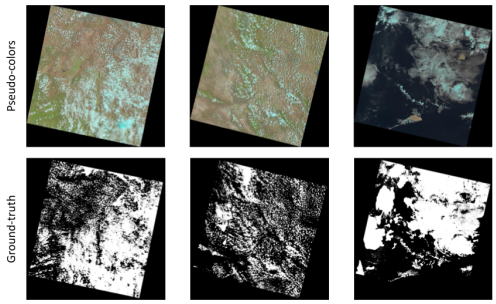

Figure 1: Example of scenes from the 38-Cloud set obtained from the Landsat 8. The first row shows views of the images in pseudo-colors, and the second row shows images of the outlined clouds.

[1]https://github.com/SorourMo/38-Cloud-A-Cloud-Segmentation-Dataset

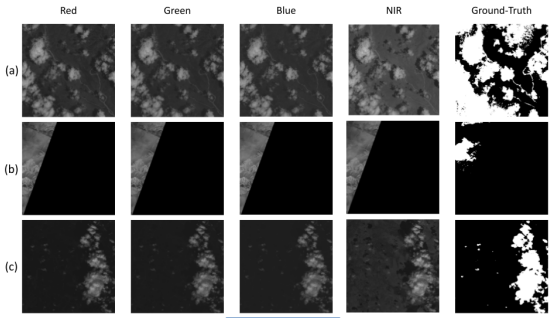

Figure 2: Example of three patches extracted from the scenes. Each patch has four channels and its corresponding ground truth with information about the clouds.

3.2 Experimental Setup

First, the training set was divided into 6,000 patches for training and 2,400 patches for validation. In this way, the test set provided by 38-Cloud remained unchanged during the model training process.

For the experiments, three U-Net configurations were evaluated: (i) U-Net with a simple backbone, (ii) U-Net with a backbone based on VGG-16 (Simonyan and Zisserman, 2014), and (iii) with a backbone based on ResNet-18 (He et al., 2016). Simple U-Net was trained from scratch and models with backbones based on VGG-16 and ResNet-18 were trained by fine-tuning pre-trained models with ImageNet. All models were trained over 50 epochs and a mini-batch with size 16. The optimizer used was Adam and the cost function was cross-entropy. The learning rate for the simple U-Net was 10^{-2} and for the U-Net models with backbones based on VGG-16 and ResNet-18, the value was 10^{-6}, since they were trained using fine-tuning on pre-trained models. Figure 3 shows a summary of the described experimental setup.

To evaluate the performance of our experiments in the segmentation task, the following metrics were used: Precision, Recall, Specificity, Jaccard, and Accuracy. These metrics are based on the relationship between different perspectives of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Equations 1, 2, 3, 4, and 5 present the mentioned metrics.

