aerospace

MDPI

Article

Deep Learning-Based Semantic Segmentation of Urban Areas Using Heterogeneous Unmanned Aerial Vehicle Datasets

Ahram Song

Department of Location-Based Information System, Kyungpook National University, Gyeongsang-daero, Gyeongsangbuk-do, Sangju 2559, Republic of Korea; ars@knu.ac.kr

Abstract: Deep learning techniques have recently shown remarkable efficacy in the semantic segmentation of natural and remote sensing (RS) images. However, these techniques heavily rely on the size of the training data, and obtaining large RS imagery datasets is difficult (compared to RGB images), primarily due to environmental factors such as atmospheric conditions and relief displacement. Unmanned aerial vehicle (UAV) imagery presents unique challenges, such as variations in object appearance due to UAV flight altitude and shadows in urban areas. This study analyzed the combined segmentation network (CSN) designed to train heterogeneous UAV datasets effectively for their segmentation performance across different data types. Results confirmed that CSN yielded high segmentation accuracy on specific classes and can be used on diverse data sources for UAV image segmentation. The main contributions of this study include analyzing the impact of CSN on segmentation accuracy, experimenting with structures with shared encoding layers to enhance segmentation accuracy, and investigating the influence of data types on segmentation accuracy.

Keywords: unmanned aerial vehicle (UAV); dataset; semantic segmentation; combined segmentation network; semantic drone dataset (SDD); UAVid

Citation: Song, A. Deep Learning-Based Semantic Segmentation of Urban Areas Using Heterogeneous Unmanned Aerial Vehicle Datasets. *Aerospace* **2023**, *10*, 880. https://doi.org/10.3390/aerospace10100880

Academic Editor: Peng Wei

Received: 12 September 2023
Revised: 8 October 2023
Accepted: 11 October 2023
Published: 12 October 2023

1. Introduction

Semantic segmentation involves categorizing a class label for each pixel in an image and is used for tasks such as scene recognition, land-cover mapping, vehicle detection, and disaster damage assessment [1,2]. In remote sensing (RS), semantic segmentation is analogous to image classification [3]. Over the past two decades, various traditional pixel-based and object-based methods have been used for the semantic segmentation of RS images [4]; however, these methods exhibit several drawbacks, such as the requirement for appropriate parameter settings and thresholds and pose algorithmic complexity.

Deep learning is a new field division of machine learning [5], which automatically derives features tailored for targeted classification tasks, making such methods a better choice for handling complicated scenarios [6]. Deep learning approaches have recently demonstrated their effectiveness in semantic segmentation [5], particularly for natural RGB and RS image analysis. Several networks based on U-Net [7], the most widely adopted architecture for semantic segmentation due to its flexibility, optimized design, and success across all medical image modalities [8], are used for RS image segmentation. A convolutional network design was also developed for rapid and accurate image segmentation [6]. Diakogiannis et al. [9] introduced ResUNet, which used a U-Net encoder–decoder backbone with residual connections, atrous convolutions, pyramid scene parsing pooling, and multitasking inference. The ResUNet-a performs robustly even when dealing with highly imbalanced classes in high-resolution RS images. Li et al. [10] proposed the spatial and channel attention network (SCAttNet), which integrated lightweight spatial and channel attention modules for high-resolution RS images. SCAttNet achieved superior semantic segmentation outputs compared to traditional models such as U-Net [7], SegNet [11], and DeepLabV3+ [12].

Several networks for RS image segmentation are being developed; however, the performance of deep learning models primarily depends on the size of the dataset. Compared with natural RGB datasets, RS datasets have limited sizes with insufficient training samples, which is a critical challenge. This is because environmental factors, such as atmospheric effects and relief displacement, impact RS images acquired from satellites and airborne sources. Additionally, large-scale image collection under similar conditions is arduous, and trained personnel are required to label objects in RS images [6].

Strategies that use transfer learning methods to leverage datasets collected from diverse platforms were proposed to address the challenges posed by limited data availability [13,14]. Panboonyuen et al. [13] performed semantic segmentation using Landsat-8 satellite images and the high-resolution International Society for Photogrammetry and RS (ISPRS) Vaihingen Challenge Dataset [15]. They employed a global convolutional network that captured images with different spatial and spectral resolutions by extracting multiscale features from distinct stages of the network. Channel attention was used to distinguish and select the most discriminative filters. Cui et al. [14] similarly used the pre-trained DensNet-121, originally trained on the RGB bands of the ImageNet dataset, as the encoder subnetwork to segment the Gaofen-2 satellite images with four spectral bands (red, green, blue, and near-infrared) and a spatial resolution of 4 m. Multiscale information was fused by integrating dense connections into the decoder subnetwork. Although transfer learning approaches efficiently handle disparate datasets, they can introduce complexities. Moreover, as the volume of data increases, the effectiveness of the transfer segmentation rules can be adversely impacted by inclusive data.

An alternative strategy for managing heterogeneous data involves sharing specific layers between different datasets during training. Meletis and Dubbelman [16] introduced a convolutional network with hierarchical classifiers for the semantic segmentation of street scenes using three distinct heterogeneous datasets. They used a hierarchical classification loss by amalgamating training data from disjoint but semantically related datasets. Ghassemi et al. [17] proposed an encoder–decoder convolutional architecture that can be applied to different images from those utilized during training by leveraging deep residual architectures. Song and Kim [3] employed a combined U-Net approach using shared initial convolutional layers to segment ground-level datasets, such as Cityscape [18], and airborne image datasets, like the ISPRS dataset. However, while the combined U-Net approach was based on the simplified U-Net architecture, the structure of the U-Net model was not fully preserved. Moreover, the model's performance with U-Net architecture and the potential impact on the segmentation accuracy of the shared layer's block were not adequately considered.

