AI

MDPI

Article

A Comparative Analysis of Deep Learning-Based Segmentation Techniques for Terrain Classification in Aerial Imagery

Martina Formichini * and Carlo Alberto Avizzano

Institute of Mechanical Intelligence, Via Alamanni, 13b, 56010 Ghezzano, Pisa, Italy; carloalberto.avizzano@santannapisa.it

* Correspondence: martina.formichini@santannapisa.it; Tel.: +39-328-5507760

Abstract

Background: Deep convolutional neural networks (CNNs) have become widely popular for many imaging applications, and they have also been applied in various studies for monitoring and mapping areas of land. Nevertheless, most of these networks were designed to perform in different scenarios, such as autonomous driving and medical imaging. Methods: In this work, we focused on the usage of existing semantic networks applied to terrain segmentation. Even though several existing networks have been used to study land segmentation using transfer learning methodologies, a comparative analysis of how the underlying network architectures perform has not yet been conducted. Since this scenario is different from the one in which these networks were developed, featuring irregular shapes and an absence of models, not all of them can be correctly transferred to this domain. Results: Fifteen state-of-the-art neural networks were compared, and we found that, in addition to slight differences in performance, there were relevant differences in the numbers and types of outliers that were worth highlighting. Our results show that the best-performing models achieved a pixel-level class accuracy of 99.06%, with an F1-score of 72.94%, 71.5% Jaccard loss, and 88.43% recall. When investigating the outliers, we found that PSPNet, FCN, and ICNet were the most effective models. Conclusions: While most of this work was performed on an existing terrain dataset collected using aerial imagery, this approach remains valid for investigation of other datasets with more classes or richer geographical extensions. For example, a dataset composed of Copernicus images opens up new opportunities for large-scale terrain analysis.

Keywords: convolutional neural networks; deep learning; remote sensing; computer vision; precision agriculture

Academic Editor: Giovanni Diraco

Received: 25 April 2025
Revised: 22 June 2025
Accepted: 23 June 2025
Published: 3 July 2025

Citation: Formichini, M.; Avizzano, C.A. A Comparative Analysis of Deep Learning-Based Segmentation Techniques for Terrain Classification in Aerial Imagery. *AI* **2025**, *6*, 145. https://doi.org/10.3390/ai6070145

1. Introduction

Due to the effective use and high performance of agricultural image processing, semantic segmentation has gained recognition as a significant research topic. It has been employed extensively in numerous agricultural domains in recent years [1–4]. Deep learning (DL) techniques are essential in the agriculture sector and in mapping land cover using high-spatial-resolution (HSR) remote sensing [5,6]. In particular, monitoring land cover is crucial for natural resource management, and these high-resolution images, known as remote sensing images, can be utilized extensively for agricultural and land cover classifications [7]. Several technological sectors, including precision agriculture [8], urban planning [9,10], environmental protection, land resource management, and land classification, take great advantage of remote sensing.

The literature on semantic segmentation is extensive. It includes many datasets, and we will give some examples of the most popular. One is SpaceNet MVIO, an open-source Multiview Overhead Imagery dataset [11] used for segmentation and object detection tasks. The related study produced outcomes in three areas:

- Expanding object detection and segmentation models to previously unseen resolutions.
- Detecting buildings.
- Confirming whether resolution adjustments for segmentation and object detection models had an impact and researching said impacts and their consequences.

We also highlight a work by Nadir Bengana [12], who aimed to apply domain adaptation (DA) to address the difficulties associated with fully utilizing satellite imagery, which are attributed to the fact that deep neural networks require a substantial amount of labeled data. Furthermore, the appearance of land cover varies from region to region; thus, labeled data from one place may not always be useful for mapping other areas. Additionally, satellite images are available in a variety of multispectral bands, ranging from RGB to more than 12 bands. The tests demonstrated significant improvements over the results obtained without data augmentation (DA). Linus Scheibenreif et al. [13] described a self-supervised, pretrained Swin transformer for segmenting and classifying land cover. Their self-supervised method led to continuous performance improvements in various downstream tasks, yielding notable gains in a low-data regime.

One dataset commonly used for semantic segmentation of agricultural patterns is Agriculture-vision, which comprises extensive aerial field images. Chiu et al. [14] analyzed this dataset using popular semantic segmentation models, specifically DeeplabV3+, gaining 43.66% of the mIoU.

A more effective Unet model was presented by Xin Zhao et al. [15] for applying segmentation to rice lodging, while Jadhav, J.K. et al. [3] introduced SOMs and a DeepLab CNN to segment remote sensing data in agriculture. Multilayer perceptrons (MLPs) and Random Forest (RF) classifiers are employed in work related to classification of crop types and land cover [4]. Semantic segmentation has been applied to unmanned aerial vehicle (UAV)-acquired images [16] by comparing various approaches, including DeeplabV3+, EfficientNet, Feature Pyramid Networks (FPNs), and Adversarial Generative Networks (AGNs). The author's approach uses AgriSegNet, a model whose backbone characteristics were extracted from DeepLabV3+.

Other works describe application of U-Net to segmentation in wheat agriculture [17] and application of an adapted version of VGG-16, a deep neural network, for semantic segmentation of mixed crops [18]. Many works in the literature address segmentation of crops, weeds, and background, of which we give two examples:

- An encoder–decoder network was trained to recognize weed, harvest, and background classes [19], and three image enhancement methods were investigated to improve the model's robustness under varying lighting conditions.
- A U-Net-based deep learning model was trained to segment weed and crop classes [20], achieving a mean Intersection over Union (mIoU) of 88.3%.

Revisiting the study of land cover, Javiera Castillo-Navarro is recognized for her work [21] on the MiniFrance suite, a dataset specifically designed for semi-supervised learning. In the corresponding study, two models (BerundaNet and W-Net) were employed to perform semantic segmentation, achieving a mean intersection over union (mIoU) of 21.20%.

Monitoring and assessing land cover and land use are essential practices in agriculture [14], and remote sensing data provides valuable support for monitoring landscapes and agricultural activities. The LandCover.ai dataset plays a significant role in this context

by offering high-resolution, annotated aerial imagery that facilitates detailed land cover classification. Its unique image patterns closely resemble key landscape features relevant to agriculture. It also enables the study of urbanization rates, deforestation, and agricultural intensity [8–10,14,22].

Although land mapping studies can be conducted using transfer learning with existing networks, a comparative analysis evaluating how transfer learning performs across different neural network architectures has yet to be undertaken. Such an analysis will be crucial for accurately assessing its applicability and effectiveness within the agricultural domain.

The objective of this research was to identify the most effective neural network models for agricultural image segmentation by analyzing outliers and evaluating the performance of existing semantic segmentation models on land cover image patterns that closely resemble key landscape features relevant to agricultural analysis. Deep learning-based semantic segmentation has recently demonstrated outstanding performance in domains such as medical imaging and autonomous driving. However, precision agriculture and environmental monitoring present unique challenges for semantic segmentation, including the following:

- The irregular field shapes commonly found in agricultural scenes;
- Complex and heterogeneous vegetation patterns;
- Varying spectral signatures due to seasonal changes.

Models developed in other domains may not be directly applicable to agricultural use cases. This highlights the need for a systematic evaluation of which deep neural networks are best suited to address the specific requirements of agricultural image segmentation. This work aimed to guide future applications of artificial intelligence tools for large-scale land monitoring, precision agriculture, and related environmental applications.

This research was conducted using the LandCover.ai dataset, available at https://landcover.ai.linuxpolska.com/ (accessed on 24 June 2025), as proposed by Adrian Boguszewski [7], to evaluate performance in terms of accuracy, intersection over union (IoU), and recall and to investigate outliers across fifteen neural networks. The analysis of outliers contributed to improved IoU scores, revealing that many errors in terrain classification stemmed from inaccuracies in the ground-truth annotations, as discussed in Section 3. In his original work, Boguszewski limited his investigation to a single neural network and evaluated only the IoU metric, reporting a value of 85.56%.

The goal of the present study was to extend these findings to segmentation of cultivated areas containing various crop types. The LandCover.ai dataset was selected because it includes classes deemed relevant for this task. Given the characteristics of the images, we expected that the results of soil segmentation obtained with LandCover.ai would be comparable to those for agricultural regions.

This work was organized into the following steps:

- First, fifteen state-of-the-art neural networks were selected based on their performance. These models were implemented using the MMSegmentation toolbox [23] for Python and applied to semantic segmentation. Evaluation metrics such as accuracy, F1-score, intersection over union (IoU), and recall were computed.
- Second, we employed statistical tools (accuracy, F1-score, IoU, and recall) to analyze the results and identify the networks that demonstrated the highest performance.
- Third, confusion maps were generated to assess the type and severity of potential misclassifications among the outliers. A comparative analysis was then conducted by examining both the quantity and quality of the errors produced by each network. Outliers were defined as image regions that exhibited substantial segmentation errors, deviating from the typical model behavior. Finally, we provide a summary of the top-performing networks characterized by low numbers of significant segmentation errors.

The novelties of this work are summarized by the three main contributions:

- A large-scale and consistent comparative analysis across multiple models;
- The introduction of a structured outlier error taxonomy based on confusion maps and IoU thresholds;
- A practical evaluation of computational efficiency to identify suitable candidates for embedded or real-time applications.

For practical applications in precision agriculture and related fields, this comprehensive investigation offers both theoretical insights and operational guidance. The remainder of this paper is organized as follows:

- Section 2 outlines the segmentation methods employed;
- Section 3 describes the methodological framework;
- Section 4 presents the experimental results along with the discussion;
- Section 5 concludes this work and suggests future research directions;
- Appendix A provides a detailed justification for the 1000-pixel threshold used to filter marginal predictions in the confusion maps;
- Appendix B evaluates the impact of using pretrained models and discusses the rationale behind choosing a training-from-scratch strategy.

2. Models' Background

In this research study, semantic segmentation of the LandCover.ai dataset was performed using Python, with 15 networks trained using the MMSegmentation tools [23] within the PyTorch framework (version 2.1.1). MMSegmentation, an open-source toolbox designed for consistent implementation and evaluation of semantic segmentation techniques, provided the computational environment for this study. Most widely known semantic segmentation models and datasets have well-optimized fine-tuning configurations available within MMSegmentation.

We selected the most promising neural networks available in the MMSegmentation framework, developed between 2014 and 2021, focusing specifically on architectures based on purely convolutional networks and excluding those based on transformers. Accuracy, intersection over union (IoU), and recall were used as evaluation metrics. They were computed as mean values, both overall and across the five defined classes.

A description of the networks used in this study is provided below.

2.1. Asymmetrical Non-Local Neural Network for Semantic Segmentation (ANN) [24]

In the Asymmetric Non-local Neural Network, semantic segmentation spatial feature extraction is enhanced through application of non-local operations that capture long-range dependencies. This improves segmentation accuracy by enabling the model to attend to both local and global contextual information within an image. An Asymmetric Pyramid Non-local Block (APNB) and an Asymmetric Fusion Non-local Block (AFNB) are integrated into a pyramid sampling module, significantly reducing computation time and memory usage while still achieving competitive results. Combining features at different levels further contributes to performance improvement.

2.2. Attention Pyramid Context Net [25]

Point cloud processing focuses on optimizing deep semantic segmentation networks by incorporating contextual features through attention mechanisms, thereby improving performance via application of APCNet (Attention Pyramid Context Network).

Multiple well-designed Adaptive Context Modules (ACMs) are embedded within the multiscale contextual representations generated by APCNet. Each ACM evaluates the

local affinity coefficient associated with its corresponding subregion, from which a context vector is computed.

2.3. BiSeNetV2 [26]

BiSeNetV2 is a real-time semantic segmentation model that enhances feature extraction through two complementary paths: a Detail Branch for capturing spatial information and a Semantic Branch for encoding contextual features. Its architecture preserves fine-grained details typically lost in standard pipelines, improving segmentation precision. The two branches are fused using an efficient aggregation mechanism that maintains both resolution and semantic richness. This dual-path design enables high accuracy with low computational cost. As a result, BiSeNetV2 achieves fast and accurate segmentation, suitable for real-time applications.

2.4. Criss-Cross Net [27]

In a Criss-Cross Network (CCNet), a novel criss-cross attention module captures contextual information for each pixel along its criss-cross path. By performing an additional recurrent operation, each pixel can ultimately model dependencies across the entire image.

In general, CCNet offers the following advantages:

1. It is GPU memory-efficient—the proposed recurrent criss-cross attention module consumes 11 times less GPU memory than the non-local block;
2. It provides computational efficiency, reducing FLOPs by approximately 85% compared to the non-local block;
3. It achieves state-of-the-art performance in semantic segmentation by effectively modeling the global context with minimal computational overhead.

2.5. Dual Attention Net [28]

The goal of focusing on relevant image features and enhancing them can be achieved using the Dual Attention Network (DANet), which integrates both spatial and channel-wise attention mechanisms.

By employing DANet for scene segmentation, it becomes possible to accurately determine the positions of various objects within an image. Detailed dependencies are effectively captured in both the spatial and channel dimensions through the use of a position attention module and a channel attention module.

2.6. DeepLabV3+ [29]

Deep neural networks for semantic segmentation tasks can employ a Spatial Pyramid Pooling module to encode multiscale contextual information by processing incoming features, or they can adopt an encoder–decoder structure to capture fine object boundaries and reconstruct detailed spatial information. DeepLabV3+ combines the advantages of both approaches by incorporating a decoding module into the original DeepLabV3 architecture, enabling more accurate and well-defined object boundaries in a segmented scene.

2.7. Fast Fully Convolutional Network [30]

FastFCN is a semantic segmentation model that combines the effectiveness of fully convolutional networks (FCNs) with fast and high-quality output. It employs a unique lightweight encoder–decoder architecture that reduces computational complexity while maintaining performance. FastFCN introduces a novel dilated convolution mechanism to expand the receptive field, thereby enhancing contextual understanding. The model also improves segmentation accuracy through a more efficient feature fusion strategy that enhances fine-grained details, making FastFCN well-suited for real-time segmentation tasks, particularly on resource-constrained devices.

2.8. Fast-SCNN [31]

A Fast Semantic Segmentation Convolutional Neural Network (Fast-SCNN) is designed to proficiently match pixel-wise class labels to images while diminishing computational complexity. Common techniques are employed, such as dilated convolutions, reduced network depth, and lightweight architectures for faster processing.

A fast segmentation network for real-time scene understanding is the focus of R. R. K. Poudel's work. This network is required when images must be analyzed rapidly in order to provide a prompt reaction or to develop a rapid or binding action based on the environmental situation. Furthermore, the work shows that further applications to auxiliary tasks do not require additional pretraining once the model has been suitably trained.

