Evaluating Multiple Combinations of Models and Encoders to Segment Clouds in Satellite Images

Jocsan Ribeiro da Luz Ferreira[1][a], Leandro Henrique Furtado Pinto Silva[1][b],
Mauricio Cunha Escarpinati[2][c], André Ricardo Backes[3][d] and João Fernando Mari[1][e]
[1] *Institute of Exacts and Technological Sciences, Federal University of Viçosa, Brazil*
[2] *School of Computer Science, Federal University of Uberlandia, Brazil*
[3] *Department of Computing, Federal University of São Carlos, Brazil*

Keywords: Cloud Segmentation, Satellite Images, Deep Learning.

Abstract: This work evaluates methods based on deep learning to perform cloud segmentation in satellite images. Wwe compared several semantic segmentation architectures using different encoder structures. In this sense, we fine-tuned three architectures (U-Net, LinkNet, and PSPNet) with four pre-trained encoders (ResNet-50, VGG-16, MobileNet V2, and EfficientNet B2). The performance of the models was evaluated using the Cloud-38 dataset. The training process was carried out until the validation loss stabilized, according to the early stopping criterion, which provides a comparative analysis of the best models and training strategies to perform cloud segmentation in satellite images. We evaluated the performance using classic evaluation metrics, i.e., pixel accuracy, mean pixel accuracy, mean IoU, and frequency-based IoU. Results demonstrated that the tested models are capable of segmenting clouds with considerable performance, with emphasis on the following values: (i) 96.19% pixel accuracy for LinkNet with VGG-16 encoder, (ii) 92.58% mean pixel accuracy for U-Net with MobileNet V2 encoder, (iii) 87.21% mean IoU for U-Net with VGG-16 encoder, and (iv) 92.89% frequency-based IoU for LinkNet with VGG-16 encoder. In short, the results of this study provide valuable information for developing satellite image analysis solutions in the context of precision agriculture.

1 INTRODUCTION

Agriculture is one of the most critical sectors for humanity. Studies in this area go beyond maximizing global food production, as aspects of the best use of all-natural resources are increasingly preponderant for our society (Pellegrina, 2022). Furthermore, agribusiness significantly moves the Gross Domestic Product of several countries. One of the countries where we can highlight the sector's importance is Brazil, where there is a prominent production of soybeans, coffee, and corn (IBGE, 2023).

The figures above from Brazil come from investments in research and technology for the sector, especially since the Brazilian Agricultural Research Corporation (Embrapa) implementation in 1973 (Embrapa, 2023). Such investments provided the implementation of increasingly robust techniques ranging from planting to harvesting. This solid technological presence can be summarized in what we know as precision agriculture and agriculture 4.0. Precision agriculture (PA) consists of the individualized treatment of each crop area. Thus, to carry out such treatment, PA makes extensive use of mapping and imaging techniques, which can come from different capture sources, such as satellites, unmanned aerial vehicles, and smartphones (Abbasi et al., 2022; da Silva et al., 2023; Silva et al., 2022).

Satellite images, in particular, can be of a multi- and hyperspectral nature, which makes it possible to obtain several relevant agronomic indices for different decision-making processes. However, the images may present artifacts (clouds, shadows, fog, and others) that may occlude interest regions and negatively influence subsequent analyses. One way to mitigate the presence of clouds is to use image processing techniques to segment the area with the presence of these artifacts. Currently, state of the art for this

[a] https://orcid.org/0009-0007-6832-3147
[b] https://orcid.org/0000-0002-5765-5206
[c] https://orcid.org/0000-0002-3792-056X
[d] https://orcid.org/0000-0002-7486-4253
[e] https://orcid.org/0000-0002-9328-8618

Ferreira, J., Silva, L., Escarpinati, M., Backes, A. and Mari, J.
Evaluating Multiple Combinations of Models and Encoders to Segment Clouds in Satellite Images.
DOI: 10.5220/0012506700003660
Paper published under CC license (CC BY-NC-ND 4.0)
In *Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2024) - Volume 3: VISAPP*, pages 233-241
ISBN: 978-989-758-679-8; ISSN: 2184-4321

task consists of semantic segmentation models based on deep learning, e.g., U-Net, DeepLab, and PSP-Net (Ronneberger et al., 2015a; Chaurasia and Culurciello, 2017; Zhao et al., 2017a; Chen et al., 2017). Thus, this work aims to evaluate different experimental configurations, varying architectures, and encoders in segmenting clouds in multispectral satellite images. We considered three deep-learn-based semantic segmentation architectures (U-Net, LinkNet, and PSP-Net) combined with four different pre-trained encoders (ResNet-50, VGG-16, MobileNet V2, and EfficientNet B2). This work continues a previous study, available at (Arakaki et al., 2023).