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

$$Specificity = \frac{TN}{TN+FP} \tag{3}$$

$$Jaccard = \frac{TP}{TP+FP+FN} \tag{4}$$

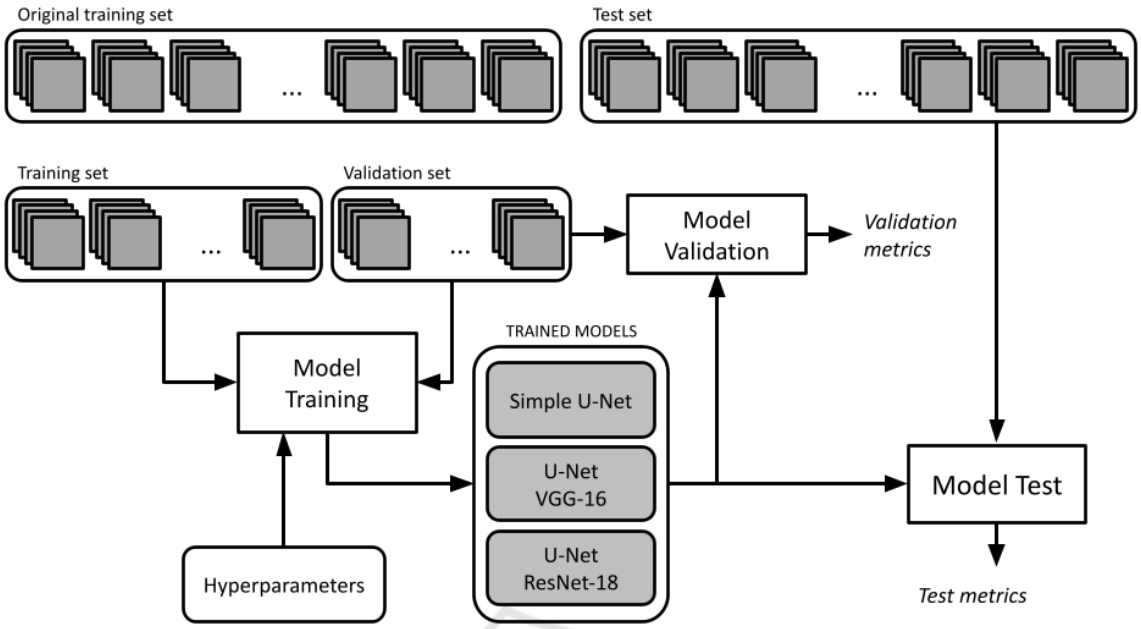

Figure 3: Experimental setup of this work. The three models, the dataset splitting, and the evaluation of the results.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

The experiments were performed on PCs equipped with a 3.0 GHz Core i5-4430 CPU and 32 GB of RAM. Two PCs have NVIDIA GTX 1080 ti GPUs (11 GB memory) and one PC has an NVIDIA Titan Xp GPU (12 GB memory). The experiments were developed using the Python 3.8 programming language and the NumPy and Matplotlib libraries for numerical processing and visualization of images and data. The Scikit-learn library was used to manipulate the set of images and analyze the classification results. The library used to implement the deep neural network models was PyTorch 1.8 and the pre-trained models of VGG-16 and ResNet-18 were obtained from the Torchvision 0.9 module. All random number generation seeds were fixed to ensure reproducibility among the experiments.

4 RESULTS

With the execution of the experiments, we can quantitatively evaluate the results. For the validation set, in terms of accuracy, the U-Net model with VGG-16 backbone obtained the highest result (98.01%). For the same validation set, the Simple U-Net and U-Net models with ResNet-18 backbone had 96.24% and 97.76% of accuracy, respectively.

For the test set, the Simple U-Net obtained better results in the Recall, Jaccard, and Accuracy indexes (84.23%, 74.48%, and 94.56%, respectively). For Precision and Specificity indices, U-Net with VGG-16 backbone obtained the best results (88.17% and 98.75%, respectively). Table 1 presents a summary of the metrics evaluated in each of the models analyzed in this work.

Figures 4, 5, and 6 present the semantic segmentation for the same five patches of the validation set considering Simple U-Net, U-Net (VGG-16), and U-Net (ResNet-18), respectively, for purposes of comparison. In the first row of figures, the segmentation is overlapped over the pseudo-color image (the Red, Green, and Blue channels were combined and converted to grayscale). In the second row, each pixel was colored according to the type of correct or incorrect classification: green: true positive (TP); black: true negative (TN); orange: false positive (FP); and red: false negative (FN).

Finally, Figure 7 shows three complete images belonging to the test set. The first two columns show the pseudo-color image and the ground truth. The other columns show the result of the segmentation performed by the Simple U-Net, U-Net with VGG-16 backbone, and U-Net with ResNet-18 backbone models, respectively.

Table 1: summary of indexes evaluated in each analyzed model in the test set.