Effective segmentation methods have been proposed primarily for heterogeneous RS datasets, such as satellite and aerial imagery datasets; however, the dataset characteristics have not been comprehensively analyzed. Recently, unmanned aerial vehicle (UAV) images have been increasingly used across various domains because of their higher spatial resolution compared to images captured via satellites and manned aircraft, particularly in urban areas. The diversity of recognizable labels in a UAV image is more pronounced than in satellite and aerial images. Furthermore, even when dealing with the same objects, their size and features within an image can vary based on the flight altitude of UAVs and the impact of shadows. Vegetated regions can be distinguished based on their spectral characteristics. Contrarily, classifying urban areas from RS images based on spectral characteristics is challenging because urban objects, such as vehicles, buildings, and roads, have varying shapes and colors. Thus, segmentation rules from heterogeneous UAV datasets captured in diverse environmental conditions must be learned.

A combined segmentation network (CSN) was used to address the aforementioned challenges in training heterogeneous UAV datasets. CSN used shared layers in the encoder block to accommodate diverse input datasets and learned segmentation rules from heterogeneous UAV datasets. As CSN builds upon the foundational U-Net architecture, it maintains a straightforward structure while benefiting from sharing layers and losses to learn from

distinct datasets. Moreover, CSN performance was compared when sharing only the initial encoder layers versus the entire encoder block to understand the impact of shared layers comprehensively. The performance of CSN when training the UAV and airborne datasets was also compared to understand the influence of data types on segmentation accuracy.

The main contributions are as follows:

1. The training was performed without adjusting the spatial resolution and class types of different UAV datasets. The segmentation accuracy between only training a single dataset versus different datasets simultaneously was compared to understand the impact of data types on the accuracy of CSN;
2. Based on the finding that CSN can enhance the segmentation accuracy of specific classes [3], a method was proposed to enhance the segmentation accuracy of the UAV datasets by modifying the shared encoding layer structure;
3. To determine whether the RS images acquired from various platforms can enhance the segmentation accuracy of UAV images, heterogeneous UAV datasets and airborne datasets were used for training. Based on the results, the type of dataset that can be used with CSN for UAV image training was determined.

2. Methods

2.1. Combined Segmentation Network

The previously introduced CSN [3] was built upon the simplified U-Net architecture and was designed to handle inputs from two distinct data sources. The original layers of U-Net were retained, and two CSNs with modified structures of shared layers were used to extract detailed features and analyze their influence on segmentation performance based on the structure of the shared layers: Case 1 with shared initial two convolutional blocks and Case 2 with shared all encoding blocks for both datasets. Figure 1a,b show the architectural diagrams for these two CSNs. For encoding, CSN employed several blocks shared between both datasets during training. Convolutional blocks comprise a pair of 2D convolutional layers and batch-normalized and activation layers. Inputs were drawn from two distinct data sources with input dimensions of $n \times n$. To accommodate the reception of two dissimilar data sources and facilitate the sharing of initial layers, the dimensions of the inputs were resized to (512 and 512). Shared blocks collectively received and shared training parameter weights from the two sources. Consequently, at the end of the encoding phase, the feature map attained the dimensions of $\frac{n}{16} \times \frac{n}{16}$. These shared layers adeptly gleaned common information spanning diverse datasets and domains. The encoder phase generally functions as a feature extractor, capturing an abstract representation of the input image. This approach was adopted because the initial convolutional layers could effectively learn common information relevant to both data sources. Lee et al. [19] previously affirmed that shared domain layers can considerably enhance CNN optimization, outperforming the solitary datasets used for enhancing classification accuracy. However, their study only shared the middle layers of the network and used only three satellite images for combined network training. Herein, sharing was extended to the encoder blocks during learning, with subsequent blocks dedicated to dataset-specific segmentation tasks. Then, the feature maps underwent decoding after encoding, signifying distinct paths for each dataset with independent training weights.

The decoding blocks primarily employ a transposed convolutional layer to upscale the feature map. Convolutional 2D transpose blocks comprise a pair of 2D convolutional transpose layers and batch-normalized, activation, and concentrate layers. Decoding concluded with a feature map of dimensions $n \times n \times c$, aligning with the shape of the data labels, where c denotes the number of label classes.