2.9. Fully Convolutional Network [32]

A fully convolutional network (FCN) is a deep learning architecture designed for semantic image segmentation. Unlike traditional CNNs, which produce fixed-size feature maps, FCNs replace fully connected layers with convolutional layers, enabling the network to generate pixel-wise predictions. FCNs adopt an encoder–decoder architecture, where the encoder captures high-level features and the decoder performs upsampling to produce segmentation maps. Skip connections are commonly employed to retain fine-grained spatial information from earlier layers. FCNs are widely utilized in tasks requiring dense, pixel-level classification, such as scene understanding and medical image analysis.

2.10. Global Context Net [33]

Segmentation accuracy can be enhanced by incorporating global context through a self-attention mechanism, as demonstrated by GCNet (Global Context Network), which captures long-range dependencies within an input image.

GCNet is able to model these long-range dependencies and connections between regions or scenes separated by significant distances or time intervals, achieving comparable accuracy with reduced computational cost. This is accomplished by simplifying the network under the assumption that many queries are similar across different regions of an image and by focusing the analysis on image areas where changes occur.

2.11. Image Cascade Network [34]

The Image Cascade Network (ICNet) is a semantic segmentation model designed to achieve efficient real-time performance by progressively refining predictions through a multi-resolution approach. It employs a cascade architecture to process images at different resolutions, effectively balancing accuracy and computational speed. ICNet enables accelerated semantic segmentation without a significant reduction in quality, making it particularly suitable for applications such as autonomous driving and robotic interaction. Furthermore, by leveraging a novel framework that conserves operations across multiple resolutions and a robust fusion unit, ICNet achieves an optimal trade-off between accuracy and efficiency.

2.12. ISANet [35]

An Image Spatial Attention Network (ISANet) enhances focus on salient regions of an image by learning spatial attention maps. It improves feature representation for object detection and image classification. ISANet emphasizes the effectiveness of the self-attention mechanism in the context of semantic segmentation. The attention modules within its structure compute sparse affinity matrices. Processing high-resolution feature maps reduces both computational and memory complexity.

2.13. Object Contextual Representations Net [36]

The Object Contextual Representations Network (OCRNet) enhances object segmentation by incorporating object-context knowledge through adaptive contextual learning and by integrating object-level contextual information. This approach captures detailed object boundaries and long-range dependencies, thereby improving segmentation accuracy. OCRNet operates in three main stages: first, a deep network generates an initial soft segmentation; second, a representation for each object region is estimated by aggregating the features of pixels within the corresponding region; and third, object-contextual representations are employed to refine each pixel's feature representation.

2.14. Pyramid Scene Parsing Net [37]

The Pyramid Scene Parsing Network (PSPNet) is a deep learning model designed for semantic segmentation. It employs a pyramid pooling module to capture multiscale contextual information from an image. Segmentation accuracy is improved by considering both local and global contextual information across multiple spatial scales.

PSPNet effectively integrates relevant global features. The pixel-level feature representation is augmented by a specially designed global pyramid pooling module, in addition to the conventional upsampled fully convolutional network (FCN) used for pixel-wise prediction. The combination of local and global contextual cues contributes to producing more consistent and accurate segmentation results.

2.15. UPerNet [38]

With Unified Perceptual Parsing (UPerNet) for scene understanding, multiple scene interpretation tasks are integrated into a single model. UPerNet simultaneously performs semantic segmentation, depth estimation, and object recognition, enabling comprehensive scene analysis.

To closely approximate the complexity of human visual perception, Unified Perceptual Parsing serves as a multi-task framework capable of simultaneously recognizing a wide range of objects and contextual situations within an image.

2.16. Criteria for Network Selection

We considered multiple factors when selecting the 15 neural network models evaluated in this study. First, we focused on architectures implemented within the MMSegmentation framework [23], which ensured consistent training configurations and reproducibility through standardized, well-maintained implementations. Second, we selected models that addressed deployment constraints across a broad range of CNN-based design paradigms: encoder–decoder structures (e.g., FCN and UPerNet); attention-based approaches (e.g., DANet and CCNet); and lightweight, real-time models (e.g., Fast-SCNN).

Transformer-based models were intentionally excluded to restrict the scope of this study to convolutional methods, which remain widely adopted in real-world precision agriculture applications due to their computational efficiency.

Although not exhaustive, this selection encompassed the most frequently cited and practically relevant CNN-based segmentation networks in the literature. Our objective was to provide a fair, clear, and robust comparison of these convolutional models specifically for the task of agricultural land cover segmentation.

3. Materials and Methods

3.1. Data Preparation

The first step involved data preparation, in which the data were processed and transformed from their original forms into one more suitable for subsequent analysis [39].

The LandCover.ai dataset comprises 8 images with a spatial resolution of 50 cm (approximately 4200 × 4700 pixels) and 33 images with a resolution of 25 cm (approximately 9000 × 9500 pixels). This dataset [7] was selected due to the high diversity of land cover elements it contains, such as forests, agricultural areas, buildings, roads, and water bodies, as well as a wide variety of vegetation types.

Following the approach of Adrian Boguszewski and considering that most networks accept input dimensions ranging from 200 to 1024 pixels, we divided the 41 original images and their corresponding masks into smaller patches of 512 × 512 pixels. This resulted in a total of 10,674 image–mask pairs, excluding any patches smaller than the target size. Figure 1 presents a sample of the dataset after splitting the original images.

Figure 1. Sample images taken from the LandCover.ai dataset.

3.2. Training

After data preparation, which involved dividing the original images into smaller patches, the dataset was split into training and test sets. We randomly selected 500 images for the test set and applied undersampling to the remaining 10,174 images, as the original dataset was imbalanced [40–42]. In our case, only a small percentage of the pixels belonged to the least represented class—water—while the majority corresponds to the woodland class.

To address this imbalance, we applied undersampling to the training set by selecting images with a more balanced class distribution, as appropriate undersampling can improve model performance. Specifically, individual image–mask pairs were only included in the training set if at least four out of five classes contained a minimum of 1000 pixels.

Before training began, we applied several preprocessing steps, including data normalization and data augmentation. The augmentation techniques used were random flipping and image resizing with the following scale ratios: 0.5, 0.75, 1.0, 1.25, 1.5, and 1.75. Figure 2 presents samples of the input images and the corresponding masks, both with and without the application of undersampling.

The training process was set to 250,000 iterations for each network, with models saved every 50,000 iterations and performance metrics (accuracy and loss) logged every 2500 iterations. The training parameters were as follows: a learning rate of 0.01, a momentum value of 0.9, a weight decay value of 0.0005, SGD as the optimization algorithm, and a batch size of 4.

The hyperparameters were kept consistent across all models. The learning rate and momentum values were fine-tuned through preliminary experiments on the first two networks to achieve a balance between training stability and the convergence time.

Each network was trained for 250,000 iterations on an NVIDIA RTX 4090 GPU. The batch size of 4 was chosen to balance training speed and memory constraints, ensuring that the model fit entirely in the GPU memory and that training was completed within 24 h for each model. Once the optimal configuration was identified through initial trials, the same setup was applied to all networks to ensure consistency.

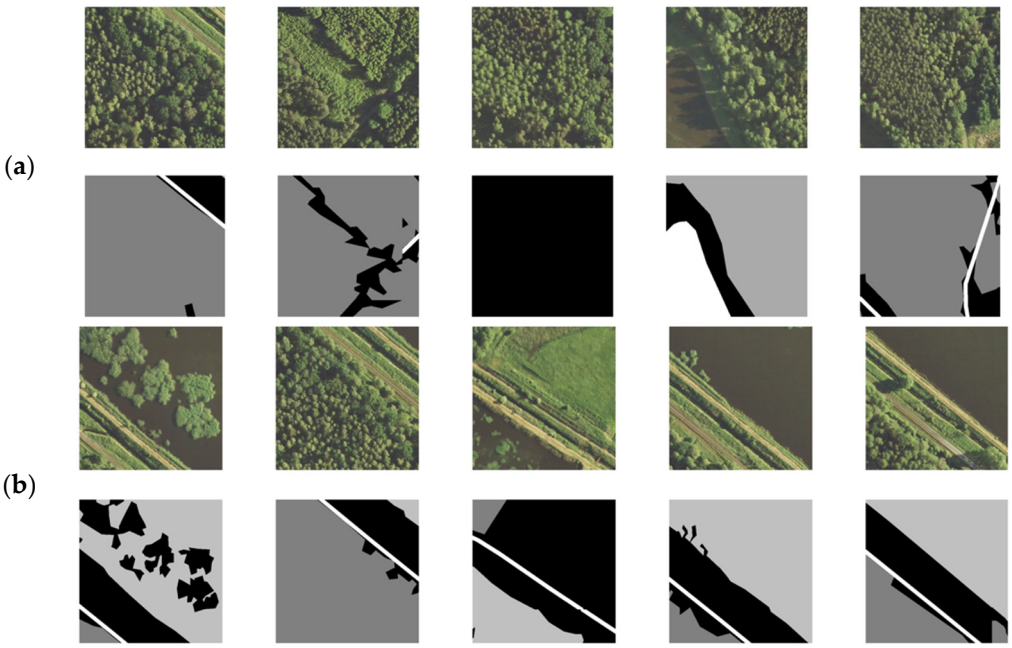

Figure 2. Sample images. Undersampling was applied due to class imbalance in the dataset. Individual images and masks were selected when at least four out of five classes contained a minimum of 1000 pixels. (**a**) Sample images with corresponding masks before undersampling. (**b**) Sample images with corresponding masks after undersampling.

As evaluation criteria, we adopted accuracy and loss for both the training and validation sets. Figure 3 illustrates the evolution of accuracy and loss during the training of PSPNet. For post-training evaluation, we used the mean accuracy, mean intersection over union (mIoU), and mean recall, computed both globally and per-class across all five land cover classes.

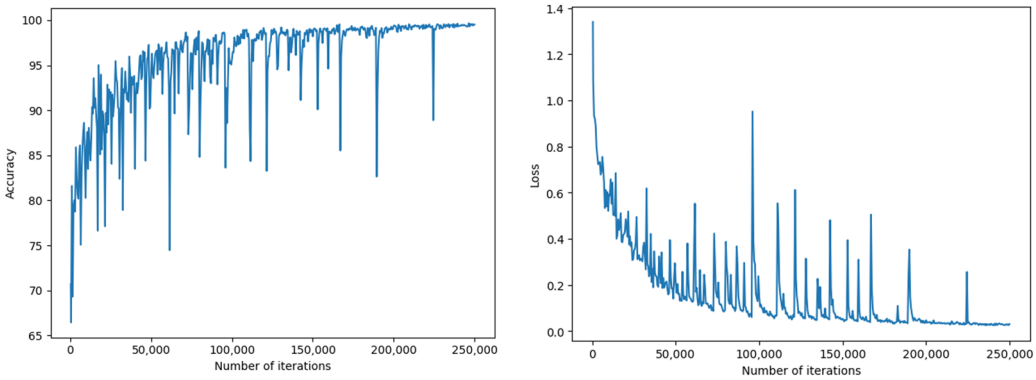

Figure 3. The trends of accuracy and loss during the training of PSPNet, the network identified as the best performer based on the outlier analysis. (**a**) The accuracy trend during PSPNet training. (**b**) The loss trend during PSPNet training.

Confusion matrices were computed for each image, and evaluation metrics were derived based on the counts of true positives (*TP*s), true negatives (*TN*s), false positives (*FP*s), and false negatives (*FN*s).

These evaluation metrics are defined below in Equations (1)–(4):

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$\text{F1-score} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{2}$$

$$\text{Intersection over union} = \frac{TP}{TP + FN + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

3.3. Analysis of Outliers

We performed a detailed examination of outliers in the context of statistical analysis following the computation of the segmentation task metrics. Outliers were identified for each class and subsequently analyzed with respect to intersection over union (IoU), as it was considered the most critical metric.

IoU is a geometric measure that quantifies the overlap between the ground-truth region and the predicted region, whereas accuracy is based on binary cross-entropy.

As a first step, we generated a "confusion map" for each class—an additional image in which the pixels are colored in three distinct ways according to their classification categories:

1. When the network prediction matches the ground truth, the corresponding pixels are colored green, as shown in Figure 4. These are referred to as true positives (see Figure 4a,b).
2. The red pixels in Figure 4b,c indicate false negatives, where the network failed to predict the relevant class, contrary to the ground truth (Figure 4).
3. The blue pixels in Figure 4d represent false positives, where the network predicted the presence of a class that is not present in the ground truth (Figure 4).

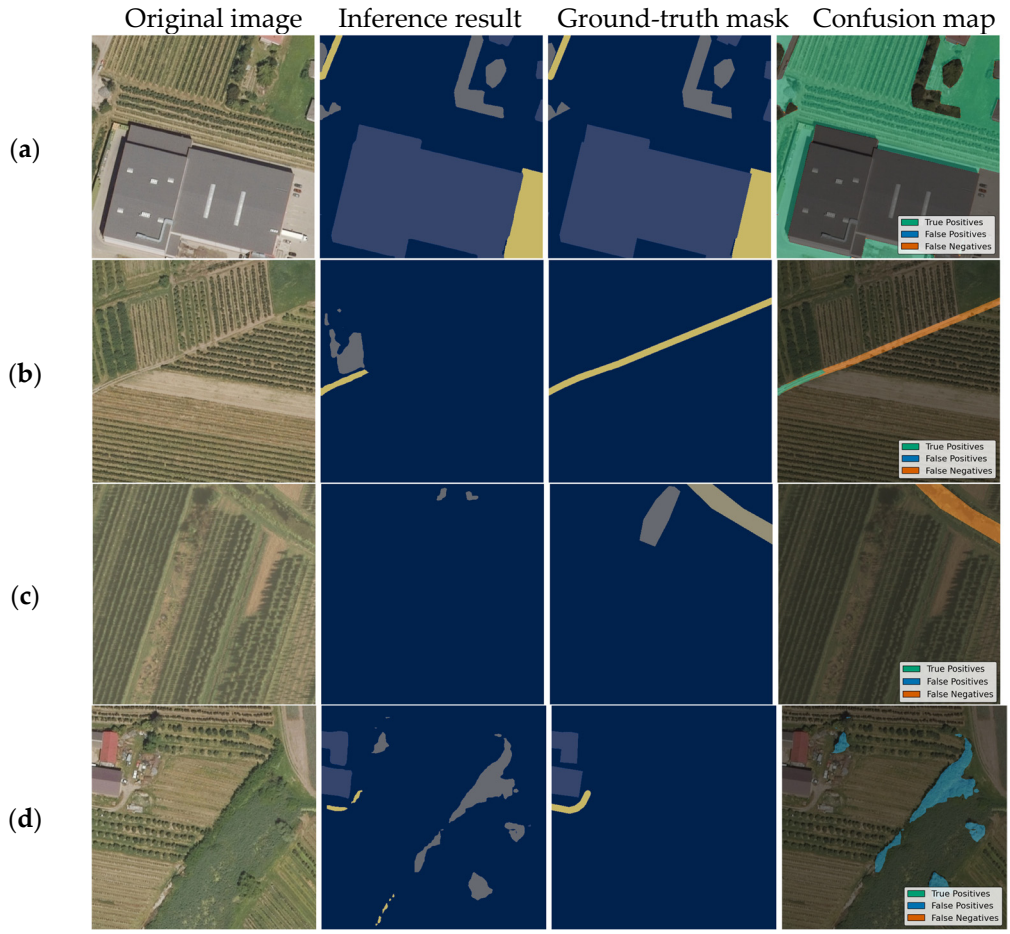

Figure 4. Outliers identified in three cases and represented using distinct colors: (**a**) Outliers in a case with true positives (the green areas in the confusion map). (**b**) Outliers in a case with true positives (green areas) and false negatives (red areas). (**c**) Outliers in a case with false negatives only (red areas). (**d**) Outliers in a case with false positives (blue areas).