Model	Metrics				
	Precision	**Recall**	**Specificity**	**Jaccard**	**Accuracy**
Simple U-Net	86.98%	**84.23%**	96.96%	**74.48%**	**94.56%**
U-Net (VGG-16)	86.15%	58.55%	98.13%	52.63%	82.45%
U-Net (ResNet-18)	**88.17%**	58.28%	**98.75%**	51.19%	82.80%

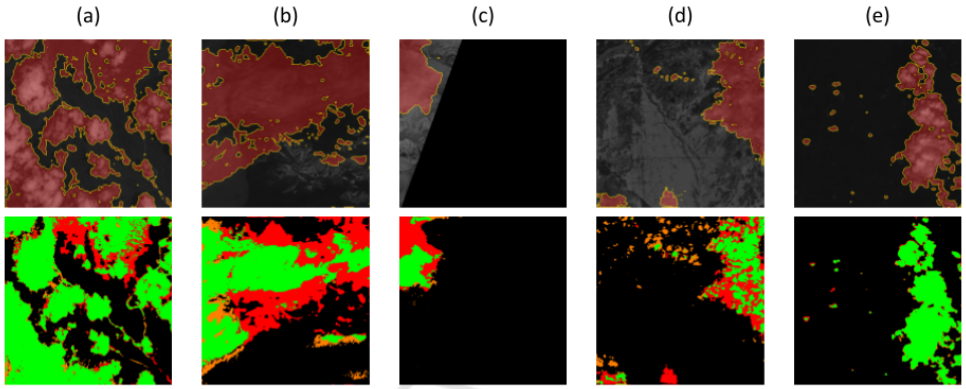

Figure 4: Segmentation results of five patches (a – e) using simple U-Net. The first row shows the segmentation superimposed on the original image (pseudo-colors). The second row shows a map with the evaluation of the segmentations.

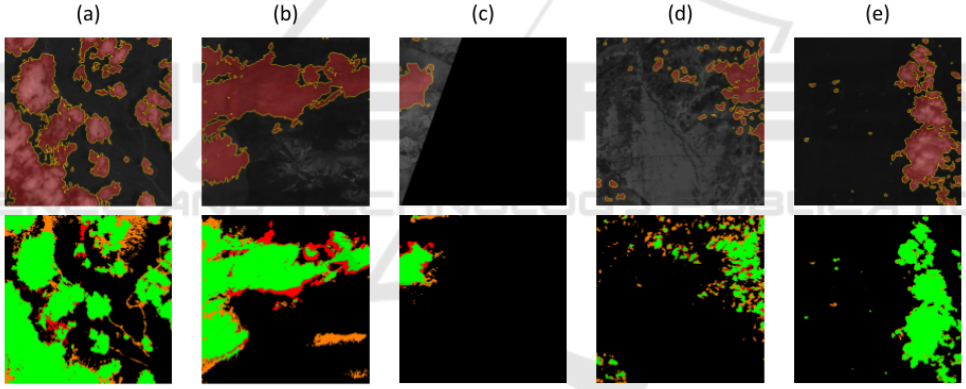

Figure 5: Segmentation results of five patches (a – e) using U-Net (VGG-16). The first row shows the segmentation superimposed on the original image (pseudo-colors). The second row shows a map with the evaluation of the segmentations.

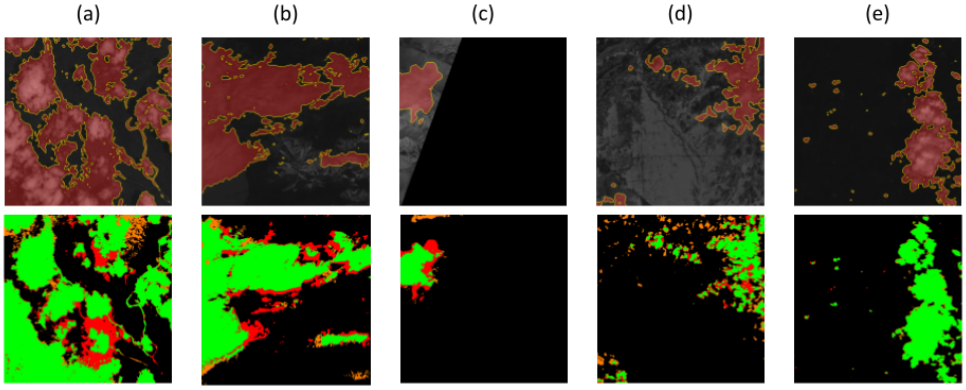

Figure 6: Segmentation results of five patches (a – e) using U-Net (ResNet-18). The first row shows the segmentation superimposed on the original image (pseudo-colors). The second row shows a map with the evaluation of the segmentations.