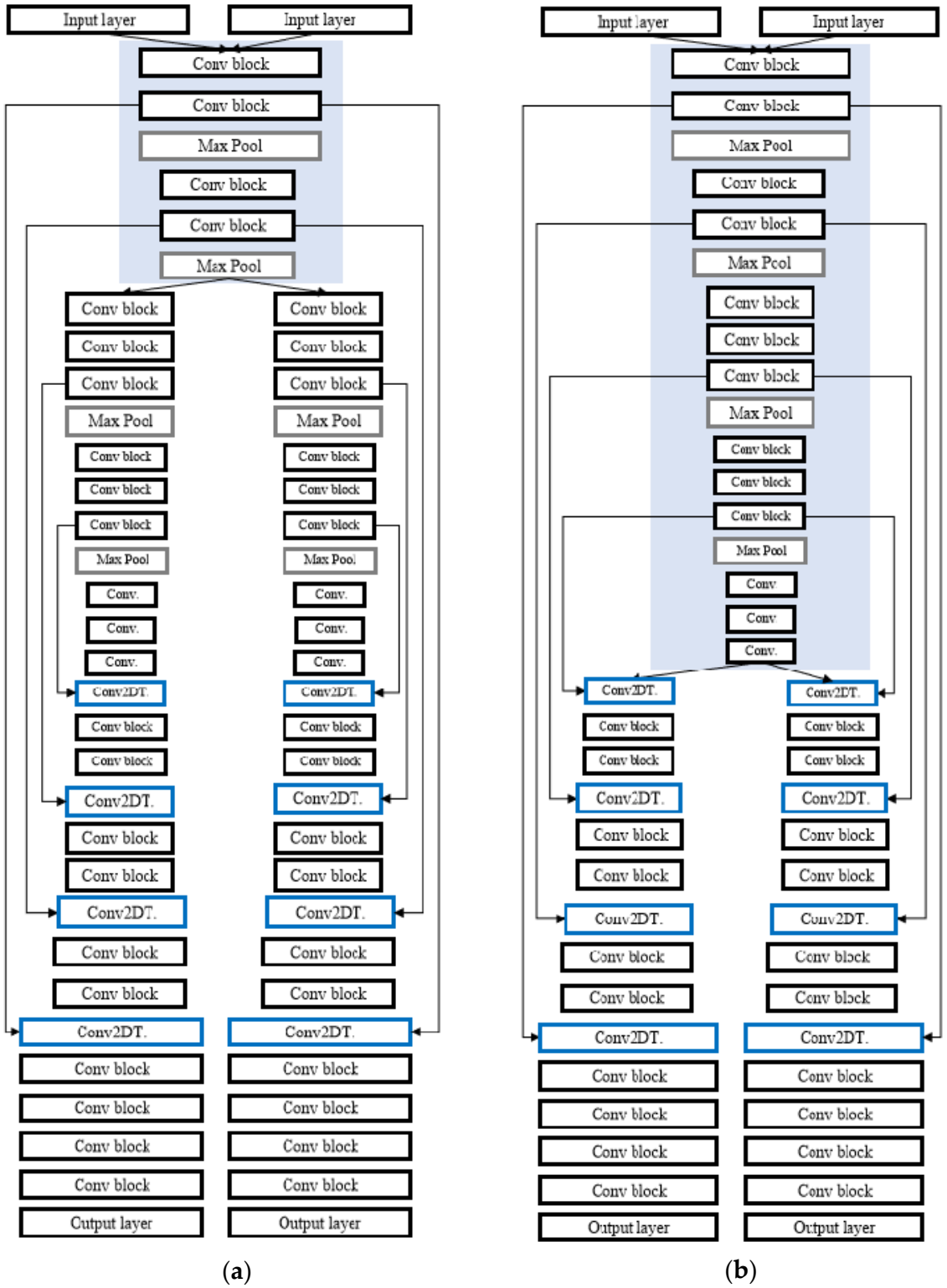


Figure 1. Two versions of CSNs: (**a**) Case 1 with shared initial two convolutional blocks and (**b**) Case 2 with shared all encoding blocks for both datasets. "conv block" or "conv." refers to a 2D convolutional block, and "Conv2dT." refers to a 2D convolutional transpose block. A convolutional block comprises a pair of 2D convolutional layers and batch-normalized and activation layers. A convolutional 2D transpose block comprises a pair of 2D convolutional transpose layers and batch-normalized, activation, and concentrate layers. The blue box represents the shared blocks influenced by both datasets.

Heterogeneous datasets were used for training, during which CSN simultaneously processed two inputs using the shared encoder blocks. The network was trained with a combined weighted loss, L_c, comprising the weighted summation of losses from the two decoding paths. The model was updated based on these combined weighted losses. The losses for the first and second paths were designated as L_{n1} and L_{n2}, respectively, whereas the spatial cross-entropy loss was mathematically defined, as shown in Equation (1).

$$L_c = \omega_1 \cdot L_{n1} + \omega_2 \cdot L_{n2} \tag{1}$$

where ω_1 and ω_2 are the weights for each path. A higher weight was assigned to prioritize the main dataset to be trained. Thus, the weight ratio between the main and auxiliary datasets was 8:2.

2.2. Evaluation Metrics

Two widely used metrics, the F1 score and kappa statistics, were used to measure the semantic segmentation performance. The F1 score is effective for evaluating the results for imbalanced classes. The F1 score was calculated for each class due to the imbalance in the dataset classes used herein. The F1 score can be described in terms of true positive (TP), true negative (TN), false negative (FN), and false positive (FP) and is calculated as the harmonic mean of precision and recall. Precision and recall quantify the accurately identified positive cases among all predicted positive and actual positive cases.