Our comprehensive and pragmatic experimental setup provides a valuable comparative analysis of the best deep-learning models and training strategies to address the challenge of segment clouds in satellite images. Our results provide useful information for the development of satellite image analysis solutions in the context of precision agriculture.

This paper is organized as follows: After this section introduces the subject, motivation, and objectives, Section 2 summarizes the state-of-the-art of semantic segmentation methods for clouds in satellite images. In Section 3, we describe our material and methods. results are presented and discussed in Section 4, and we present our conclusions in Section 5.

2 RELATED WORK

Mohajerani et al. (Mohajerani et al., 2018a) proposed a cloud segmentation framework based on a fully connected network (FCN) inspired by U-Net. The fully connected encoder is connected to a fully connected decoder with some skip connections. the dataset 38-Cloud was first introduced in this work. In (Mohajerani and Saeedi, 2019), Mohajerani et al. proposed Cloud-Net, a fully connected network intended for cloud segmentation. Cloud-Net is composed of convolutional blocks containing addition, concatenation, and copy layers, followed by ReLu activation functions. Considering Jaccard, Precision, Recall, Specificity, and Accuracy, Cloud-Net improved all indexes when compared with (Mohajerani et al., 2018b).

Gonzales and Sakla (Gonzales and Sakla, 2019) trained and evaluated a model based on U-Net using transfer learning to perform semantic segmentation of clouds in satellite images. Evaluation of the proposed approach used traditional segmentation metrics (e.g., Jaccard, Precision, and Specificity). Experiments were conducted using the 38-Cloud dataset, which considers images of a multispectral nature. In this sense, the proposed approach performed better using the pre-trained ImageNet encoder for three channels (red, green, and blue). In contrast, there is better performance for the Near Infrared (NIR) channel when considering random initialization of weights.

Meraner et al. (Meraner et al., 2020) proposed an approach based on a Residual Convolutional Neural Network (ResNet) to remove clouds in multispectral images from the Sentinel-2 satellite. The model consists of a fully connected architecture, which can perform on input images with arbitrary spatial dimensions during the training process. The approach proposed by the authors was performed on a dataset from the geographic region corresponding to the European continent. To train the approach, such a dataset has images separated geographically and by seasons to have the gold standard for subsequent reconstruction of a region with clouds. In short, the approach proposed by (Meraner et al., 2020) made it possible to remove extremely thick clouds and reconstruct an optical representation of the Earth's surface obstructed in the image by the cloud.

Buttar and Sachan (Buttar and Sachan, 2022) proposed a deep learning-based approach called SE-UNet++ to perform the cloud segmentation problem on the 95-Cloud dataset. In general, SEUNet++ is based on U-Net++ with a lightweight channel attention mechanism. Furthermore, different backbones were tried as encoders for the proposed approach (e.g., ResNet-18, ResNet-34, ResNet-50, ResNet-101, DenseNet-264, CSPNet, and EfficientNet-B8) for performance comparison purposes. The experiments showed that SEUNet++ obtained an Intersection over Union (IoU) value of 91.8%, improving the state of the art by 0.23%. In addition to IoU, SE-UNet++ also performed better in indices such as accuracy, precision, and recall, which generates defined cloud boundaries capable of segmenting thinner cloud layers. Finally, the authors demonstrated that using the transfer learning technique had a practical impact on the task.

(Arakaki et al., 2023) also aimed to evaluate methods based on deep learning (CNNs in particular) for segmenting clouds in satellite images. For this, three models based on the classic U-Net were compared, each with adaptations in their encoders. The three models were called Simple U-Net (with no changes to the basic structure of the traditional network), U-Net with VGG-16 backbone, and U-Net with ResNet-18 backbone. The models were trained using the 38-Cloud dataset and evaluated according to the Recall, Jaccard, Accuracy, Precision, and Specificity metrics. The results showed that U-Net Simples performed better for the Recall, Jaccard, and Accuracy indices. When considering Precision and Specificity,