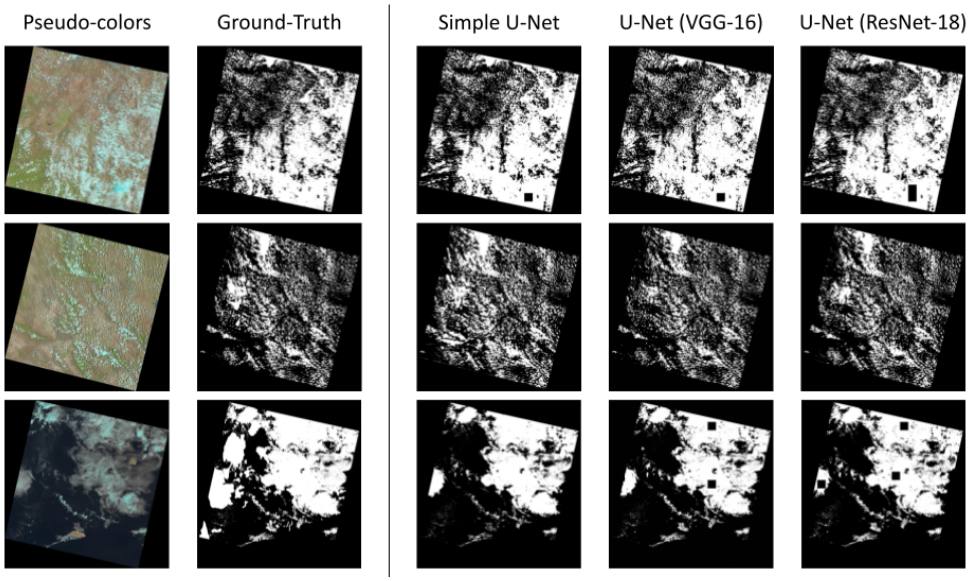

Figure 7: Segmentation result of three scenes extracted from the test set.

5 CONCLUSIONS

In this work, we presented the implementation and comparison of three models based on deep neural networks for satellite image segmentation. The models based on the U-Net architecture with different backbones: (i) a simple backbone with three contraction blocks and three expansion blocks, (ii) a backbone based on VGG-16 pre-trained with the ImageNet image set, and (iii) a backbone based on ResNet-18 also pre-trained with ImageNet. The models were tested using the 38-Cloud image set.

The results indicate that the three implemented and tested methods are capable of segmenting the clouds in satellite images. Simple U-Net performed better for the Recall, Jaccard, and Accuracy indices. On the other hand, when we analyzed Precision and Specificity, better results were noted for the model with ResNet-18 backbone. As deep learning training is related to empirical aspects (e.g., tuning hyperparameters and model selection), our contribution to this work is to initiate an evaluation to find the best model based on deep learning for the segmentation of clouds in satellite images.

5.1 Future Works

As future works, it is intended to: (i) test other models based on deep learning for cloud segmentation, (ii) consider other metrics to evaluate the results, (iii) perform tests with other sets of images, and (iv) provide hyperparameter optimization.

Subsequently, with robust results for the task in question, it is expected to perform the cloud removal task and the consequent image reconstruction in applications in precision agriculture also using techniques based on deep learning. From these tasks, it is possible to estimate more accurate agronomic indices (e.g., water and nitrogen stress, sowing estimate, and future harvest prediction) and thus enhance decision-making in the agricultural sector.

ACKNOWLEDGEMENTS

André R. Backes gratefully acknowledges the financial support of CNPq (National Council for Scientific and Technological Development, Brazil) (Grant #307100/2021-9). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - Finance Code 001. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the TITAN Xp GPU used for this research. Laura G. M. Arakaki received a scholarship from PIBIC/CNPq.