While errors are sometimes present in the ground-truth annotations, the neural network can still produce correct predictions. For example, in Figure 4d, the network correctly identifies the road, whereas the ground truth mislabels it.

We therefore proceeded to classify the various types of misclassifications, followed by a detailed analysis of the outliers.

At this stage, we relied on three pixel categories to examine each of the 500 test set images containing outliers. As previously described, we generated five confusion maps for each original image in the dataset, where the pixels were colored according to their respective classes.

We then selected the confusion maps in which the overall intersection over union (IoU) of the associated classes exceeded 0.38%. This threshold was determined by analyzing the behavior of the violin plots, placing the cutoff at the extreme tail of the distribution (beyond the whiskers). This choice did not appear to limit the analysis, since, as will be shown in Section 4.2, the violin plot distributions exhibited a bimodal pattern, and both relevant groups remained preserved after filtering.

Accordingly, we defined the threshold confusion maps as those in which at least 1000 pixels were either red or blue—i.e., where the prediction deviated significantly from the ground truth. Notably, this threshold condition aligned with selecting confusion maps where the corresponding IoU exceeded 0.38%.

The next step was to classify the three types of misclassifications for each selected image based on the color-coded pixel categories described in Figure 5:

- Network mistakes refer to cases in which the neural network produces an incorrect prediction, while the ground truth is correct. For example, in Figure 5a a confusion map is generated with respect to the woodland class, but the neural network fails to recognize a portion of the image that should be classified as woodland, while the ground truth correctly labels it.
- Ground-truth mistakes: In this case, the ground truth contains an error, while the neural network correctly classifies the content of the image. Figure 5b shows an example where the neural network correctly identifies a tree, whereas the ground truth fails to do so.
- Ambiguous mistakes are cases that do not clearly fall into either of the two categories above. These occur when the original image contains visual ambiguity, making it difficult to assign responsibility for the error to either the network or the ground truth. For example, Figure 5c represents a confusion map generated with respect to the background class. The neural network classifies the shadowed area as background (in blue), whereas the ground truth does not. Whether shadows should be classified as background is itself an ambiguous matter. Similarly, red pixels indicate uncertainty between background and woodland where the boundary of tree coverage is unclear.

The use of violin plots to examine the lower tail of the IoU distribution across the dataset supported the choice of the 0.38% threshold. This cutoff captured only the samples with the lowest IoU values, corresponding to the region beyond the whiskers. Although heuristic, this approach provided a consistent and conservative definition of outliers. The rationale behind the choice of the 1000-pixel threshold (approximately 0.38% IoU) is discussed in detail in Appendix A.

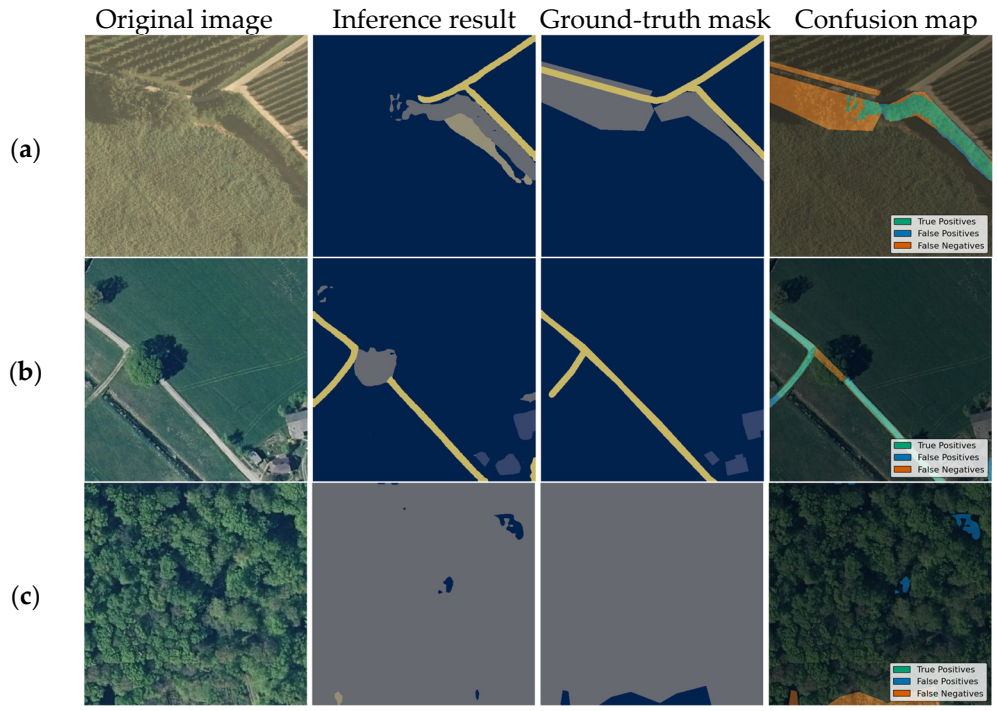

Figure 5. A sample of outliers representing three types of misclassifications: (**a**) A network mistake, shown as red pixels in the confusion map. (**b**) A ground-truth mistake, shown as red pixels in the confusion map. (**c**) Ambiguous mistakes, with red and blue pixels indicating areas of uncertainty in the confusion map.

4. Experiments

4.1. Experimental Setup

The performance of fifteen distinct neural networks was evaluated using semantic segmentation techniques. In the first phase, these models were assessed based on performance metrics such as recall, accuracy, and intersection over union (IoU), following a training period of approximately 11 h per model. For training, we did not use pretrained weights; instead, all models were initialized with random weights. This decision ensured uniform experimental conditions and isolated the effect of network architecture. Although pretrained weights are commonly employed in transfer learning, we deliberately excluded them due to the potential domain mismatch of the aerial and agricultural imagery in the LandCover.ai dataset. A comparative analysis of pretrained versus non-pretrained models is presented in Appendix B.

In the second phase of the experiment, we identified outliers by generating confusion maps for the 500 test set images according to the five semantic categories. This enabled a detailed analysis of the mistake frequencies attributed to different sources—namely, network mistakes, ground-truth mistakes, and ambiguous cases—for each of the fifteen networks across the entire test set.

All experiments were conducted on a desktop system with the following specifications: a 12th-Gen Intel Core i9-13900K × 32 processor, 128 GB of RAM, an NVIDIA RTX 4090 GPU, running Ubuntu 24.04.2 LTS, and using Python 3 as the development environment.

4.2. Results of Semantic Segmentation

Accuracy, F1-score, IoU, and recall were the parameters used to assess the model for the semantic segmentation problem. In this work, they were computed for all five classes, and their mean values were computed for each of the fifteen networks chosen, which are reported in the boxplots in Figure 6. The following mean values were observed for all networks, as shown in the boxplots (Figure 6): intersection over union was above 60%, the F1-score was above 64%, accuracy was above 98%, and recall was above 85%.

Figure 7 presents violin plots for the fifteen networks, illustrating the distributions of mean accuracy, mean intersection over union (IoU), and mean recall. The proposed models achieved a mean accuracy of 99.06%, an F1-score of 72.94%, a mean IoU of 71.5%, and a mean recall of 88.43%.

Table 1 reports the accuracy results for each class, along with the corresponding mean values. Table 2 presents the F1-score results for each class, as well as their mean values. Table 3 shows the intersection over union (IoU) results for all classes, together with the overall mean values. Table 4 provides the recall results for each class and their respective mean values.

In Table 5, the computational costs, the model sizes, and the ratios between accuracy and computational cost are presented for each network.

Although the class-wise accuracy values are generally above 0.99, these high figures can be partly attributed to the dominance of the “background” and “woodland” classes, which contribute substantially to the number of true negatives (TNs). In imbalanced datasets, accuracy can be misleading, as it may overstate performance for majority classes due to the inclusion of TNs in its calculation.

For underrepresented classes such as “water” and “roads”, recall provides a clearer indication of the model’s ability to detect those classes, as it explicitly penalizes false negatives (FNs). Similarly, intersection over union (IoU) offers a balanced view of segmentation performance by accounting for both false positives (FPs) and false negatives. For these reasons, our evaluation prioritized recall and IoU as more reliable performance indicators than absolute accuracy.

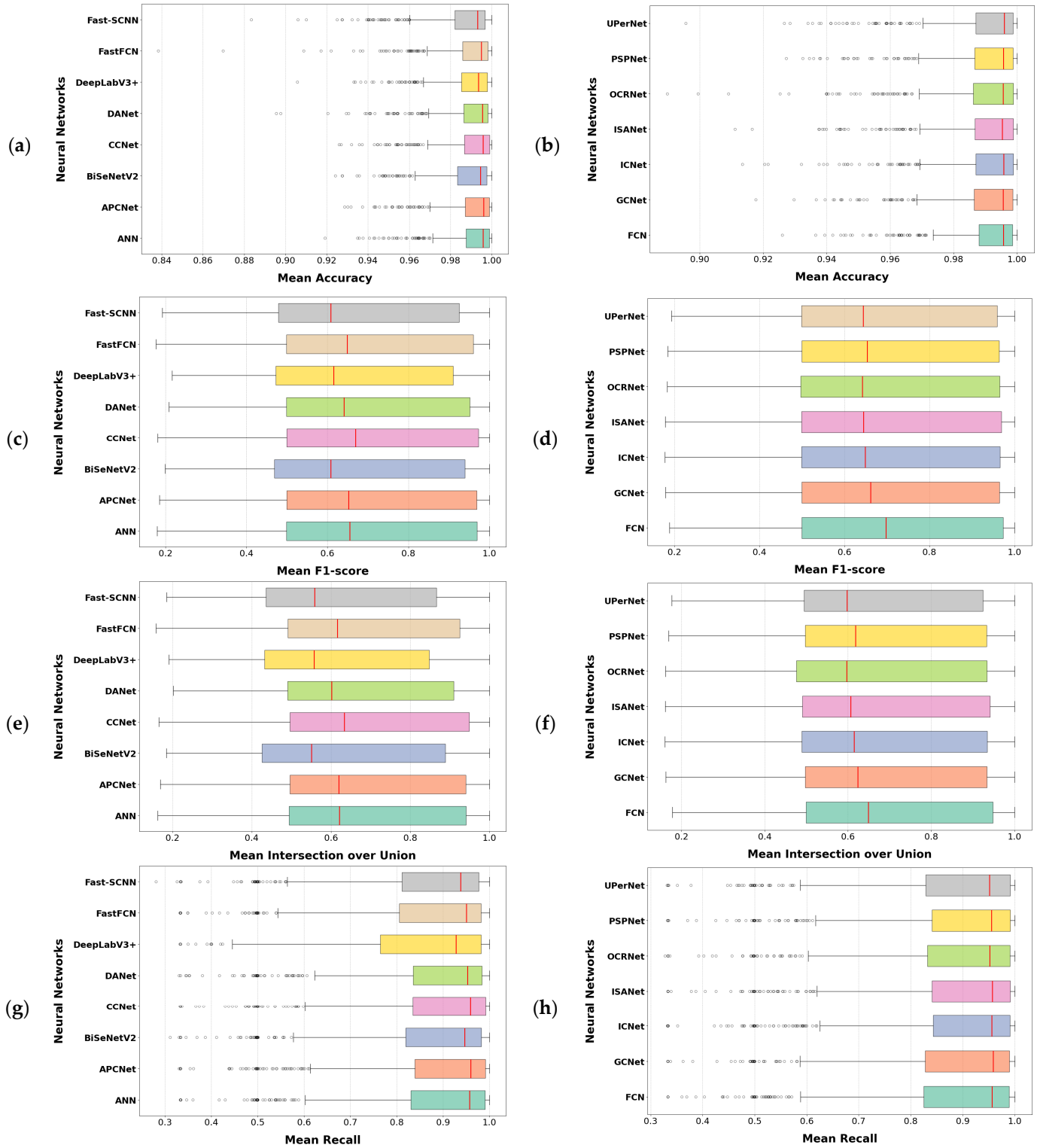

Figure 6. The accuracy, intersection over union, and recall distributions across fifteen networks: (**a**) The mean accuracy distributions across 8 networks. (**b**) The mean accuracy distributions across 7 networks. (**c**) The mean F1-score distributions across 8 networks. (**d**) The mean F1-score distributions across 7 networks. (**e**) The mean IoU distributions across 8 networks. (**f**) The mean IoU distributions across 7 networks. (**g**) The mean recall distributions across 8 networks. (**h**) The mean recall distributions across 7 networks. The boxplots show percentile ranges, with outliers represented as small circles beyond the whiskers.