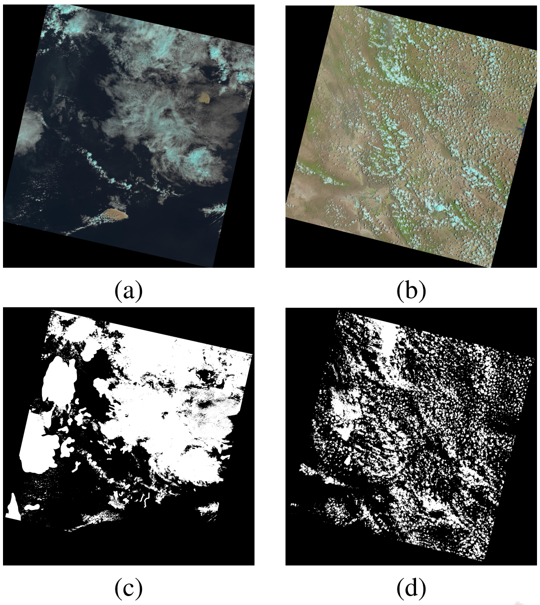

Figure 1: Two samples of the entire scenes available in the 38-Cloud dataset. In pseudo-colors (a - b) and the respective ground truths (c - d).

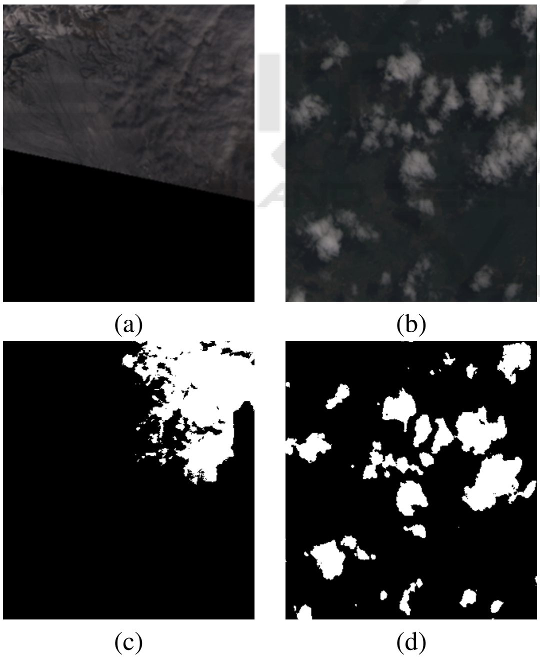

Figure 2: Two samples of the patch images from the training set of the 38-Cloud dataset. In gray-scale (a - b) and the respective ground truths (c - d).

U-Net with ResNet-18 backbone achieved better performance.

3 MATERIAL AND METHODS

3.1 Dataset

The dataset 38-Cloud[1] (Mohajerani et al., 2018a) (Mohajerani and Saeedi, 2019) was considered for the experiments. 38-Cloud is composed of 38 complete scenes obtained by the Landsat 8 satellite, 18 scenes for training and 20 for testing. Each scene image was tiled in patches with 384 × 384 pixels. The training scene images were tiled in 8,400 patches, and the testing scenes in 9,201 patches. The scenes are multispectral images, therefore composed of four channels: Red, Green, Blue, and Near-Infrared (NIR). Each training set patch has a binary reference image (ground truth), with the clouds manually delineated. The test set ground truth is provided only for the complete scenes, and thus, it is necessary to rebuild the scenes after the segmentation to enable segmentation evaluation. Figure 1 shows two scenes from the testing set (in pseudo-colors) and their respective ground truths (binary images). Figure 2 shows two samples of patches and respective ground truths extracted from the scenes of the training set.

3.2 CNN Architectures and Encoders

The literature presents many approaches to performing image segmentation using CNN architectures. To accomplish this work, we selected three of them. These CNNs were selected due to their popularity in many segmentation tasks and good results across the literature. They are: U-Net, LinkNet, and PSPNet.

U-Net is a popular convolutional neural network (CNN) architecture primarily used for image segmentation tasks in the field of computer vision and medical image analysis and in various applications that require pixel-level image classification (Ronneberger et al., 2015b). Its architecture consists of two main parts: the encoding path and the decoding path. The encoding path captures features from the input image at multiple scales. It typically uses a series of convolutional and pooling layers to gradually reduce the spatial dimensions of the input while increasing the number of feature channels. This helps the network learn both low-level and high-level features. The decoding path is a symmetric counterpart to the encoding path. It involves a series of upsampling and convolutional layers, and its goal is to recover the spatial resolution of the input image while also incorporating the learned features from the encoding path. At the

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

end of the decoding path, a convolutional layer is used to produce the final segmentation mask by assigning to each pixel a specific class or category.

LinkNet is another CNN designed for semantic segmentation tasks (Chaurasia and Culurciello, 2017). Similar to U-Net, it also follows an encoder-decoder architecture, where the encoding path extracts features from the input image while the decoding path recovers the spatial information and generates the segmentation mask. LinkNet uses residual blocks, inspired by ResNet, in its architecture to help address the vanishing gradient problem and enable the training of deep networks. They also contribute to the network's ability to capture and learn complex features from the input image.