REFERENCES

Ciaparrone, G., Bardozzo, F., Priscoli, M. D., Kallewaard, J. L., Zuluaga, M. R., and Tagliaferri, R. (2020). A comparative analysis of multi-backbone mask r-cnn for surgical tools detection. In *2020 International*

Joint Conference on Neural Networks (IJCNN), pages 1–8. IEEE.

Eppenhof, K. A., Lafarge, M. W., Veta, M., and Pluim, J. P. (2019). Progressively trained convolutional neural networks for deformable image registration. *IEEE Transactions on Medical Imaging*, 39(5):1594–1604.

Francis, A., Sidiropoulos, P., and Muller, J.-P. (2019). CloudFCN: Accurate and Robust Cloud Detection for Satellite Imagery with Deep Learning. *Remote Sensing*, 11(19):2312.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., and Lew, M. S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, 187:27–48.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

Ikeno, K., Fukuda, T., and Yabuki, N. (2021). An enhanced 3d model and generative adversarial network for automated generation of horizontal building mask images and cloudless aerial photographs. *Advanced Engineering Informatics*, 50:101380.

LeCun, Y., Bengio, Y., and Hinton, G. E. (2015). Deep learning. *Nature*, 521(7553):436–444.

Li, X., Yang, X., Li, X., Lu, S., Ye, Y., and Ban, Y. (2022). GCDB-UNet: A novel robust cloud detection approach for remote sensing images. *Knowledge-Based Systems*, 238:107890.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440.

Meraner, A., Ebel, P., Zhu, X. X., and Schmitt, M. (2020). Cloud removal in sentinel-2 imagery using a deep residual neural network and sar-optical data fusion. *ISPRS Journal of Photogrammetry and Remote Sensing*, 166:333–346.

Mohajerani, S., Krammer, T. A., and Saeedi, P. (2018a). Cloud Detection Algorithm for Remote Sensing Images Using Fully Convolutional Neural Networks. arXiv:1810.05782 [cs].

Mohajerani, S., Krammer, T. A., and Saeedi, P. (2018b). Cloud detection algorithm for remote sensing images using fully convolutional neural networks. *arXiv preprint arXiv:1810.05782*.

Mohajerani, S. and Saeedi, P. (2019a). Cloud-Net: An End-To-End Cloud Detection Algorithm for Landsat 8 Imagery. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 1029–1032, Yokohama, Japan. IEEE.

Mohajerani, S. and Saeedi, P. (2019b). Cloud-net: An end-to-end cloud detection algorithm for landsat 8 imagery. In *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 1029–1032. IEEE.

Ponti, M. A., Ribeiro, L. S. F., Nazaré, T. S., Bui, T., and Collomosse, J. (2017). Everything you wanted to know about deep learning for computer vision but were afraid to ask. In *SIBGRAPI Tutorials*, pages 17–41. IEEE Computer Society.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer.

Siddiqi, A. A. (2000). *Challenge to Apollo: the Soviet Union and the space race, 1945-1974*, volume 4408. US National Aeronautics & Space Administration.

Silva, L. H. F. P., Jünior, J. D. D., Mari, J. F., Escarpinati, M. C., and Backes, A. R. (2022). Non-linear co-registration in uavs' images using deep learning. In *2022 35th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, volume 1, pages 1–6.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., and Yeh, I.-H. (2020). Cspnet: A new backbone that can enhance learning capability of cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 390–391.

Whitfield, S. J. (1996). *The culture of the Cold War*. JHU Press.

Zhang, R., Du, L., Xiao, Q., and Liu, J. (2020). Comparison of backbones for semantic segmentation network. In *Journal of Physics: Conference Series*, volume 1544, page 012196. IOP Publishing.

Zhu, Z., Wang, S., and Woodcock, C. E. (2015). Improvement and expansion of the fmask algorithm: Cloud, cloud shadow, and snow detection for landsats 4–7, 8, and sentinel 2 images. *Remote sensing of Environment*, 159:269–277.