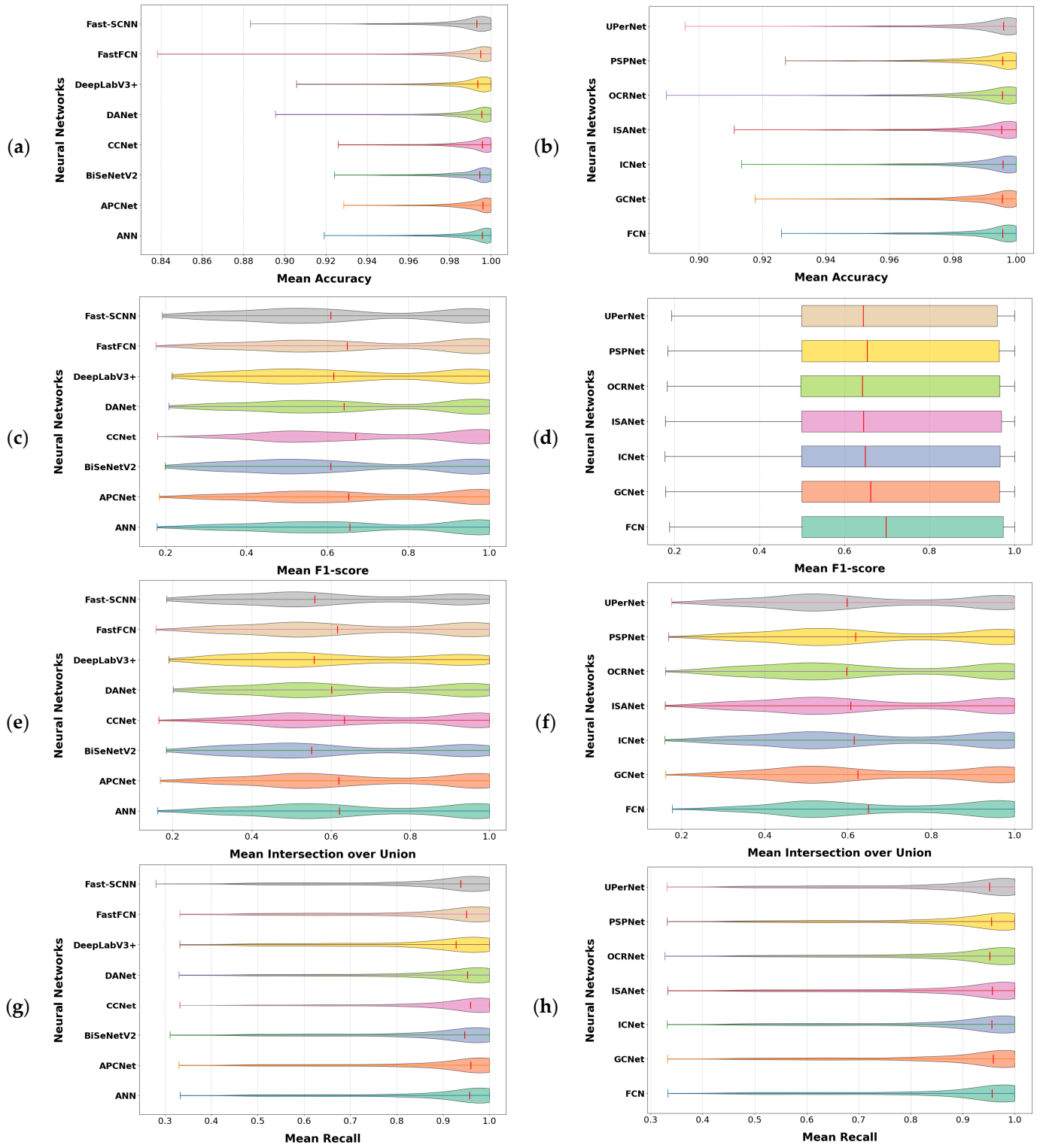

Figure 7. Accuracy, intersection over union, and recall distributions across fifteen networks: (**a**) Violin plot of mean accuracy distributions across 8 networks. (**b**) Violin plot of mean accuracy distributions across 7 networks. (**c**) Violin plot of mean F1-score distributions across 8 networks. (**d**) Violin plot of mean F1-score distributions across 7 networks. (**e**) Violin plot of mean IoU distributions across 8 networks. (**f**) Violin plot of mean IoU distributions across 7 networks. (**g**) Violin plot of mean recall distributions across 8 networks. (**h**) Violin plot of mean recall distributions across 7 networks.

Table 1. Accuracy across 15 networks.

Accuracy						
Neural Network	**Background**	**Buildings**	**Woodland**	**Water**	**Roads**	**Mean Accuracy**
ANN	0.9770	0.9990 (1)	0.9825	0.9974	0.9957	0.9903
APCNet	0.9775 (2)	0.9990 (1)	0.9830 (2)	0.9978 (1)	0.9956	0.9906 (2)
BiSeNetV2	0.9732	0.9984	0.9800	0.9969	0.9952	0.9887
CCNet	0.9770	0.9990 (1)	0.9825	0.9976 (2)	0.9956	0.9904
DANet	0.9752	0.9989 (2)	0.9821	0.9963	0.9956	0.9896
DeepLabV3	0.9742	0.9987	0.9811	0.9974	0.9949	0.9896
FastFCN	0.9739	0.9987	0.9814	0.9960	0.9953	0.9891
Fast-SCNN	0.9686	0.9981	0.9770	0.9958	0.9942	0.9867
FCN	0.9779 (1)	0.9990 (1)	0.9835 (1)	0.9977	0.9960 (1)	0.9908 (1)
GCNet	0.9764	0.9990 (1)	0.9824	0.9981	0.9951	0.9902
ICNet	0.9769	0.9989 (2)	0.9828	0.9973	0.9958	0.9904
ISANet	0.9759	0.9990 (1)	0.9817	0.9972	0.9957	0.9899
OCRNet	0.9745	0.9990 (1)	0.9815	0.9960	0.9955	0.9893
PSPNet	0.9765	0.9989 (2)	0.9820	0.9978 (1)	0.9956	0.9902
UPerNet	0.9758	0.9989 (2)	0.9821	0.9965	0.9959 (2)	0.9899

(1) The maximum value corresponding to each label and corresponding to the mean value. (2) The second value corresponding to each label and corresponding to the mean value.

Table 2. F1-score across 15 networks.

F1-Score						
Neural Network	**Background**	**Buildings**	**Woodland**	**Water**	**Roads**	**Mean Accuracy**
ANN	0.8630	0.7049	0.6692	0.3528	0.5738	0.6992
APCNet	0.8673 (1)	0.7105 (2)	0.6612	0.3895	0.5619	0.7012
BiSeNetV2	0.8530	0.5731	0.6191	0.3112	0.5057	0.6500
CCNet	0.8637	0.6992	0.6839 (2)	0.4303	0.5681	0.7129 (2)
DANet	0.8627	0.6933	0.6664	0.3558	0.5187	0.6819
DeepLabV3	0.8596	0.6086	0.6246	0.3319	0.4560	0.6485
FastFCN	0.8580	0.6589	0.6594	0.4000	0.5348	0.6899
Fast-SCNN	0.8508	0.5657	0.6238	0.3012	0.4875	0.6439
FCN	0.8637	0.7475 (1)	0.6925 (1)	0.4661 (2)	0.5972 (1)	0.7294 (1)
GCNet	0.8614	0.6974	0.6595	0.5145 (1)	0.5564	0.7110
ICNet	0.8669 (2)	0.6726	0.6623	0.3591	0.5670	0.6964
ISANet	0.8617	0.7013	0.6594	0.3525	0.5610	0.6922
OCRNet	0.8614	0.6578	0.6682	0.3462	0.5486	0.6856
PSPNet	0.8635	0.6493	0.6564	0.4477	0.5618	0.6986
UPerNet	0.8574	0.6954	0.6619	0.3580	0.5780 (2)	0.6914

(1) The maximum value corresponding to each label and corresponding to the mean value. (2) The second value corresponding to each label and corresponding to the mean value.

Table 3. Intersection over union across 15 networks.

Intersection Over Union						
Neural Network	**Background**	**Buildings**	**Woodland**	**Water**	**Roads**	**Mean IoU**
ANN	0.8391 (1)	0.6702	0.6341	0.3382	0.5254	0.6689
APCNet	0.8438	0.6746	0.6276	0.3741	0.5139	0.6711
BiSeNetV2	0.8267	0.5311	0.5804	0.2932	0.4566	0.6172
CCNet	0.8389 (2)	0.6636	0.6474 (2)	0.4118	0.5196	0.7150 (1)
DANet	0.8374	0.6502	0.6289	0.3385	0.4699	0.6496
DeepLabV3	0.8338	0.5642	0.5805	0.3136	0.4017	0.6132
FastFCN	0.8325	0.6194	0.6210	0.3843	0.4822	0.6567
FCN	0.8402	0.7086 (2)	0.6526 (1)	0.4467 (2)	0.5456 (1)	0.6967 (2)
Fast-SCNN	0.8224	0.5165	0.5815	0.2861	0.4338	0.6082
GCNet	0.8367	0.6594	0.6232	0.4931 (1)	0.5060	0.6782
ICNet	0.8417	0.7085 (1)	0.6305	0.3429	0.4967	0.6654
ISANet	0.8370	0.6660	0.6241	0.3380	0.5145	0.6550
OCRNet	0.8355	0.6259	0.6312	0.3307	0.5034	0.6551
PSPNet	0.8391 (1)	0.6168	0.6214	0.4278	0.5116	0.6673
UPerNet	0.8321	0.6604	0.6265	0.3428	0.5299 (2)	0.6608

(1) The maximum value corresponding to each label and corresponding to the mean value. (2) The second value corresponding to each label and corresponding to the mean value.

Table 4. Recall across 15 networks.

Recall						
Neural Network	**Background**	**Buildings**	**Woodland**	**Water**	**Roads**	**Mean IoU**
ANN	0.9375	0.8414	0.8482	0.8278 (1)	0.7802	0.8815
APCNet	0.9389	0.8405	0.8506	0.8105	0.7798	0.8823
BiSeNetV2	0.9333	0.8148	0.8400	0.8025	0.7472	0.8704
CCNet	0.9338	0.8483 (2)	0.8575	0.8141	0.7807	0.8828
DANet	0.9403	0.8276	0.8502	0.7998	0.7594	0.8776
DeepLabV3	0.9443 (2)	0.7832	0.8001	0.8187	0.6769	0.8540
FastFCN	0.9356	0.8245	0.8297	0.7573	0.7447	0.8656
Fast-SCNN	0.9264	0.7986	0.8334	0.7759	0.7359	0.8628
FCN	0.9501 (1)	0.8202	0.8336	0.7902	0.7615	0.8772
GCNet	0.9382	0.8398	0.8523	0.7850	0.7771	0.8807
ICNet	0.9417	0.8452	0.8544	0.8100	0.7788	0.8843 (1)
ISANet	0.9356	0.8378	0.8570	0.8198 (2)	0.7872 (1)	0.8836
OCRNet	0.9340	0.8403	0.8628 (1)	0.8195	0.7847 (2)	0.8834
PSPNet	0.9364	0.8498 (1)	0.8609 (2)	0.8148	0.7808	0.8842 (2)
UPerNet	0.9371	0.8475	0.8517	0.8129	0.7684	0.8806

(1) The maximum value corresponding to each label and corresponding to the mean value. (2) The second value corresponding to each label and corresponding to the mean value.

Table 5. Computational costs and sizes of neural networks.

Computational Costs, Sizes, and Ratios Between Accuracy and Computational Costs				
Neural Network	**Computational Cost**	**Size**	**Acc/CC (*)**	**Acc/Size (**)**
ANN	185 GFLOPS	46.217 Mparams	5.353×10^{-3}	2.142×10^{-2}
APCNet	204 GFLOPS	56.346 Mparams	4.856×10^{-3}	1.758×10^{-2}
BiSeNetV2	12 GFLOPS	14.789 Mparams	8.239×10^{-2}	6.685×10^{-2}
CCNet	201 GFLOPS	49.815 Mparams	4.927×10^{-3}	1.988×10^{-2}
DANet	211 GFLOPS	49.821 Mparams	4.69×10^{-3}	1.986×10^{-2}
DeepLabV3+	176 GFLOPS	43.579 Mparams	5.623×10^{-3}	2.271×10^{-2}
FastFCN	130 GFLOPS	68.7 Mparams	7.609×10^{-3}	1.44×10^{-2}
Fast-SCNN	0.5 GFLOPS	1.454 Mparams	1.973	6.786×10^{-1}
FCN	197 GFLOPS	49.486 Mparams	5.029×10^{-3}	2.002×10^{-2}
GCNet	197 GFLOPS	49.619 Mparams	5.026×10^{-3}	1.996×10^{-2}
ICNet	15 GFLOPS	47.824 Mparams	6.603×10^{-2}	2.071×10^{-2}
ISANet	149 GFLOPS	37.696 Mparams	6.644×10^{-3}	2.626×10^{-2}
OCRNet	267 GFLOPS	68.192 Mparams	3.705×10^{-3}	1.451×10^{-2}
PSPNet	178 GFLOPS	48.964 Mparams	5.563×10^{-3}	2.022×10^{-2}
UPerNet	236 GFLOPS	48.964 Mparams	4.195×10^{-3}	2.022×10^{-2}

(*) Accuracy/computational cost; (**) accuracy/size.

Furthermore, it is important to note that the neural networks tended to perform better in the presence of buildings and woodland, while performance significantly decreased for classes such as water and roads.

In terms of performance across the different neural networks, FCN achieved the highest accuracy, followed by APCNet. The best-performing network in terms of intersection over union (IoU) was CCNet, followed by FCN. With respect to recall, ICNet performed best, followed by PSPNet.

Using the inference functionality of the MMSegmentation library, we performed model predictions on test images. The process involved loading a trained model, supplying an input image, and executing an inference. Figure 8 illustrates sample inference results obtained using three different neural networks on images from the LandCover.ai dataset, displaying both the original images and the corresponding predicted segmentation masks. In the examined case, the inference output exhibits near-complete spatial correspondence with the ground truth annotation.

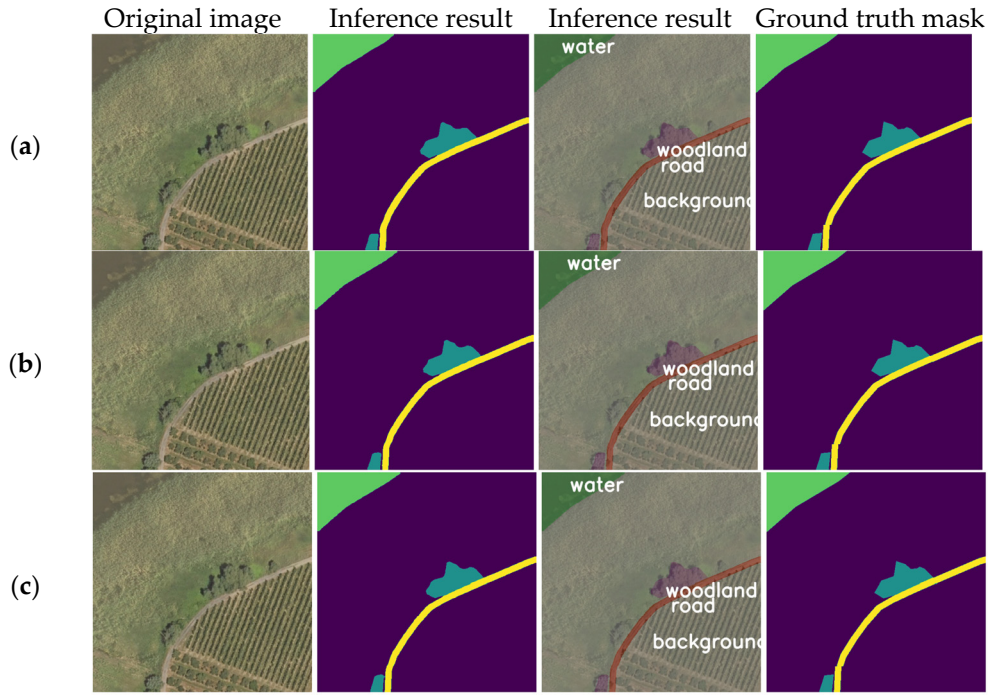

Figure 8. Inferences of three networks based on the LandCover.ai dataset: (**a**) An inference using FCN. (**b**) An inference using PSPNet. (**c**) An inference using ICNet.