PSPNet, which stands for Pyramid Scene Parsing Network (Zhao et al., 2017b), is a semantic image segmentation CNN known for its ability to capture global context information effectively, making it particularly suitable for tasks where understanding the relationships between objects and their surroundings is crucial, such as scene parsing and object recognition within images. The core innovation of PSPNet is the Pyramid Pooling Module, which captures multi-scale context information from different regions of the input image. It divides the feature map into a grid of fixed-size bins and applies average pooling in each bin. By doing this at multiple scales, PSPNet gathers contextual information from local to global scales. This helps the network make informed decisions about the class labels of pixels in the image.

These CNNs use the structure of an encoder, also named backbone. The models can be interchanged among multiple encoders. The chosen encoder may impact the number of learnable parameters and the performance of the CNN. Thus, we opt to evaluate different backbones:

- ResNet-50 is a variant of the ResNet (Residual Network) family and is renowned for its depth and skip connections (He et al., 2016). It has 50 layers and employs residual blocks, which enable the training of very deep networks by mitigating the vanishing gradient problem. These residual blocks skip connections, or "shortcuts", that allow the network to learn residual (difference) functions, making it easier to optimize and improve accuracy.
- VGG-16, developed by the Visual Geometry Group at the University of Oxford, is a classic CNN architecture known for its simplicity and effectiveness (Simonyan and Zisserman, 2015). It consists of 16 layers, primarily using small 3×3 convolutional filters stacked on top of each other. The network follows a straightforward pattern, making it easy to understand and modify, and it is often used as a baseline model for various computer vision tasks and serves as a benchmark in image classification.
- MobileNet V2 is a CNN architecture designed for mobile and embedded devices with limited computational resources, created by Google Research (Sandler et al., 2018). It focuses on efficiency and reducing computational demands while maintaining competitive performance. MobileNetV2 utilizes depthwise separable convolutions, which significantly reduce the number of parameters and computations required compared to traditional convolutions.
- EfficientNet B2 is part of the EfficientNet family of CNN architectures (Tan and Le, 2019), known for their impressive scaling principles that balance model depth, width, and resolution to optimize performance and efficiency. EfficientNet B2 is a mid-sized variant with a moderate number of parameters, making it more efficient than larger models while still delivering strong results.

For the experiments, we combined each CNN architecture with all encoders available, thus resulting in a total of 12 (3 architectures and 4 encoders) different combinations.

3.3 Experiment Design

For this study, we used architectures and pre-trained encoders available in the Segmentation PyTorch Library (SMP)[2] (Iakubovskii, 2019), which offers semantic segmentation models such as U-Net, PSPNet, LinkNet, DeepLabV3, among others. SMP is compatible with PyTorch libraries and the Python programming language.

We evaluated our trained models using the 38-Cloud dataset, described in Section 3.1. The 38-Cloud dataset provides a training set composed of 8,400 images with the size of 384×384 pixels, as described in Section 3.1. The dataset also provides a separate test set with 9,201 images to evaluate the quality and generalization capability of the trained models.

We randomly separated 30% of training images to build a validation set. This validation set was used to evaluate the training process, searching for overfitting, and as a parameter for the stopping training early strategy.

We trained all combinations of architectures and encoders using the Adam optimizer with an initial