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \tag{2}$$

The kappa statistic (Equation (3)), referred to as Cohen's kappa, is a statistical tool that measures the consensus between two evaluators or classifiers that goes beyond what could be attributed to random chance. It considers the level of agreement achieved by mere coincidence and offers a more resilient assessment of the agreement between raters or classifiers. Kappa values range from -1 to 1: negative values imply less agreement than expected by chance, values around 0 signify agreement akin to chance, and positive values denote agreement surpassing what chance would account for.

$$\textit{Kappa Statistic} = \frac{2 \times (\text{TP} \times \text{TN} - \text{FN} \times \text{FP})}{(\text{TP} + \text{FP})(\text{FP} + \text{TN}) + (\text{TP} + \text{FN})(\text{FN} + \text{TN})} \tag{3}$$

3. Materials

3.1. Datasets

Two UAV datasets, i.e., the UAVid dataset [20] and semantic drone dataset (SDD) [21], and an aerial dataset, i.e., ISPRS Potsdam, were utilized. These datasets contain various classes and spatial resolutions because they were acquired in diverse environments. Notably, unlike the ISPRS Potsdam dataset, UAV images acquired from the UAVid dataset and SDD have different spatial resolutions because of flexible flight schedules and the altitudes of UAVs [20]. Although these datasets have different spatial resolutions and classes, the images were acquired primarily in urban areas, where roads, buildings, and vehicles were the predominant features.

Herein, the segmentation accuracy of UAV imagery was enhanced using various CSN structures from heterogeneous datasets. The SDD was chosen as the reference dataset, based on which two versions of CSN were trained individually with the UAVid and ISPRS Potsdam datasets. Table 1 describes these three datasets in detail. As a preprocessing step, all images were subdivided into small patches of 512 $\times$ 512 to facilitate their simultaneous training on CSN.

Table 1. Information about datasets.

Dataset	Platform	Type of View	Flight Altitude	Number of Images (Number of Classes)	Category
UAVid	UAV	side-view	0–50 m	Train: 3300 Val: 1500 (class 8)	Building Road Static car Tree Low vegetation Human Moving Car Background clutter
Semantic drone dataset (SDD)	UAV	bird-view	5–30 m	Train: 3300 Val: 1500 (class 23)	Background Paved area Dirt Grass Gravel Water Rocks Pool Vegetation Roof Wall Window Door Fence Fence pole Person Dog Car Bicycle Tree Bald tree AR marker Obstacle Conflicting
ISPRS Potsdam	Aircraft	bird-view	800 m	Train: 3300 Val: 1500 (class 6)	Impervious surfaces Building Low vegetation Tree Car Clutter/background

3.1.1. Semantic Drone Dataset

The SDD contains UAV images of urban areas. The imagery depicts over 20 houses from a nadir view, captured at altitudes of 5–30 m above the ground. In general, the flight height of the SDD is lower than that of the UAVid. Thus, even within the same class, the SDD has a higher spatial resolution than the UAVid dataset. For instance, the spatial resolution of the "Car" in the SDD is higher than that in the UAVid dataset, making it appear larger. Moreover, unlike the UAVid dataset, images in the SDD (6000 × 4000 pixels) were captured from a bird's-eye view. Originally, the training set contained 400 images, and 200 images were utilized for the test set. When cropping the original images to a size of 512 × 512 pixels, only a portion of the object might be included due to the high spatial resolution of the SDD. Thus, an initial crop to 3000 × 2000 pixels was performed, followed by resizing to 512 × 512 pixels. Image augmentation techniques, such as rotation, cropping, flipping, brightness adjustments, noise addition, and blurring, were then used to increase the number of training images in line with other datasets. Finally, randomly selected 3300 cropped images were used for training, and 1500 images were used for validation.

The label classes in the SDD dataset were limited to 23 categories, including trees, rocks, dogs, fences, grass, water, cars, fence poles, other vegetation, paved areas, bicycles, windows, dirt, pools, roofs, doors, gravel, people, walls, and obstacles. Figure 2 shows the examples of cropped images and corresponding labels of the SDD.

3.1.2. UAVid Dataset

The UAVid dataset comprises 300 images, each with dimensions of 4096 × 2160 or 3840 × 2160. Among these, 120 and 80 images were used for training and validation, respectively. The remaining images were used for testing. To train using the same size of images as in the SDD, images were cropped to 512 × 512 pixels. A total of 3300 randomly

selected cropped images were used for training, and 1500 images were used for validation. These images were captured in complex urban settings containing stationary and moving objects observed from oblique angles at an approximate flight height of 50 m. The UAVid images were captured from a side view as opposed to the SDD and ISPRS datasets.