4.3. The Results of the Analysis of the Outliers

In Table 6, the results of the outlier analysis conducted in this study are reported. We examined the outliers with the objective of classifying the types of misclassifications. The errors were manually assigned to three categories: network mistakes, ground-truth mistakes, and ambiguous cases. The neural networks that performed best were identified as those with the fewest mistakes attributable to the networks themselves. The top-performing model was PSPNet, with 21 network-related mistakes, followed by FCN with 26 errors and ICNetNet with 27 errors. When considering the total number of mistakes, including ground-truth mistakes and ambiguous cases, the best-performing networks were FCN, PSPNet, and ANN.

It can be stated that, with regard to the per-class results, there was no single network that consistently outperformed the others. For example, PSPNet achieved the best performance in terms of the total number of mistakes related to the 'background' class, but it was not the best-performing network for the 'woodland' and 'roads' classes. Given the very close mistake margins, the limited size of the test set, and the overall low number of errors, the dataset did not provide a sufficient sample size to conduct a statistically meaningful ANOVA test.

For the 'background' class, the best-performing networks were DANet, PSPNet, and UPerNet. Regarding the 'buildings' class, the top-performing models were CCNet, DeepLabV3+, FCN, ICNet, ISANet, and PSPNet. For the 'woodland' class, the networks showing the best performance were PSPNet, DANet, FCN, ICNet, and UPerNet. In the case of the 'water' class, the most effective models were DeepLabV3+, FCN, and PSPNet, while for the 'roads' class, the best results were achieved by FCN, ICNet, and UPerNet.

For this analysis, we selected intersection over union (IoU) as the primary evaluation metric, as it quantifies the degree of overlap between the predicted segmentation and the ground-truth region.

Table 6. The analysis of the outliers. The best-performing networks were FCN, PSPNet, and ICNet.

	ANN	APCNet	BiSeNetV2	CCnet	DANet	DeepLabV3+	FastFCN	Fast_SCNN	FCN	GCNet	ICNet	ISANet	OCRNet	PSPNet	UPerNet
Network mistakes															
background	9	8	11	7	3	10	11	11	11	15	9	12	15	5	6
buildings	1	3	1	0	1	0	2	2	0	2	0	0	4	0	1
woodland	19	16	12	8	7	11	14	14	7	17	7	9	13	4	7
water	9	26	9	3	8	0	2	2	1	13	4	3	33	0	7
road	8	23	16	13	10	20	19	19	7	25	7	9	22	12	7
Total network mistakes	46	76	49	31	29	41	48	48	26	72	27	33	87	21	28
Ground-truth mistakes															
background	17	7	3	13	7	11	14	14	9	15	9	14	17	16	11
buildings	0	4	0	0	1	0	2	2	0	3	0	0	3	0	1
woodland	10	53	28	22	17	23	27	27	17	55	16	20	60	15	16
water	0	2	0	0	0	0	0	0	0	1	0	0	0	0	0
road	14	22	16	11	13	10	13	13	10	27	13	12	23	11	8
Total ground-truth mistakes	41	88	47	46	38	44	56	56	36	101	38	46	103	42	36
Ambiguous mistakes															
background	29	35	44	30	47	41	53	53	33	24	33	35	32	16	39
buildings	0	5	2	0	0	2	4	4	1	0	0	0	1	0	0
woodland	43	4	65	45	59	58	68	68	36	6	52	43	3	66	66
water	2	1	7	4	7	6	3	3	3	3	6	5	2	0	4
road	13	15	19	19	20	20	32	32	10	17	21	18	15	20	9
Total ambiguous mistakes	87	60	137	98	133	127	160	160	83	50	112	101	53	102	118
Total															
background	55	50	58	50	57	62	78	78	53	54	51	61	64	37	56
buildings	1	12	3	0	2	2	8	8	1	5	0	0	8	0	2
woodland	72	73	105	75	83	92	109	109	60	78	75	72	76	85	89
water	11	29	16	7	15	6	5	5	4	17	10	8	35	0	11
road	35	60	51	43	43	50	64	64	27	69	41	39	60	43	24
Total	174	224	233	175	200	212	264	264	145	223	177	180	243	165	182

The divergence between a neural network's predictions and the ground truth was not always attributable to the network itself; in some cases, it was due to human annotation errors, and in others it was due to inherently ambiguous regions. This suggests that the actual predictive capacity of the neural networks may be higher than what is reflected in the raw metrics. In our study, we analyzed all such cases, identifying the best-performing networks based on accuracy, F1-score, intersection over union, and recall. These metrics were computed as functions of true positives, false positives, true negatives, and false negatives.

4.4. Discussion

The CNN architecture of our models is sufficiently flexible to adapt to the diverse data elements and attributes characteristic of agricultural imagery, as semantic segmentation methods and their applications in agriculture have rapidly evolved in recent years. We deliberately chose to focus on convolutional architectures, primarily for reasons of comparability, training stability, and computational feasibility. All networks were selected from the MMSegmentation framework in its 2021 configuration, consistently prioritizing models that offered a unified training interface and comprehensive documentation.

We excluded transformer-based models due to their recent introduction and higher computational demands, which would have significantly affected the feasibility of uniformly training 15 architectures on a single-GPU setup. However, we will consider their application in future work, particularly for exploring integration with agricultural datasets such as Sentinel-2 or multi-temporal drone imagery.

As previously noted, classification mistakes are not always attributable to the network; in many cases, they result from inaccuracies in ground-truth annotations. Additionally, there are instances where the human eye may also struggle due to ambiguous or visually confusing images.

In many related studies [12–14], only performance metrics are reported, without any detailed examination of outliers. Similarly, in the work of Adrian Boguszewski [7], only a single neural network, DeepLabV3+, is evaluated, without any comparative analysis.

In the present study, we trained fifteen neural networks using the MMSegmentation deep learning toolbox to develop semantic segmentation models. Four evaluation metrics—accuracy, F1-score, intersection over union (IoU), and recall—were computed for each of the five semantic classes, as well as their mean values, following an undersampling procedure to address class imbalance. We assessed outliers based on IoU and explicitly considered network-induced mistakes in the performance evaluation. Our findings indicate that PSPNet, FCN, and ICNet were the top-performing models.

As shown in Table 4, a network with high accuracy did not necessarily exhibit a high accuracy-to-computational-cost ratio. For instance, ICNet, Fast-SCNN, and BiSeNetV2 demonstrated favorable ratios despite not achieving the highest overall accuracy.

Thanks to its accuracy-to-GFLOP (Acc/GFLOP) ratio—which was higher than that of any other model evaluated—Fast-SCNN exhibited exceptional computational efficiency. Despite its lower mean accuracy and IoU, it represents a compelling trade-off between performance and resource consumption, making it particularly suitable for real-time edge device deployment in precision agriculture. Moreover, its extremely compact architecture (1.45 million parameters) makes it ideal for memory-constrained environments.

5. Conclusions

This study investigated the increasing application of deep convolutional neural networks (CNNs) in agricultural image segmentation, following an exploration of the LandCover.ai dataset and semantic segmentation techniques. The goal was to identify the model that performed best when analyzing precision agriculture and its specific characteristics.

We considered that precision agriculture presents distinct image patterns that differ significantly from those found in the domains where most segmentation networks were originally developed (e.g., medical imaging and autonomous driving). Consequently, the effectiveness of transfer learning across these domains may vary. In response, we conducted a study of these agricultural patterns using semantic segmentation algorithms by selecting fifteen neural networks and analyzing the outliers. The aim was to identify the models that performed best, not only based on standard metrics such as accuracy, F1-score, intersection over union (IoU), and recall but also considering the types of misclassifications observed.

Based on the results obtained in real-world scenarios, the most effective approach appears to be the combined use of multiple neural network models. The findings of this work provide a baseline that may stimulate further research and development in precision agriculture, as well as in other relevant domains such as urban planning and land monitoring.

Future challenges include extending this study to a greater number of semantic classes, a broader geographical scale (e.g., large-scale satellite imagery), or real-time drone-acquired images. Moving forward, improving segmentation performance across a wider range of agricultural contexts will require leveraging the additional spectral information provided by Sentinel-2 (or Copernicus) imagery. This will enable models to account for crop rotation, seasonal variation, and other dynamic characteristics of agricultural landscapes.

Author Contributions: M.F.: methodology, software, validation, formal analysis, investigation, data curation, writing—original draft preparation, writing—review and editing, visualization, and funding acquisition. C.A.A.: conceptualization, methodology, software, investigation, resources, data curation, review and editing, supervision, project administration, and funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This study was funded by the National Recovery and Resilience Plan (NRRP) (Mission 4, Component 2, Investment 1.4—Strengthening research infrastructures and creating "National Champions in R&D" in selected Key Enabling Technologies) and the National Research Centre for Agricultural Technologies (AGRITECH) (CN_00000022, funded by the European Union through NextGenerationEU).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset on which this research was carried out is publicly available at https://landcover.ai.linuxpolska.com/ (accessed on 24 June 2025).

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of this study; in the collection, analyses, or interpretation of the data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
Acc	Accuracy
IoU	Intersection over union

Appendix A

During the design of the experiments and the analysis of the results, pixel thresholds were applied prior to determining whether the classification performed by the network was acceptable. More specifically, since the segmentation analysis was conducted at the pixel level, we introduced these thresholds to exclude cases that may be of limited relevance

to the overall geometry of an area (e.g., isolated or sparse pixel misclassifications within an image region) or to the evaluation of boundary properties (e.g., minor discrepancies in border delineation between the network outputs and expert annotations).

This appendix elaborates on the rationale behind the selection of appropriate pixel thresholds for performance evaluation in semantic segmentation. Although a general threshold of 1000 pixels—corresponding to approximately 0.38% of a standard 512×512 image—was found to be optimal in most scenarios, class-specific analysis revealed slight adjustments that enhance robustness. The objective was to determine suitable cutoff values that effectively filter out marginal predictions while preserving meaningful, class-specific detections.

To this end, we performed a per-class analysis on the dataset and generated representative histograms showing the distributions of true positives (TPs), false positives (FPs), and false negatives (FNs) across the dataset images. These histograms provided insight into the prevalence of small-area misclassifications for each class and supported the derivation of appropriate thresholds tailored to different segmentation scenarios.

A multi-step evaluation strategy was adopted. Initially, TP, FP, and FN pixel distributions were calculated for each land cover class across the training images. Threshold candidates were then assessed based on their ability to exclude minimally relevant FP and FN errors. These values were validated both visually and statistically to identify the class-specific breakpoints that yielded the most favorable trade-offs between precision and recall.

The analysis was conducted using segmentation models developed with the MMSegmentation framework, which were applied to a land cover dataset annotated for five semantic classes. Custom Python scripts were used to compute per-image pixel distributions and generate evaluation metrics. Histograms were produced to visualize TP, FP, FN, and total ground-truth pixel counts per image. Additionally, confusion maps were generated to spatially analyze misclassifications, with particular attention given to boundary regions, where errors were most frequently observed.

The visual tools used included Matplotlib (version 3.8.2) for generating histogram plots and OpenCV for creating confusion maps. Each map overlaid misclassification patterns using color-coded pixels to separately highlight true positives (TPs), false positives (FPs), and false negatives (FNs). This dual-visualization approach offered both statistical insight and spatial interpretability.

In parallel with the analysis, we also extracted a representative image from the dataset for each segmented class, where the number of misclassified pixels slightly exceeded the selected threshold. For each case, we present a confusion map to support a detailed misclassification analysis, illustrating the actual discrepancies between the ground truth (GT) and the predicted mask.

Figures A1–A10 present the complete spectrum of the analysis. Each figure focuses on a specific class or threshold configuration. For the background class, Figure A1 demonstrates that most true positive (TP) pixels were preserved and that a threshold of approximately 1500 pixels effectively minimized errors while avoiding under-segmentation. The histogram in Figure A3 supports this finding for the class buildings, although a threshold of around 1000 pixels was sufficient in less complex cases. Figures A2 and A4 display the corresponding confusion maps, confirming that false positives (FPs) and false negatives (FNs) were primarily concentrated along class boundaries or restricted to small, isolated regions.

The background class was the most “noisy” because it encompassed all regions not attributed to the other classes. It included large cultivated areas, uncultivated fields, and occasionally vegetation in urban areas not classified as woodland. This ambiguity in the class content resulted in a wide variety of error types and scales. However, by applying the

previously identified threshold, nearly all misclassifications involving small-area segments were eliminated.