[2] https://github.com/qubvel/segmentation_models.pytorch

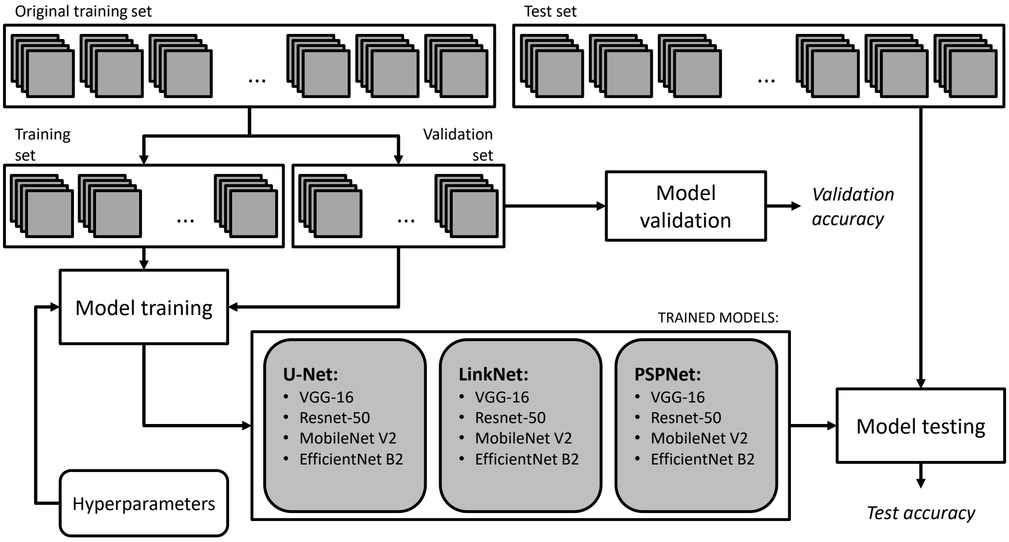

Figure 3: Illustration of the experiment design, including dataset splitting, model training, and evaluation.

learning rate of 0.0001, and the loss function was the cross-entropy. All encoders were pre-trained with ImageNet. We reduced the learning by multiplying it by a factor of 0.1 whenever the validation loss had not improved along 10 epochs; this strategy is called reducing the learning rate on plateaus. If the validation loss does not improve for two complete cycles of learning rate reductions, i.e., a patience value of 21 epochs, we early stop the training process, avoiding the model entering an overfitting state. We used batch sizes of 16 for VGG-16, ResNet-50, and EfficientNet B2 and a batch size of 24 for MobineNet V2.

The experiment design is illustrated in Figure 3.

3.4 Model Evaluation

We evaluate the trained models considering validation and test sets and the metrics describes in Equations 1-4, where k is the number of classes (in this work $k = 2$), n_{jj} is the number of true positives, n_{ij} is the number of false positives, n_{ji} is the number of false negatives, and t_j is the total number of pixels in class j.

The pixel accuracy, *PA*, corresponds to the ratio between true positive classifications and the total number of pixels:

$$PA = \frac{\sum_{j=0}^{k-1} n_{jj}}{\sum_{j=0}^{k-1} t_j} \tag{1}$$

The mean pixel accuracy, *mPA*, considers the mean of the accuracy of each class:

$$mPA = \frac{1}{k}\sum_{j=0}^{k-1} \frac{n_{jj}}{t_j} \tag{2}$$

The mean IoU, *mIoU*, is the mean of the IoU computed over each class and is defined as follows:

$$mIoU = \frac{1}{k}\sum_{j=0}^{k-1} \frac{n_{jj}}{n_{ij}+n_{ji}+n_{jj}} \tag{3}$$

In the frequency-based IoU, *fwIoU*, the mean of the classes IoU are weighted by its frequency, and it is computed as follows:

$$fwIoU = \frac{1}{\sum_{j=1}^{k} t_j}\sum_{j=0}^{k-1} \frac{n_{jj}}{n_{ij}+n_{ji}+n_{jj}} \tag{4}$$

3.5 Computational Environment

Experiments were conducted in three PC computers equipped with an i5 processor with 3.0 GHz and 32 GB of RAM. Two PCs were equipped with GPUs NVIDIA GTX 1080 ti with 11 GB of memory and one with a NVIDIA Titan XP with 12 GB of memory. The operating system was Ubuntu 22.04 LTS, and the experiments were programmed using Python 3.9, PyTorch 2.0.1, torchvision 0.15.12, and CUDA 11.0. The segmentation architectures and pre-trained encoders were from Segmentation Models PyTorch (SMP) 0.3.2. We also used Scikit-learn 1.2.0 and Matplotlib 3.7.1 for plotting.

Table 1: Results of the experiment over the validation and test sets.

		VAL				TEST				
Model	**Encoder**	*PA*	*mPA*	*mIoU*	*fwIoU*	*PA*	*mPA*	*mIoU*	*fwIoU*	**Epochs**
U-Net	**VGG-16**	***0.9799***	***0.9298***	***0.9031***	***0.9648***	**0.9603**	0.9154	***0.8721***	**0.9262**	196
	ResNet-50	0.9782	0.9279	0.8998	0.9623	0.9579	0.9182	0.8676	0.9217	168
	MobileNet V2	0.9779	0.9233	0.8973	0.9618	0.9569	***0.9258***	0.8625	0.9205	139
	EfficientNet B2	0.9745	0.9211	0.8895	0.9568	0.9588	0.9194	0.8496	0.9240	75
PSPNet	**VGG-16**	0.9735	0.9121	0.8811	0.9538	**0.9571**	0.9095	**0.8584**	**0.9206**	132
	ResNet-50	**0.9738**	**0.9144**	**0.8841**	**0.9545**	0.9558	0.9029	0.8501	0.9181	129
	MobileNet V2	0.9647	0.8975	0.8642	0.9418	0.9433	0.9079	0.8080	0.9002	92
	EfficientNet B2	0.9651	0.8985	0.8655	0.9424	0.9465	**0.9169**	0.8166	0.9057	78
LinkNet	**VGG-16**	0.9785	0.9291	0.9007	0.9629	***0.9619***	0.9126	**0.8705**	***0.9289***	147
	ResNet-50	**0.9788**	**0.9296**	**0.9009**	**0.9633**	0.9584	**0.9206**	0.8606	0.9228	151
	MobileNet V2	0.9758	0.9215	0.8913	0.9584	0.9543	0.9074	0.8503	0.9159	115
	EfficientNet B2	0.9720	0.9181	0.8850	0.9532	0.9548	0.9127	0.8540	0.9166	111