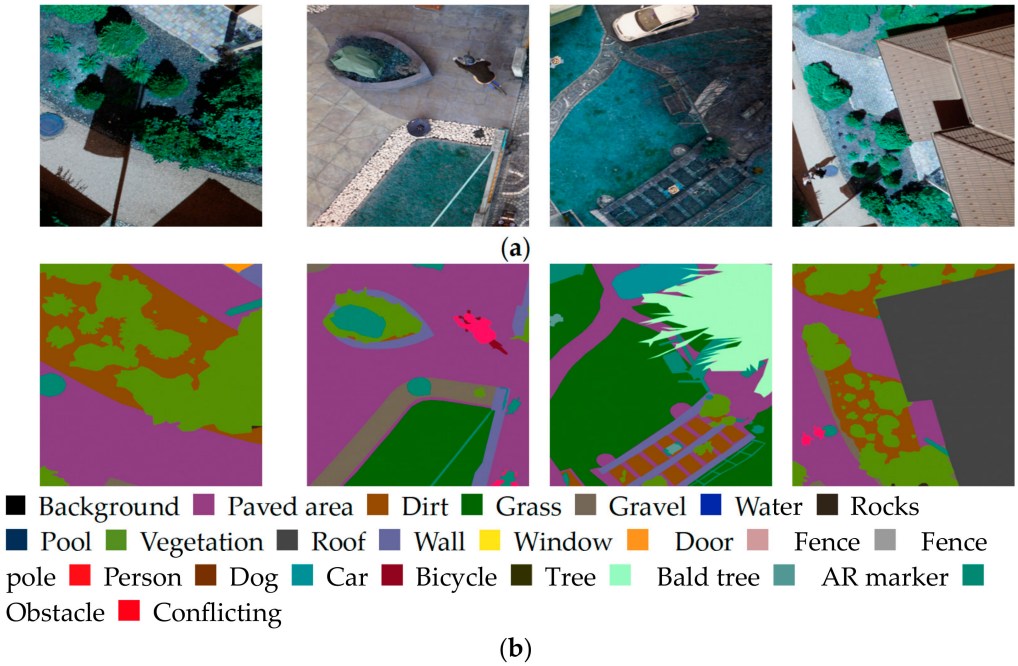


Figure 2. Examples from the SDD (**a**) images and (**b**) labels.

The UAVid dataset contains eight distinct classes: building, road, static car, tree, low vegetation, human, moving car, and background clutter. The building class includes residential houses and under-construction buildings. The static car class comprises stationary cars, and the moving car class includes cars in motion on the road. A road class denotes a surface intended for legal car movement, excluding parking lots and sidewalks, categorized as background clutter. The majority of UAVid images predominantly feature buildings, trees, and roads. In contrast, moving and static cars and human figures collectively account for approximately 3% of the total class distribution. Figure 3 shows the examples of RGB images obtained from the UAVid dataset with corresponding labels.

3.1.3. ISPRS Potsdam Semantic Labeling Dataset

The ISPRS Potsdam dataset comprises 38 patches of ortho-rectified aerial red–green–blue–near-infrared (RGBIR) images, each of 6000 × 6000 pixels and a spatial resolution of 8 cm (the flight altitude above ground: 800 m). The spatial resolution of the ISPRS dataset is inferior to that of the SDD and UAVid datasets. The ISPRS dataset offers three bands with infrared–red–green (IR–R–G), red–green–blue (R–G–B), and four bands with red–green–blue–infrared (R–G–B–IR) combinations. Herein, only RGB images from 24 patches were employed for concurrent training with other datasets. Similar to the SDD and UAVid datasets, images were cropped to a size of 512 × 512 pixels. Randomly selected, 3300 images were used for training, and 1500 images were used for validation.

These images contain six classes: impervious surfaces, buildings, low vegetation, trees, cars, and background/clutter. Impervious surfaces comprise roads and sidewalks. Objects other than the five specified materials, such as water bodies, are categorized under the background/clutter class. Figure 4 shows examples of cropped images and corresponding labels of the ISPRS Potsdam dataset.

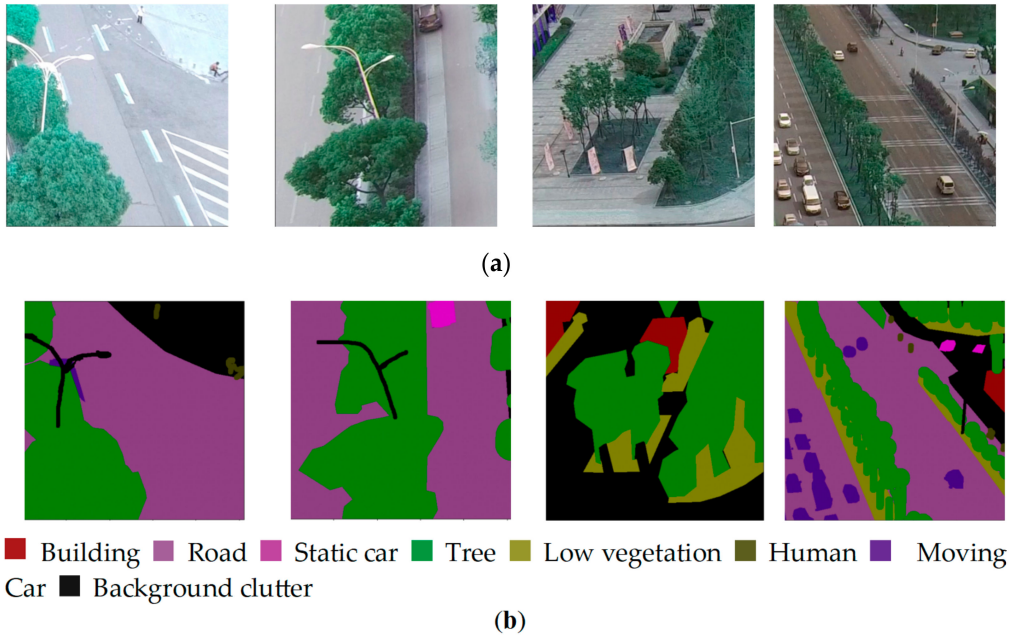


Figure 3. Examples from the UAVid dataset (**a**) images and (**b**) corresponding labels.