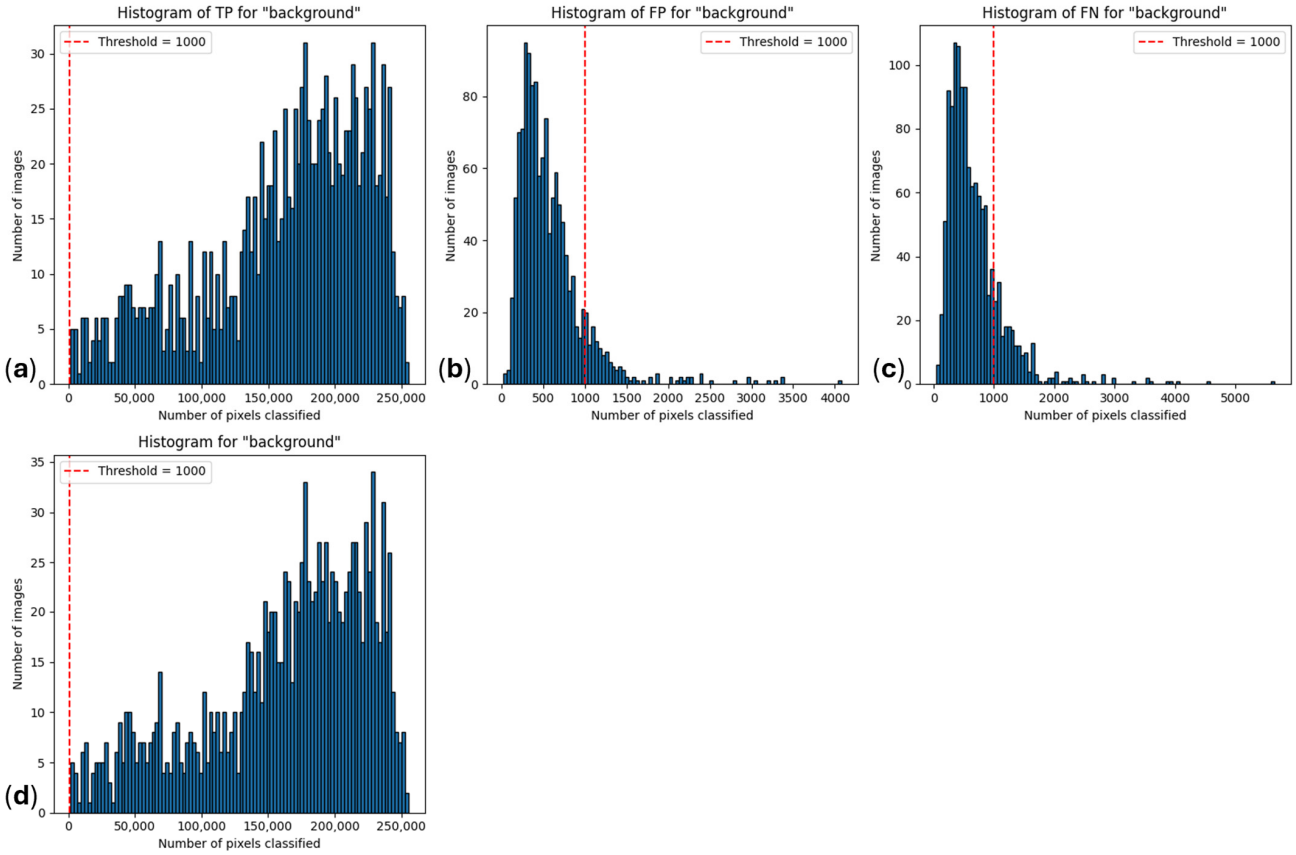

Figure A1. (**a**) A histogram of the number of true positives per image for the class 'background'. (**b**) A histogram of the number of false positives per image for the class 'background'. (**c**) A histogram of the number of false negatives per image for the class 'background'. (**d**) A histogram of the number of ground-truth pixels classified as 'background'.

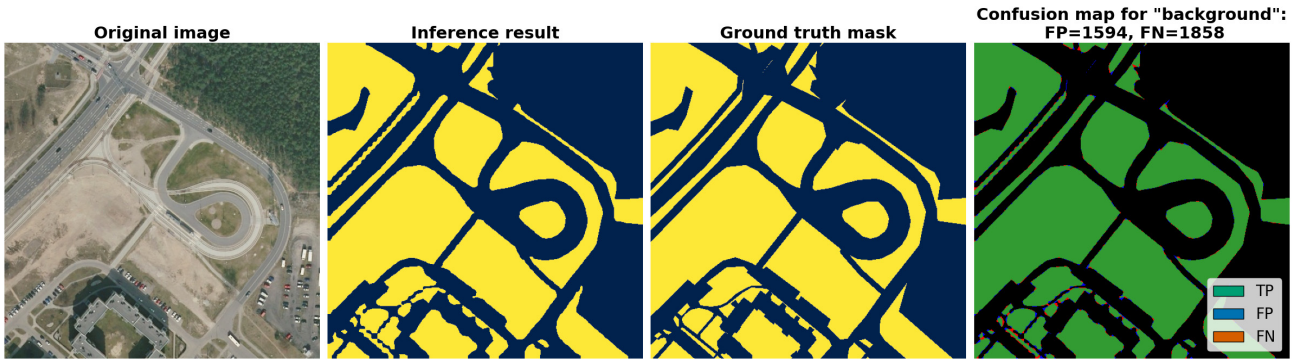

Figure A2. A confusion map for the class "background". The false positives and false negatives are predominantly attributable to the borders.

In the case of woodland segmentation (Figure A5), the findings differed slightly. A threshold of approximately 800 pixels already yielded the optimal segmentation performance by effectively excluding erroneous boundary detections. Nevertheless, the number of images containing fewer than 1000 pixels was relatively limited compared to those for buildings and was approximately equal (as shown in the histogram) to the number of images below the 800-pixel threshold. Figure A6 illustrates the spatial distribution

of the segmentation discrepancies for this class and presents insights consistent with Figures A2 and A4.

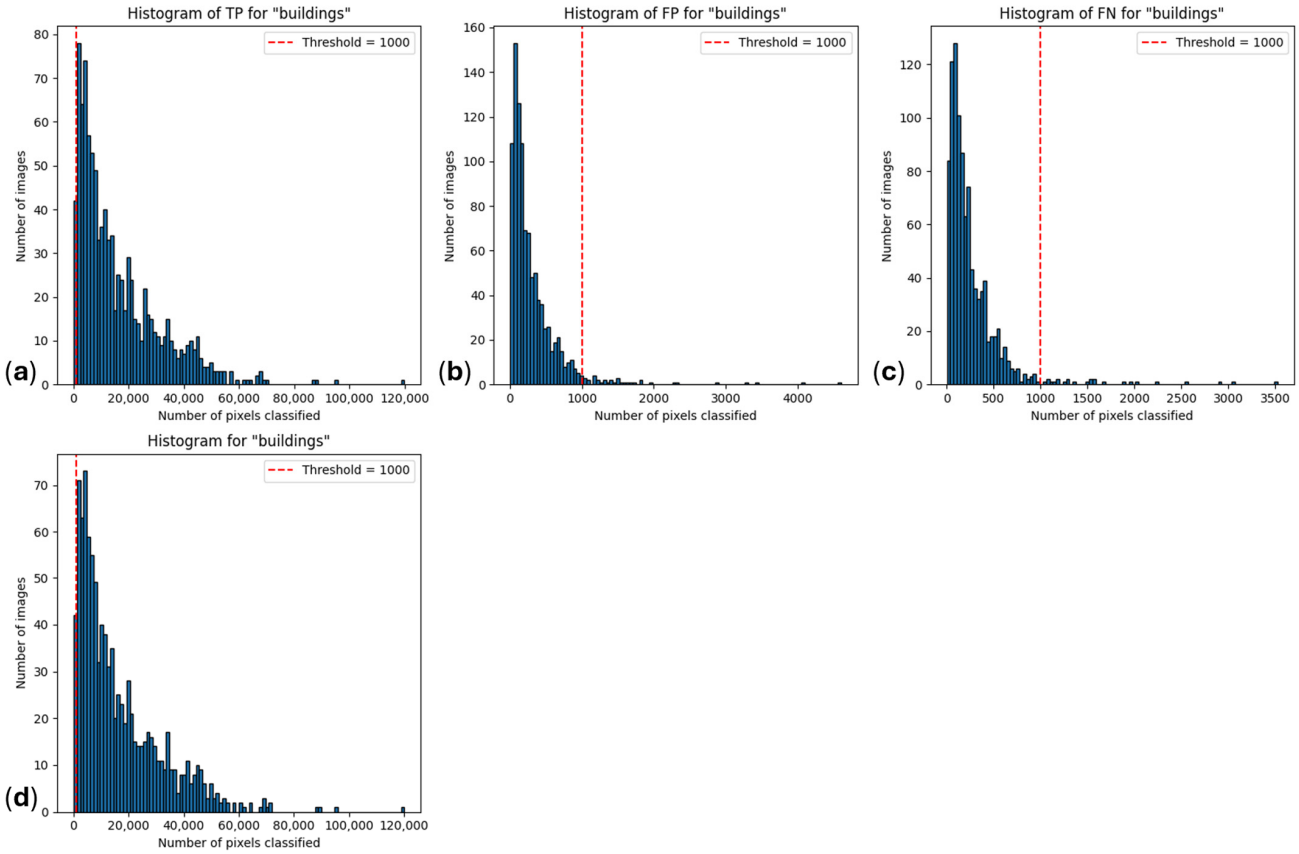

Figure A3. (**a**) A histogram of the number of true positives per image for the class 'buildings'. (**b**) A histogram of the number of false positives per image for the class 'buildings'. (**c**) A histogram of the number of false negatives per image for the class 'buildings'. (**d**) A histogram of the number of ground-truth pixels classified as 'buildings'.

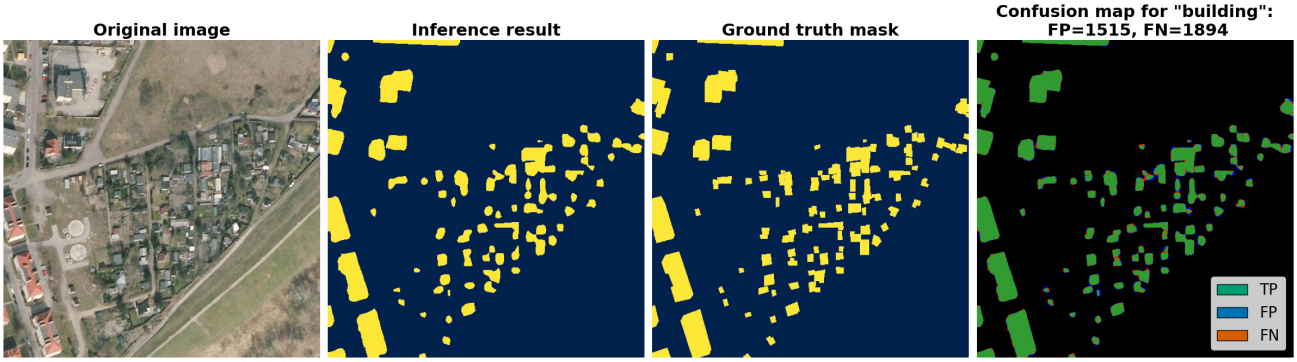

Figure A4. A confusion map for the class "buildings". The false positives and false negatives are predominantly attributable to the borders.

The water segmentation results are presented in Figures A7 and A8. Owing to the higher geometric regularity of water bodies relative to the other classes, the number of small-sized errors was lower. The histogram analysis revealed fewer areas with small spatial footprints and more regular boundary delineations. These combined factors resulted in a significantly lower optimal threshold of around 300 pixels. In Figure A8, the class boundaries also exhibit reduced ambiguity.

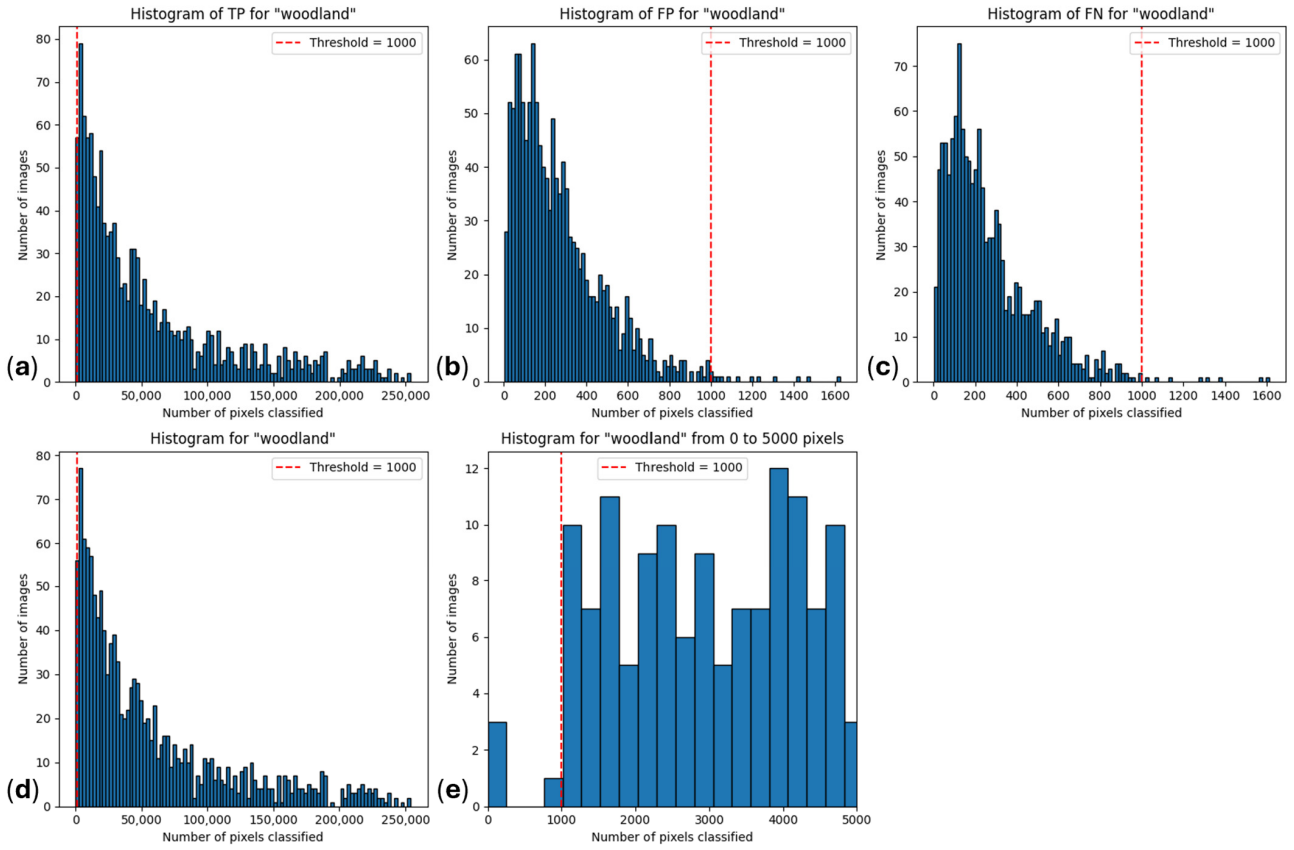

Figure A5. (**a**) A histogram of the number of true positives per image for the class 'woodland'. (**b**) A histogram of the number of false positives per image for the class 'woodland'. (**c**) A histogram of the number of false negatives per image for the class 'woodland'. (**d**) A histogram of the number of ground-truth pixels classified as 'woodland'. (**e**) A histogram of the number of ground-truth pixels classified as 'woodland' limited to the range 0–5000 pixels.