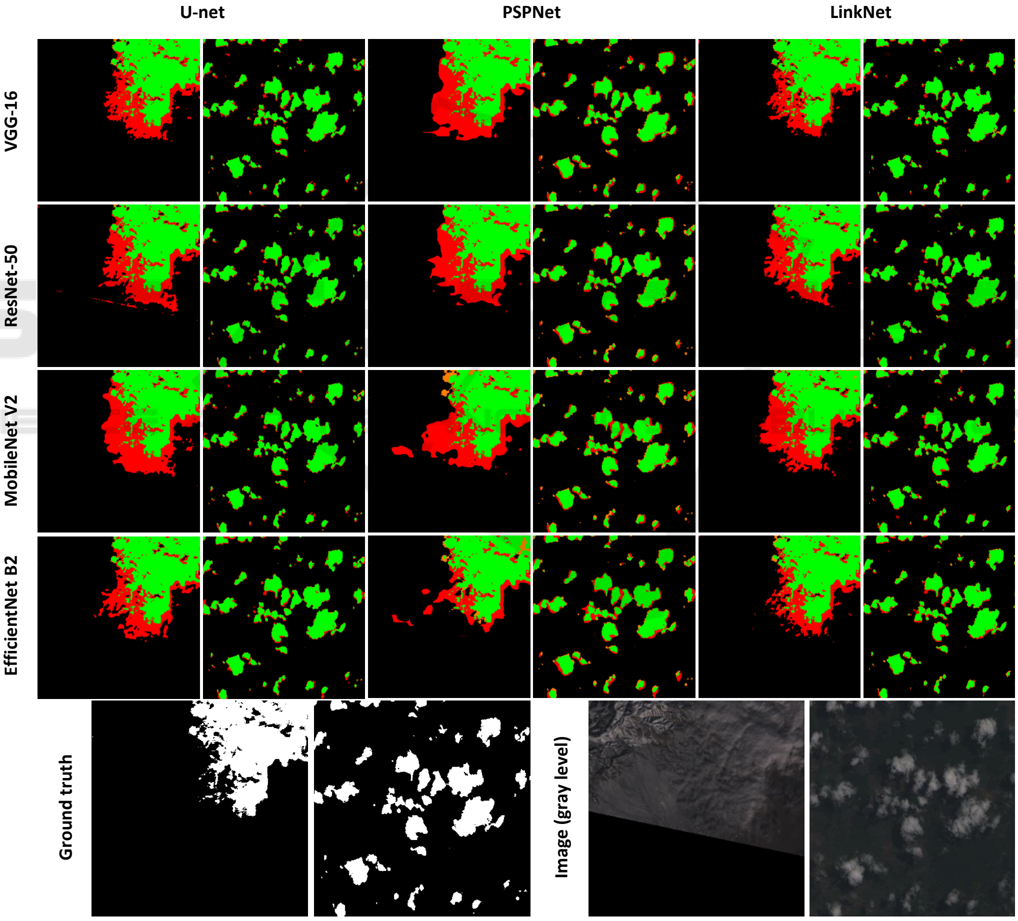

Figure 4: Evaluation maps of the segmentation of two images (patches) taken from the validation set.