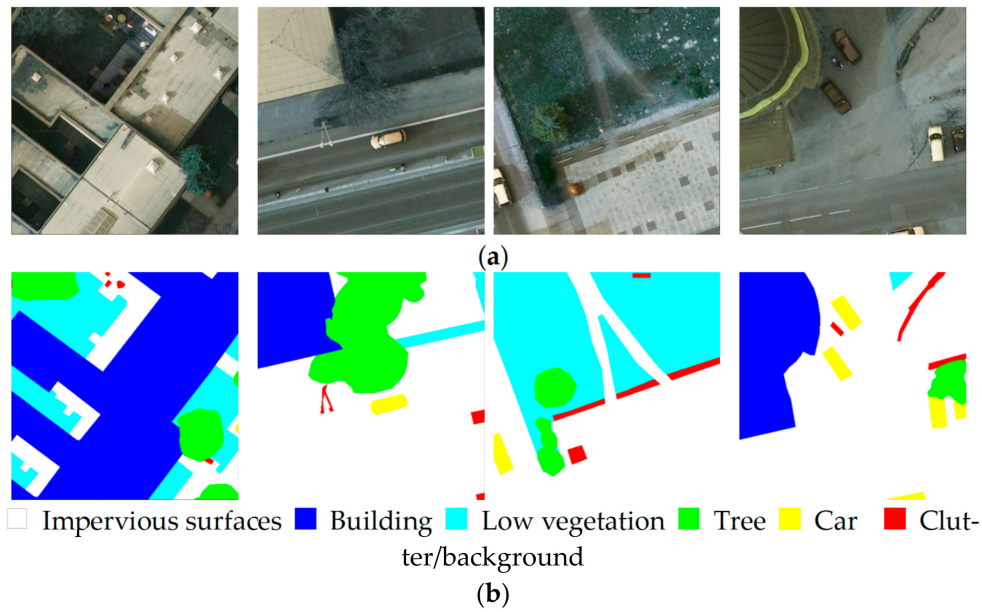


Figure 4. Examples from the ISPRS Potsdam dataset of (**a**) images and (**b**) corresponding labels.

3.2. Test Images

The flight altitudes of UAVs vary when acquiring images over urban areas, leading to differences in spatial resolutions. Additionally, images are captured from various angles, such as side view, as seen in the UAVid dataset, because UAVs can be more susceptible to factors such as wind and weather conditions compared to traditional RS platforms. These differences in spatial resolutions and viewing angles of images imply that each dataset may contain diverse classes, and the same objects may appear in images with different sizes and proportions.

Herein, different UAV datasets were trained using CSN, and segmentation accuracy was evaluated on artificially transformed images as the reference data. The segmentation accuracy was determined using distorted images to assess how efficiently CSN can classify the same objects that appear differently in images when trained simultaneously

on different datasets. Due to the variations in size, shape, and proportions of the same objects across different UAV images, distortions were applied to the original UAV image using augmentation tools. Figure 5 shows the results of the augmentations applied to the original SDD images, mainly including rotation, angle variations, and random cropping. Augmentation was performed using albumentations, which is a Python library for fast and flexible image augmentations, to observe significant enlargements of vehicles and trees as well as the deformations in the shape of vehicles and roads in the images. The test images were randomly selected and transformed from the SDD dataset, which was not used for training. A total of 500 images were used for testing.

(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

Figure 5. Examples of the test images generated from (**a**) the original SDD image by applying (**b**) center crop, (**c**) random rotation of 90°, (**d**) grid distortion, (**e**) random crop, and (**f**) shift scale rotate.

4. Experiments and Results

4.1. Training Settings

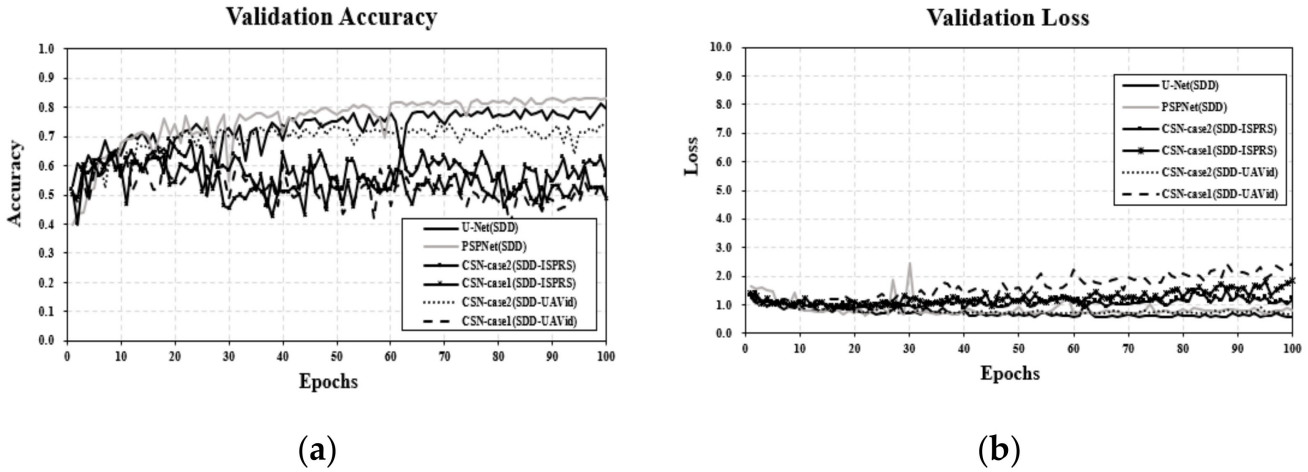
Model implementation and evaluation were conducted in Google Colab Pro+, a Jupyter Notebook environment with pre-installed packages. From the three datasets, 3300 images were used for training, and 1500 images were trained for validation. As CSN can simultaneously train on two datasets, 6600 images were used for training, and 3000 images were used for validation.