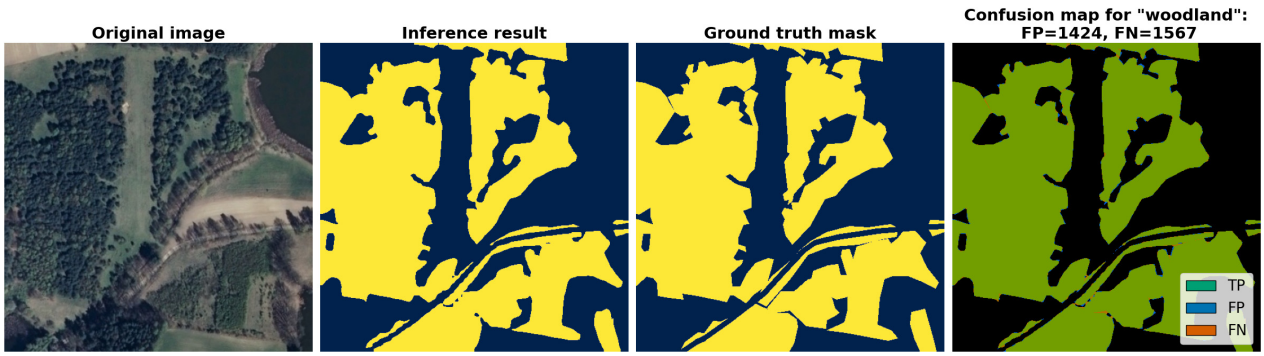

Figure A6. A confusion map for the class "woodland". The false positives and false negatives are predominantly attributable to the borders.

The road class, depicted in Figures A9 and A10, showed an optimal threshold of approximately 600 pixels. The true-positive coverage remained stable, while false positives and false negatives were substantially reduced, particularly near image boundaries. Across all five classes, the class-specific thresholds improved segmentation clarity and reduced evaluation noise.

The confusion maps and histograms consistently indicate that most segmentation errors occurred along class boundaries. These transitional zones often displayed mixed features, leading to prediction uncertainty. For background, the 1500-pixel threshold shown in Figure A1 reflected the need for a higher cutoff due to its dominant spatial extent.

However, under less complex conditions, as demonstrated in Figure A3, a 1000-pixel threshold was sufficient. For buildings, an 800-pixel threshold yielded optimal results.

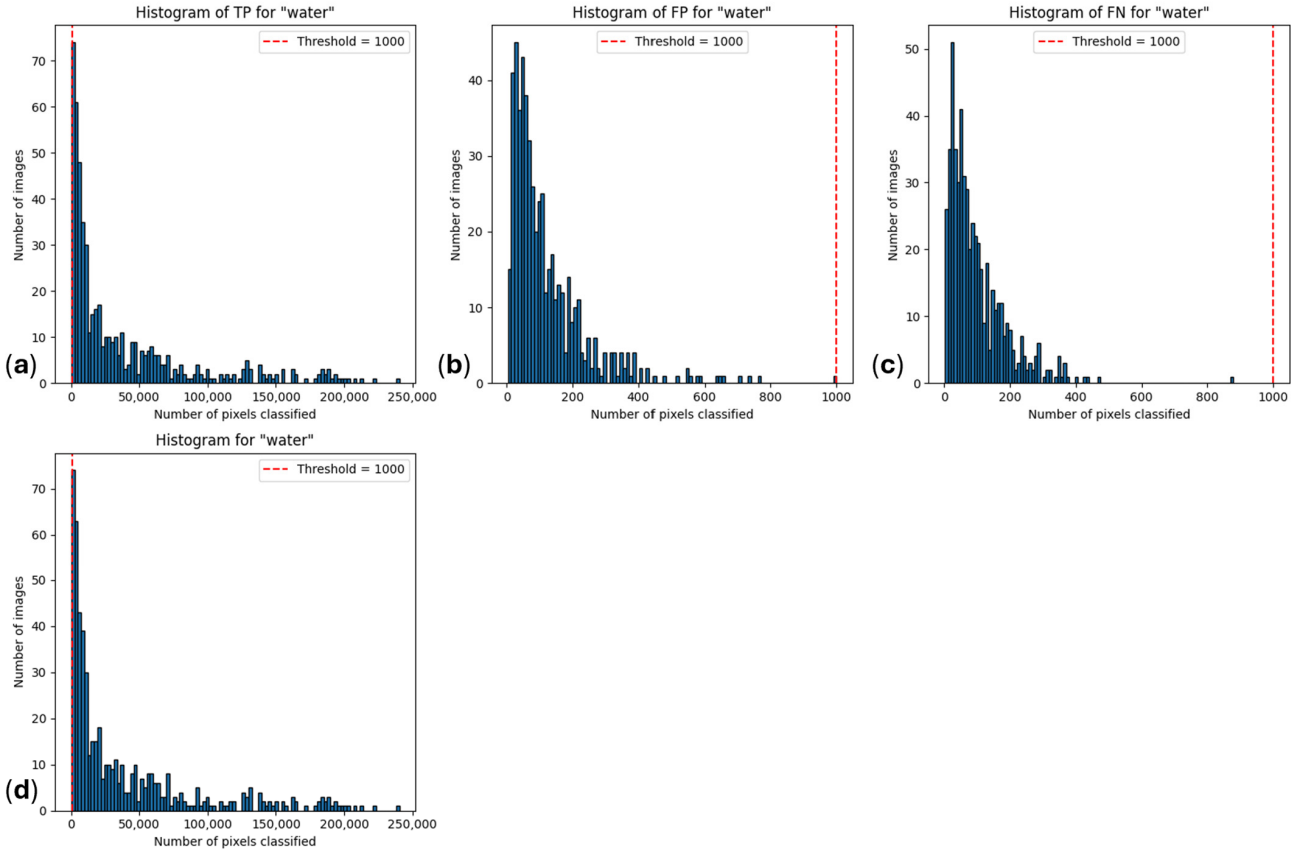

Figure A7. (**a**) A histogram of the number of true positives per image for the class 'water'. (**b**) A histogram of the number of false positives per image for the class 'water'. (**c**) A histogram of the number of false negatives per image for the class 'water'. (**d**) A histogram of the number of ground-truth pixels classified as 'water'.

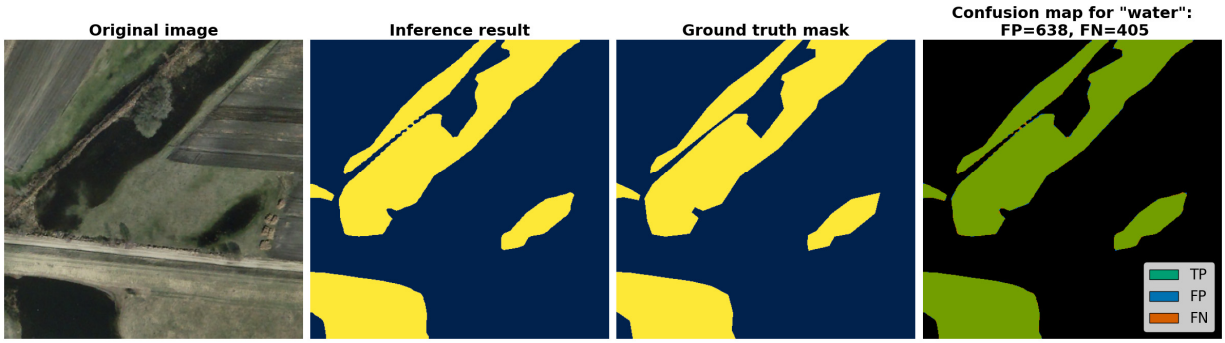

Figure A8. A confusion map for the class "water". The false positives and false negatives are predominantly attributable to the borders.

Woodland, being less dominant and spatially sparse, benefited from a lower threshold. The 300-pixel threshold in Figure A7 effectively eliminated false positives and false negatives, preserving the number of correct predictions. Similarly, for roads, a 600-pixel threshold (Figure A9) balanced the removal of isolated misclassifications with the preservation of linear continuity. The consistency of these results supports the hypothesis that class-specific thresholds enhance evaluation interpretability.

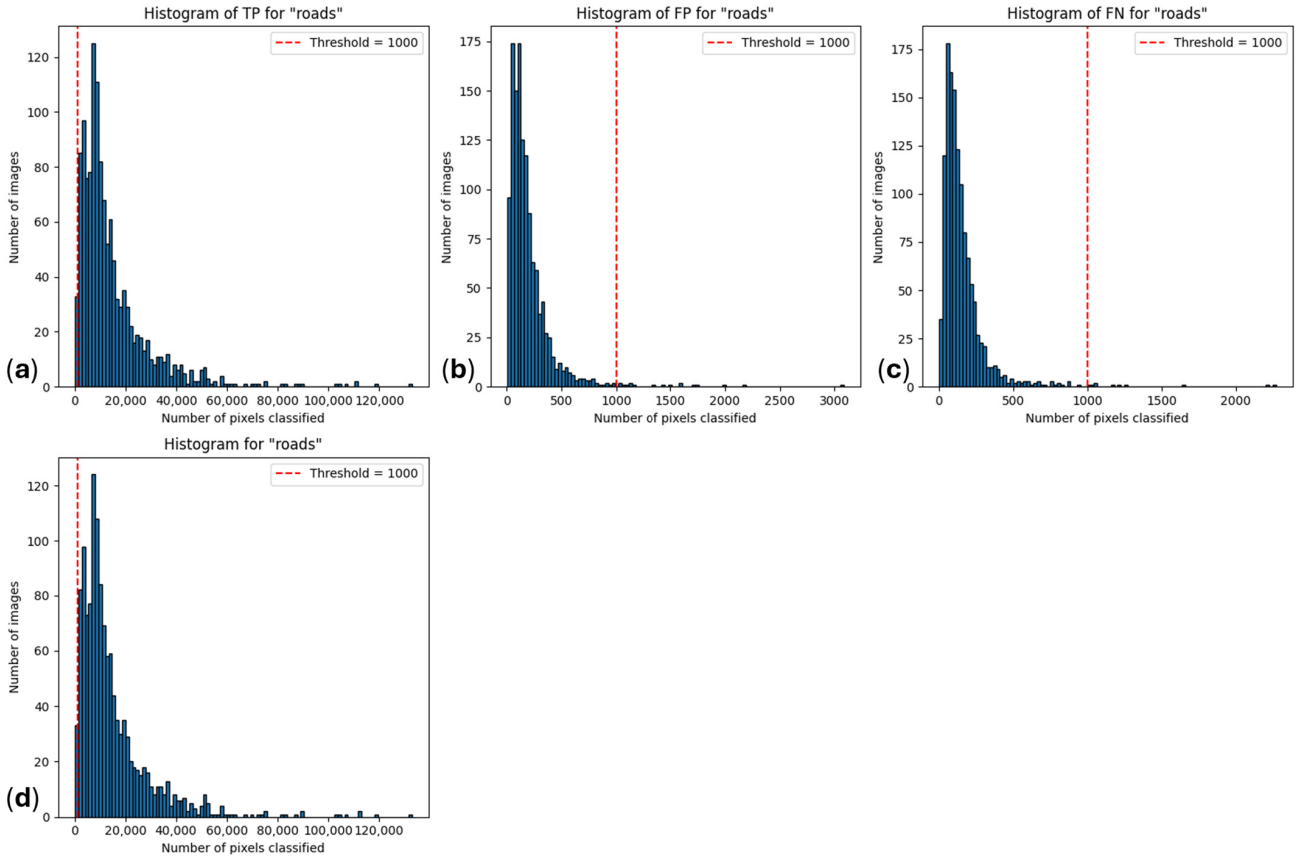

Figure A9. (**a**) A histogram of the number of true positives per image for the class 'roads'. (**b**) A histogram of the number of false positives per image for the class 'roads'. (**c**) A histogram of the number of false negatives per image for the class 'roads'. (**d**) A histogram of the number of ground-truth pixels classified as 'roads'.

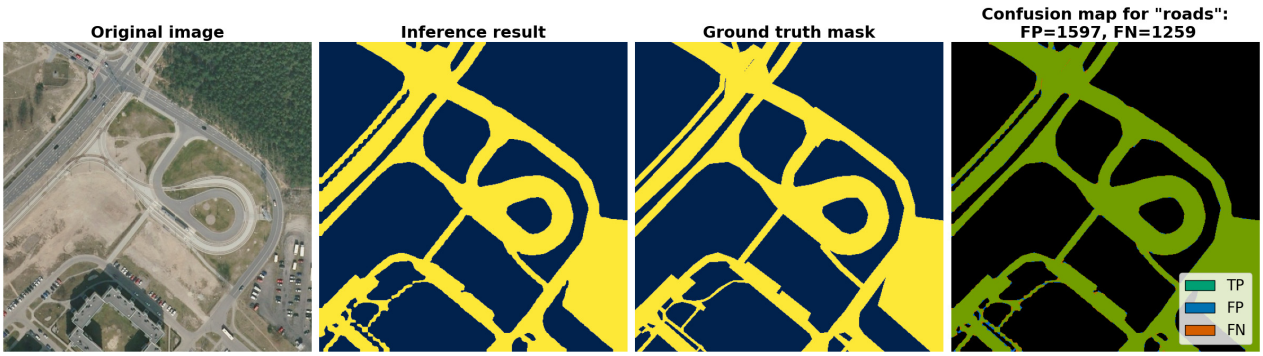

Figure A10. A confusion map for the class "roads". The false positives and false negatives are predominantly attributable to the borders.

Although these tailored thresholds yielded incremental improvements, applying a general 1000-pixel threshold remained a reliable and interpretable default. It performed consistently across most classes and simplified the evaluation process. Especially in operational scenarios or standardized benchmarking, a unified threshold ensures fairness and comparability.

This extended analysis demonstrates that, while class-specific thresholds enhanced per-class clarity, a general threshold of 1000 pixels provided the most consistent overall results. The identified values—1500 pixels for background (Figure A1), 1000 pixels forbuildings (Figure A3), 800 pixels for woodland (Figure A5), 300 pixels for water (Figure A7), and

600 pixels for roads (Figure A9)—illustrate how spatial and structural characteristics guide the selection of appropriate thresholds.

The 1000-pixel threshold eliminated nearly all false positives and false negatives while preserving the number of correct detections. Visual inspection supported these findings, revealing that most FPs and FNs occurred in transition zones. Figures A1–A10 illustrate the practical impact of applying both general and class-specific thresholds. While class-specific tuning provides enhanced precision, the 1000-pixel threshold performs effectively across various land cover categories, confirming its role as the most balanced and reliable choice. This threshold ensures fair evaluation and improves the interpretability of segmentation results in real-world applications.