4 RESULTS AND DISCUSSION

Table 1 shows the results obtained over the validation and test sets in terms of accuracy, mean accuracy, mean IoU, and frequency-based IoU. The values for the validation set (VAL columns) are the means for the indexes obtained from each one of the 384 × 384 pixels patches. Values for the test set (TEST columns)

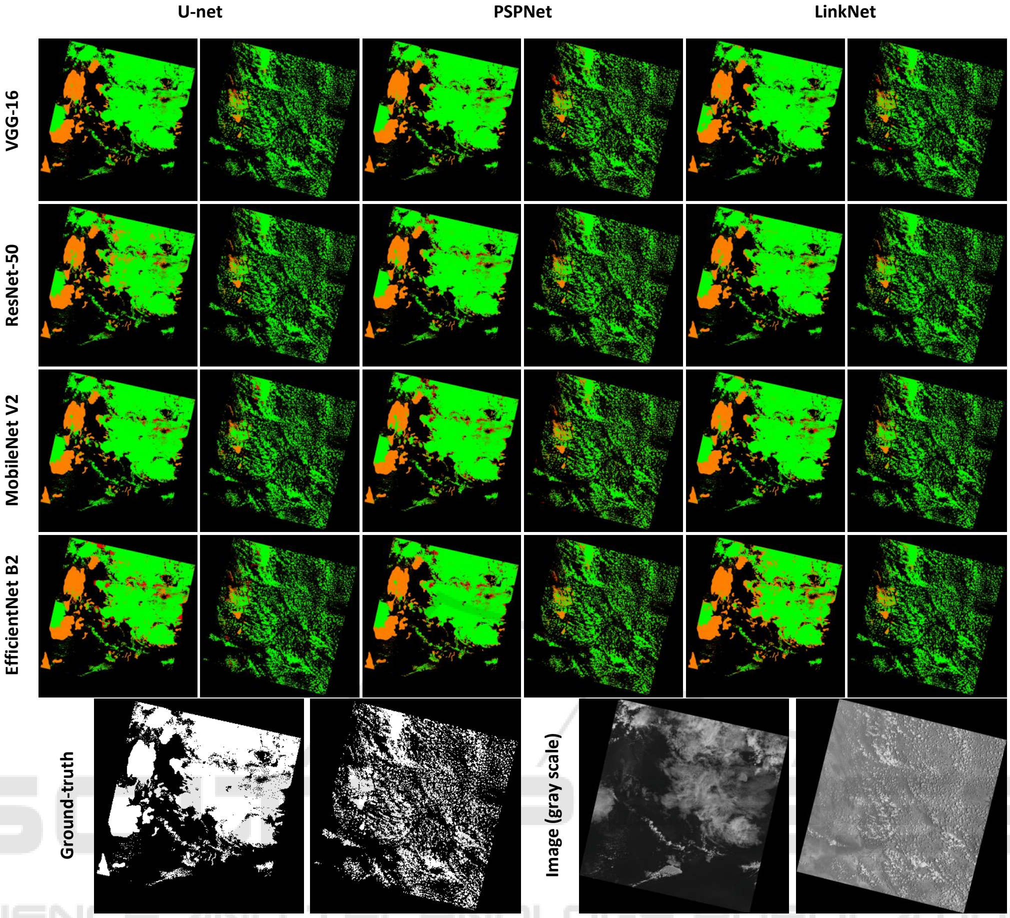

Figure 5: Evaluation maps of the segmentation of two complete images (entire images) taken from the test set.

are the means for the indexes obtained for each scene, reconstructed from the segmented patches. Table 1 also shows the number of training epochs of the model, i.e., when the early stopping strategy was activated. We marked in bold the best value for each metric in a CNN model. We also marked in italics the best metric value considering all combinations of architecture and encoder.The best values obtained over the test set were 96.19% pixel accuracy for LinkNet with VGG-16 encoder, 92.58% mean pixel accuracy for U-Net with MobileNet V2 encoder, 87.21% mean IoU for U-Net with VGG-16 encoder, and 92.89% frequency-based IoU for LinkNet with VGG-16 encoder.

Results show that, as expected, deeper network encoders, such as VGG-16 and ResNet-50, require more training epochs. They also present a slightly superior performance compared to shallow encoders, regardless of the CNN architecture used. Though U-Net is a simpler CNN model than the compared ones, it maintains similar performance for the same encoder in both test and validation sets. This is not true for other architecture, where ResNet-50 obtains the best results for the validation set, but VGG-16 is preferred in the test set (PSPNet). This behavior indicates that U-Net, in combination with the VGG-16 encoder, has a greater generalization capacity for learned features.

Figure 4 shows, for each combination of architecture and encoder, a map with the evaluation of the segmentation for two patches taken from the validation set. The maps have each pixel colored according to the type of correct or incorrect classification: true positives (TP) are green; true negatives (TN) are black; false positives (FP) are red; and false negatives (FN) are orange. In the last row, we show the ground truth of the images and the gray-level version of the original patches. Independent of the combination of CNN architecture and encoder, we notice a tendency to classify the background as cloud more often than the opposite. An analysis of the two patches shows that misclassification is due to the characteristics of the terrain in the image. Images containing irregular terrain tend to generate a higher rate of false negatives, i.e., more terrain areas are confused with cloud regions.