To compare the performance of CSN, representative semantic segmentation networks, such as U-Net [7] and the Pyramid Scene Parsing Network (PSPNet) [22], were used to train the SDD. As CSN follows the U-Net architecture, the training efficiencies of U-Net were analyzed to compare with CSN. PSPNet employs a pyramid parsing module to leverage global context information through context aggregation based on different regions.

There are two main hyper-parameters for training CSN: learning rate and epoch. The learning rate controls the extent to which the network weights are adjusted concerning the loss gradient, while an epoch represents the total number of iterations over all training data in one cycle. The initial learning rate was set to 0.001, and the number of epochs was set to 150. Early stopping was applied to enhance the training efficiency when the segmentation loss did not decrease for approximately 15 epochs; after 100 epochs, the learning rate was reduced to 0.0001.

4.2. Experimental Results

Three models, U-Net, PSPNet, and CSN, were used for training. For CSN, two cases were considered: Case 1 with shared initial two convolutional blocks and Case 2 with shared all encoding blocks for both datasets. Moreover, U-Net and PSPNet were trained only on the SDD dataset, whereas CSN was trained on the SDD as a baseline and then on the ISPRS and UAVid datasets. Figure 6 shows the learning graph of the validation set for each epoch. During training, PSPNet could effectively learn only from the SDD in terms of accuracy and loss on the validation set. When using CSN to train on multiple datasets, the loss on the validation set was relatively higher than U-Net and PSPNet because CSN considered the loss from the SDD and additional dataset during training. Thus, the training duration for CSNs is longer than that for the single-dataset models, leading to comparatively lower accuracy on the validation set.



(**a**) (**b**)

Figure 6. Learning graph of the validation set for each epoch (**a**) accuracy and (**b**) loss.

Table 2 compares segmentation performance for the test set containing various transformed SDD images. It displays the kappa coefficient for all 23 classes of the SDD and the F1 score for 15 selected classes. As CSN was trained with heterogeneous datasets, common objects, such as roads, vegetation, and vehicles, present in both datasets were chosen for segmentation accuracy comparison. The highest accuracy was achieved when training PSPNet with the SDD. Additionally, except for CSN Case 2 (SDD-UAVid), where the SDD and UAVid were trained simultaneously, CSN generally exhibited lower accuracy than U-Net and PSPNet. This is consistent with the results from the validation set. Both datasets have relatively high accuracy for classes such as roads, trees, vegetation, and building roofs. In contrast, the accuracy for classes related to vehicles, such as cars and bicycles, was lower because the bicycle class was not present in the UAVid dataset. Additionally, the car class in the UAVid dataset may have been captured from the side view and has a different shape than that from the SDD. Thus, the combined training of these two datasets does not yield significant benefits for these classes.

Table 2. Comparison of segmentation performance. Kappa represents the kappa coefficient, and the F1 score shows the per-class F1 score.

Model (Training Dataset)	Kappa	F1 Score												
		Paved Area	Dirt	Grass	Gravel	Water	Rocks	Vegetation	Roof	Person	Car	Bicycle	Tree	Bald Tree
U-Net (SDD)	0.63	0.89	0.43	0.84	0.66	0.56	0.32	0.70	0.78	0.52	0.34	0.00	0.34	0.21
PSPNet (SDD)	0.83	0.95	0.70	0.93	0.85	0.91	0.80	0.78	0.93	0.53	0.78	0.25	0.85	0.83
CSN Case 1 (SDD–ISPRS)	0.35	0.72	0.27	0.66	0.17	0.00	0.00	0.35	0.18	0.04	0.00	0.00	0.08	0.00
CSN Case 1 (SDD–UAVid)	0.38	0.76	0.21	0.59	0.44	0.16	0.04	0.22	0.52	0.11	0.01	0.00	0.09	0.03
CSN Case 2 (SDD–ISPRS)	0.45	0.80	0.23	0.63	0.40	0.07	0.03	0.26	0.59	0.21	0.02	0.00	0.51	0.10
CSN Case 2 (SDD–UAVid)	0.72	0.90	0.60	0.85	0.76	0.70	0.31	0.69	0.85	0.62	0.51	0.03	0.58	0.64

Although the accuracy of CSN Case 2 was lower than that of PSPNet, the actual predicted results can reflect real-world situations. Figure 7 shows the prediction results of the test set, PSPNet, and CSN Case 2 trained with the SDD–UAVid datasets.