Appendix B

This appendix investigates the rationale behind the decision not to employ pretrained weights in the evaluated segmentation models. The analysis was based on a comparative evaluation of both pretrained and non-pretrained versions of two widely used architectures: PSPNet and DANet. The objective was to assess the influence of pretraining on the models' behavior in the presence of extreme misclassifications—commonly referred to as outliers—using a range of intersection over union (IoU) thresholds. The analysis was conducted using the same semantic segmentation framework and evaluation pipeline applied throughout this study. Specifically, the models were either trained from scratch or initialized with pretrained weights and then tested under identical conditions. The metric used for outlier identification was based on the number of test images in which the IoU for a given class fell above fixed thresholds: 0.19%, 0.27%, 0.38%, and 0.57%. Additional annotations were included to distinguish between clear model errors, ground-truth inconsistencies, and ambiguous cases.

Two summary tables are provided. Table A1 reports the numbers of network mistakes, ground-truth mistakes, and ambiguous cases for the non-pretrained versions of the two models across all IoU thresholds. Table A2 presents the same metrics for the pretrained versions. The datasets and class structure remained identical, ensuring a direct comparison of model robustness under different initialization conditions. Each table includes four threshold levels to capture the progressive effect of relaxing or tightening the outlier definition.

The results reveal that pretraining did not consistently enhance model robustness related to extreme errors. In several cases, models initialized with pretrained weights exhibited equal or higher outlier rates compared to their non-pretrained counterparts. For example, when pretrained, PSPNet showed a slight increase in the number of outliers detected at the 0.38% IoU threshold, rising from 21 for the non-pretrained version to 24 for the pretrained one. This trend was observed across other thresholds and architectures. These findings suggest that pretraining may introduce latent biases—possibly inherited from the source dataset—that affect the network's ability to generalize under noisy or ambiguous conditions.

Based on the comparative outlier analysis, pretraining does not universally guarantee improved performance in terms of robustness against severe segmentation failures. The results indicate that training from scratch can, in some cases, yield more stable behavior under strict evaluation criteria. Consequently, the decision not to use pretrained weights in this study was justified, as it avoided unintended biases and ensured that the networks' performance was attributable solely to learning from the target dataset.

Table A1. Outlier analysis when PSPNet and DANet were not pretrained.

	PSPNet				DANet			
Thresholds	0.19% (*)	0.27% (*)	0.38% (*)	0.57% (*)	0.19% (*)	0.27% (*)	0.38% (*)	0.57% (*)
Network mistakes								
background	6	6	5	5	4	4	3	3
buildings	0	0	0	0	1	1	1	1
woodland	6	5	4	4	7	7	7	6
water	1	0	0	0	9	9	8	7
road	13	13	12	10	11	10	10	7
Total network mistakes	26	24	21	19	32	31	29	24
Ground-truth mistakes								
background	16	16	16	16	7	7	7	7
buildings	0	0	0	0	1	1	1	1
woodland	16	15	15	15	17	17	17	16
water	0	0	0	0	1	1	0	0
road	11	11	11	10	14	14	13	11
Total ground-truth mistakes	43	42	42	41	40	40	38	35
Ambiguous mistakes								
background	22	21	16	16	56	56	47	42
buildings	1	0	0	0	2	0	0	0
woodland	80	80	66	64	69	62	59	51
water	1	1	0	0	9	9	7	6
road	37	32	20	16	36	29	20	14
Total ambiguous mistakes	141	134	102	96	172	156	133	113
Total								
background	44	43	37	37	67	67	57	52
buildings	1	0	0	0	4	2	2	2
woodland	102	100	85	83	93	86	83	73
water	2	1	0	0	19	19	15	13
road	61	56	43	36	61	53	43	32
Total	210	200	165	156	244	227	200	172

(*) Percentage of IoU.

Table A2. Outlier analysis when PSPNet and DANet were pretrained.

	Pretrained PSPNet				Pretrained DANet			
Thresholds	0.19% (*)	0.27% (*)	0.38% (*)	0.57% (*)	0.19% (*)	0.27% (*)	0.38% (*)	0.57% (*)
Network mistakes								
background	7	7	6	6	8	7	7	7
buildings	0	0	0	0	0	0	0	0
woodland	8	8	8	7	4	4	4	3
water	2	2	1	1	4	4	3	3
road	10	9	9	9	11	9	9	8
Total network mistakes	27	26	24	23	27	24	23	21
Ground-truth mistakes								
background	16	16	16	16	7	7	7	7
buildings	0	0	0	0	1	1	1	1
woodland	16	15	15	15	17	17	17	16
water	0	0	0	0	1	1	0	0
road	11	11	11	10	14	14	13	11
Total ground-truth mistakes	43	42	42	41	40	40	38	35
Ambiguous mistakes								
background	32	26	21	17	40	39	28	22
buildings	1	1	0	0	1	1	0	0
woodland	63	59	49	44	62	51	46	38
water	2	2	2	1	4	4	3	3
road	22	15	11	8	24	20	14	6
Total ambiguous mistakes	120	103	83	70	131	115	91	69
Total								
background	55	49	43	39	55	53	42	36
buildings	1	1	0	0	2	2	1	1
woodland	87	82	72	66	83	72	67	57
water	4	4	3	2	9	9	6	6
road	43	35	31	27	49	43	36	25
Total	190	171	149	134	198	179	152	125

(*) Percentage of IoU.

References

1. Rong, C.; Fu, W. A Comprehensive Review of Land Use and Land Cover Change Based on Knowledge Graph and Bibliometric Analyses. *Land* **2023**, *12*, 1573. [CrossRef]
2. Luo, Z.; Yang, W.; Yuan, Y.; Gou, R.; Li, X. Semantic segmentation of agricultural images: A survey. *Inf. Process. Agric.* **2024**, *11*, 172–186. [CrossRef]
3. Nkwocha, C.L.; Wang, N. Deep learning-based semantic segmentation with novel navigation line extraction for autonomous agricultural robots. *Discov. Artif. Intell.* **2025**, *5*, 73. [CrossRef]
4. Lei, L.; Yang, Q.; Yang, L.; Shen, T.; Wang, R.; Fu, C. Deep learning implementation of image segmentation in agricultural applications: A comprehensive review. *Artif. Intell. Rev.* **2024**, *57*, 149. [CrossRef]
5. Dhanya, V.; Subeesh, A.; Kushwaha, N.; Vishwakarma, D.K.; Kumar, T.N.; Ritika, G.; Singh, A. Deep learning based computer vision approaches for smart agricultural applications. *Artif. Intell. Agric.* **2022**, *6*, 211–229. [CrossRef]
6. Attri, I.; Awasthi, L.K.; Sharma, T.P.; Rathee, P. A review of deep learning techniques used in agriculture. *Ecol. Inform.* **2023**, *77*, 102217. [CrossRef]
7. Boguszewski, A.; Batorski, D.; Ziemba-Jankowska, N.; Dziedzic, T.; Zambrzycka, A. LandCover.ai: Dataset for automatic mapping of buildings, woodlands, water and roads from aerial imagery. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021. [CrossRef]
8. Maggiori, E.; Tarabalka, Y.; Charpiat, G.; Alliez, P. Convolutional neural networks for large-scalere mote-sensing image classification. *IEEE Trans. Geosci. Remote Sens.* **2016**, *55*, 645–657. [CrossRef]
9. Wang, W.; Wang, X. BAFNet: Bilateral Attention Fusion Network for Lightweight Semantic Segmentation of Urban Remote Sensing Images. *arXiv* **2024**, arXiv:2409.10269. [CrossRef]
10. Liu, J.; Wu, J.; Xie, H.; Xiao, D.; Ran, M. Semantic Segmentation of Urban Remote Sensing Images Based on Deep Learning. *Appl. Sci.* **2024**, *14*, 7499. [CrossRef]
11. Weir, N.; Lindenbaum, D.; Bastidas, A.; Etten, A.V.; McPherson, S.; Shermeyer, J.; Kumar, V.; Tang, H. Spacenet mvoi: A multi-view overhead imagery dataset. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 992–1001. [CrossRef]
12. Bengana, N.; Heikkilä, J. Improving land cover segmentation across satellites using domain adaptation. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *14*, 1399–1410. [CrossRef]
13. Scheibenreif, L.; Hanna, J.; Mommert, M.; Borth, D. Self-supervised vision transformers for land-cover segmentation and classification. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, New Orleans, LA, USA, 19–20 June 2022; pp. 1421–1430. [CrossRef]
14. Chiu, M.T.; Xu, X.; Wei, Y.; Huang, Z.; Schwing, A.G.; Brunner, R.; Khachatrian, H.; Karapetyan, H.; Dozier, I.; Rose, G.; et al. Agriculture-vision: A large aerial image database for agricultural pattern analysis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 14–19 June 2020; pp. 2828–2838. [CrossRef]
15. Zhao, X.; Yuan, Y.; Song, M.; Ding, Y.; Lin, F.; Liang, D.; Zhang, D. Use of unmanned aerial vehicle imagery and deep learning unet to extract rice lodging. *Sensors* **2019**, *19*, 3859. [CrossRef] [PubMed]
16. Anand, T.; Sinha, S.; Mandal, M.; Chamola, V.; Yu, F.R. AgriSegNet: Deep aerial semantic segmentation framework for IoT-assisted precision agriculture. *IEEE Sens. J.* **2021**, *21*, 17581–17590. [CrossRef]
17. Liu, G.; Bai, L.; Zhao, M.; Zang, H.; Zheng, G. Segmentation of wheat farmland with improved U-Net on drone images. *J. Appl. Remote Sens.* **2022**, *16*, 034511. [CrossRef]
18. Mortensen, A.K.; Dyrmann, M.; Karstoft, H.; Jørgensen, R.N.; Gislum, R. Semantic segmentation of mixed crops using deep convolutional neural network. In Proceedings of the International Conference on Agricultural Engineering (CIGR-AgEng), Aarhus, Denmark, 26–29 June 2016.
19. Wang, A.; Xu, Y.; Wei, X.; Cui, B. Semantic segmentation of crop and weed using an encoder-decoder network and image enhancement method under uncontrolled outdoor illumination. *IEEE Access* **2020**, *8*, 81724–81734. [CrossRef]
20. Sahin, H.M.; Miftahushudur, T.; Grieve, B.; Yin, H. Segmentation of weeds and crops using multispectral imaging imaging and CFR-enhanced U-Net. *Comput. Electron. Agric.* **2023**, *211*, 107956. [CrossRef]
21. Castillo-Navarro, J.; Le Saux, B.; Boulch, A.; Audebert, N.; Lefèvre, S. Semi-Supervised Semantic Segmentation in Earth Observation: The MiniFrance suite, dataset analysis and multi-task network study. *Mach. Learn.* **2022**, *111*, 3125–3160. [CrossRef]
22. Md Jelas, I. Deforestation detection using deep learning-based semantic segmentation techniques: A systematic review. *Front. For. Glob. Change* **2024**, *7*, 1300060. [CrossRef]
23. MMS Contributors. MMSegmentation: OpenMMLab Semantic Segmentation Toolbox and Benchmark. 2020. Available online: https://github.com/open-mmlab/mmsegmentation (accessed on 3 June 2025).
24. Zhu, Z.; Xu, M.; Bai, S.; Huang, T.; Bai, X. Asymmetric non-local neural networks for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 593–602. [CrossRef]

25. He, J.; Deng, Z.; Zhou, L.; Wang, Y.; Qiao, Y. Adaptive pyramid context network for semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7519–7528. [CrossRef]
26. Yu, C.; Gao, C.; Wang, J.; Yu, G.; Shen, C.; Sang, N. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *Int. J. Comput. Vis.* **2021**, *129*, 3051–3068. [CrossRef]
27. Huang, Z.; Wang, X.; Huang, L.; Huang, C.; Wei, Y.; Liu, W. Ccnet: Criss-cross attention for semantic segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 603–612. [CrossRef]
28. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual attention network for scene segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 3146–3154. [CrossRef]
29. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 801–818. [CrossRef]
30. Wu, H.; Zhang, J.; Huang, K.; Liang, K.; Yu, Y. Fastfcn: Rethinking dilated convolution in the backbone for semantic segmentation. *arXiv* **2019**, arXiv:1903.11816. [CrossRef]
31. Poudel, R.P.; Liwicki, S.; Cipolla, R. Fast-scnn: Fast semantic segmentation network. *arXiv* **2019**, arXiv:1902.04502. [CrossRef]
32. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440. [CrossRef]
33. Liu, J.; Zhou, W.; Cui, Y.; Yu, L.; Luo, T. GCNet: Grid-like context-aware network for RGB-thermal semantic segmentation. *Neurocomputing* **2022**, *506*, 60–67. [CrossRef]
34. Zhao, H.; Qi, X.; Shen, X.; Shi, J.; Jia, J. Icnet for real-time semantic segmentation on high-resolution images. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; pp. 405–420. [CrossRef]
35. Huang, L.; Yuan, Y.; Guo, J.; Zhang, C.; Chen, X.; Wang, J. Interlaced Sparse Self-Attention for Semantic Segmentation. *arXiv* **2019**, arXiv:1907.12273. [CrossRef]
36. Yuan, Y.; Chen, X.; Wang, J. Object-Contextual Representations for Semantic Segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Glasgow, UK, 23 August–7 September 2020; pp. 173–190. [CrossRef]
37. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890. [CrossRef]
38. Wang, R.; Jiang, H.; Li, Y. UPerNet with ConvNeXt for Semantic Segmentation. In Proceedings of the 2023 IEEE 3rd International Conference on Electronic Technology, Communication and Information (ICETCI), Changchun, China, 26–28 May 2023; pp. 764–769. [CrossRef]
39. Fernandes, A.A.; Koehler, M.; Konstantinou, N.; Pankin, P.; Paton, N.W.; Sakellariou, R. Data preparation: A technological perspective and review. *SN Comput. Sci.* **2023**, *4*, 425. [CrossRef]
40. Kotsiantis, S.; Kanellopoulos, D.; Pintelas, P. Handling imbalanced datasets: A review. *GESTS Int. Trans. Comput. Sci. Eng.* **2006**, *30*, 25–36.
41. Werner de Vargas, V.; Schneider Aranda, J.A.; dos Santos Costa, R.; da Silva Pereira, P.R.; Victória Barbosa, J.L. Imbalanced data preprocessing techniques for machine learning: A systematic mapping study. *Knowl. Inf. Syst.* **2023**, *65*, 31–57. [CrossRef]
42. Johnson, J.M.; Khoshgoftaar, T.M. Survey on deep learning with class imbalance. *J. Big Data* **2019**, *6*, 1–54. [CrossRef]