In Figure 5, we show the evaluation maps for two entire scenes taken from the test set. As in Figure 4, we colored each pixel according to the type of correct or incorrect classification: true positives (TP) are green; true negatives (TN) are black; false positives (FP) are red; and false negatives (FN) are orange. In the last row, we show the ground truth of the images and the gray-level version of the original patches. When considering the entire map, we notice that all combinations of architecture and encoder miss large chunks of clouds and classify them as terrain, thus resulting in a large rate of false negatives (orange color) in some maps. This usually happens when the map contains cirrus clouds, which present a delicate and wispy appearance with white strands. This type of cloud allows us to see through it, so cloud and terrain information are mixed, thus explaining the poor CNN results in these regions.

5 CONCLUSIONS

In this paper, we addressed the problem of cloud segmentation in satellite images using deep learning. We investigated three traditional semantic segmentation networks, namely U-Net, LinkNet, and PSPNet. We also evaluated four different encoders (ResNet-50, VGG-16, MobileNet V2, and EfficientNet B2) on the segmentation results of each network. These encoders are responsible for extracting meaningful features from the input image and have a direct impact on the network performance. Results showed that deeper network encoders, such as VGG-16 and ResNet-50, present a slightly superior performance compared to shallow encoders, regardless of the CNN architecture used. In terms of architecture, U-Net performed better in comparison to other CNN models, and it was capable of providing a better generalization of learned features between test and validation sets. In future work, we intend to address the problem of false negatives (clouds classified as background) due to the inability of the network to correctly detect clouds presenting a delicate and wispy appearance.

ACKNOWLEDGEMENTS

André R. Backes gratefully acknowledges the financial support of CNPq (Grant #307100/2021-9). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. Jocsan R. da Luz Ferreira received a scholarship from PIBIC/CNPq. We gratefully acknowledge the support of NVIDIA Corporation, USA with the donation of the TITAN Xp GPU used for this research

REFERENCES

Abbasi, R., Martinez, P., and Ahmad, R. (2022). The digitization of agricultural industry–a systematic literature review on agriculture 4.0. *Smart Agricultural Technology*, 2:100042.

Arakaki, L. G., Silva, L. H., da Silva, M. V., Melo, B. M., Backes, A. R., Escarpinati, M. C., and Mari, J. F. (2023). Evaluation of u-net backbones for cloud segmentation in satellite images. In *VISIGRAPP (4: VISAPP)*, pages 452–458.

Buttar, P. K. and Sachan, M. K. (2022). Semantic segmentation of clouds in satellite images based on u-net++ architecture and attention mechanism. *Expert Systems with Applications*, 209:118380.

Chaurasia, A. and Culurciello, E. (2017). Linknet: Exploiting encoder representations for efficient semantic segmentation. In *2017 IEEE Visual Communications and Image Processing (VCIP)*, pages 1–4. IEEE.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848.

da Silva, M. V., Silva, L. H., Junior, J. D. D., Escarpinati, M. C., Backes, A. R., and Mari, J. F. (2023). Generating synthetic multispectral images using neural style transfer: A study with application in channel alignment. *Computers and Electronics in Agriculture*, 206:107668.

Embrapa (2023). Sobre a embrapa. Available at: https://www.embrapa.br/sobre-a-embrapa. Access at: 07/28/2023.

Gonzales, C. and Sakla, W. (2019). Semantic segmentation of clouds in satellite imagery using deep pre-trained u-nets. In *2019 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pages 1–7. IEEE.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Iakubovskii, P. (2019). Segmentation models pytorch. https://github.com/qubvel/segmentation_models.pytorch.

IBGE (2023). Pib cresce 1,9% no 1º trimestre de 2023. Available at: https://agenciadenoticias.ibge.gov.br/agencia-sala-de-imprensa/2013-agencia-de-noticias/releases/37029-pib-cresce-1-9-no-1-trimestre-de-2023. Access at: 07/28/2023.

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 preprint arXiv:1810.05782*.

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

Mohajerani, S. and Saeedi, P. (2019). 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.

Pellegrina, H. S. (2022). Trade, productivity, and the spatial organization of agriculture: Evidence from brazil. *Journal of Development Economics*, 156:102816.

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

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

Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., and Chen, L. (2018). Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. *CoRR*, abs/1801.04381.

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. IEEE.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition.

Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017a). Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. (2017b). Pyramid Scene Parsing Network. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, page 6230–6239. IEEE.