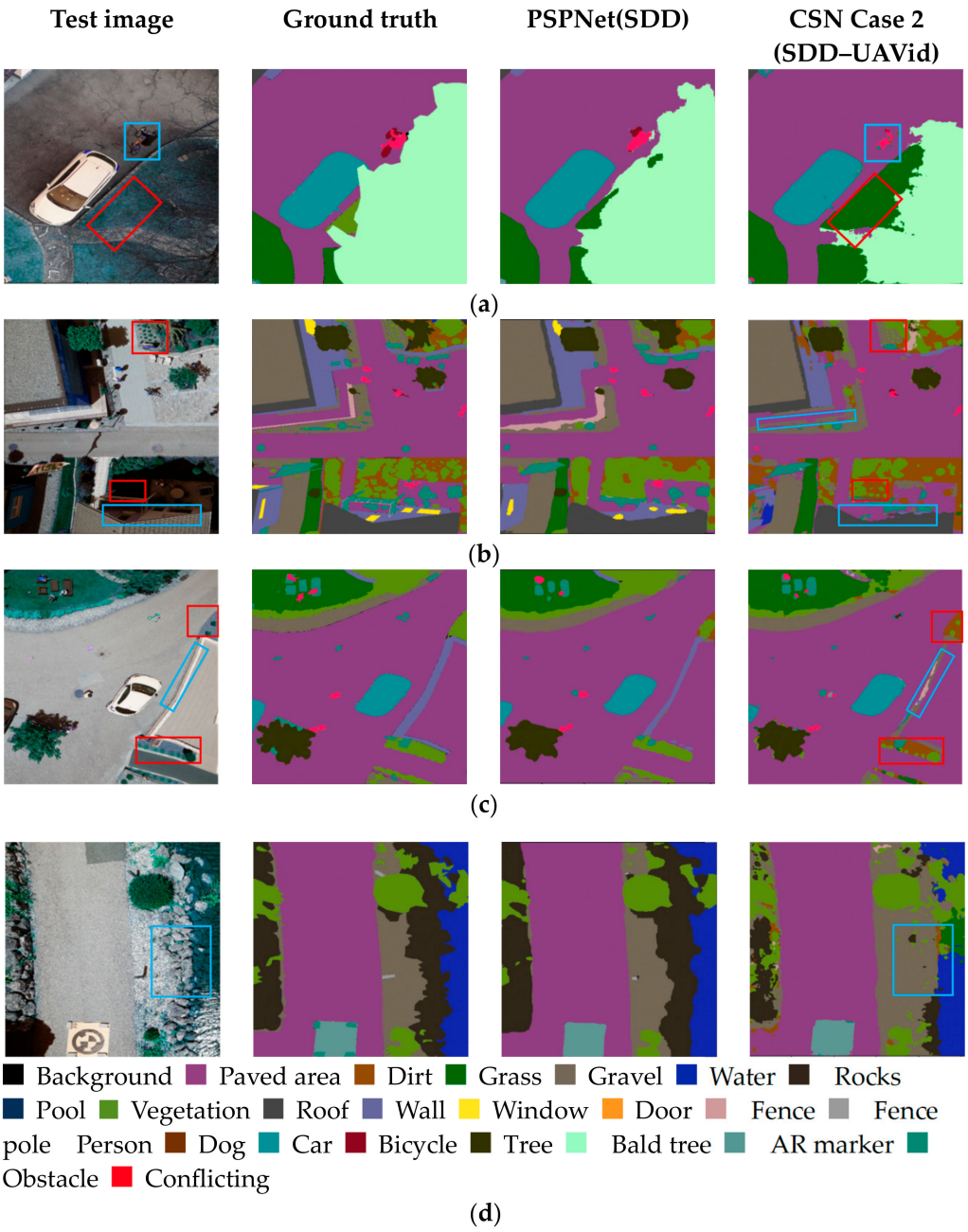


Figure 7. Segmentation results of the PSPNet (SDD) and CSN Case 2 (SDD–UAVid) for four test sets (**a**–**d**). The red boxes represent areas where CSN's prediction closely matches the reality of the input image, while the blue boxes indicate cases where CSN's segmentation results are not better than PSPNet.

Figure 7a shows the images of a car, a human with a bicycle, vegetation, and a bald tree. In the UAV image, vegetation is visible through the branches of the bald tree (the corresponding area is marked with a red box); however, in the ground truth, this area is defined as bald trees. However, in CSN Case 2, the surface is predicted to be vegetation. As the vegetation is visible in the UAV image, predicting it as a vegetation class can be considered a better recognition of the actual surface. However, in the case of a person riding a bicycle marked with a blue box, the person's shape was predicted, but the bicycle class was rarely predicted.

Figure 7b shows the images of buildings with walls and windows, roads, trees, etc. CSN could not recognize features such as windows, walls on buildings, and fences (indi-

cated by a blue box). As these artificial classes were not included in the UAVid dataset, training them using CSN is ineffective. However, for the vegetation class, CSN could predict the crown of plants better than PSPNet (indicated by a red box); this result can also be observed in Figure 7c. The areas marked with the red box are those where the crown of plants is distinct from the ground surface; however, both ground truth and PSPNet recognize this area as vegetation. In contrast, CSN could predict the differentiation between the ground and the crown of plants, but the prediction accuracy for artificial structures, such as fences, was low (indicated by a blue box). Figure 7d shows the images of gravel, rock, and water classes. When gravel and rocks were mixed (indicated by a blue box), CSN had difficulty distinguishing between the gravel and rock classes. Furthermore, in areas where the water level was shallow, and rocks were visible, it was impossible to classify the water properly as rocks or grass.

5. Discussion

In Section 4, the segmentation accuracy was simultaneously compared for U-Net and PSPNet when trained on the SDD and CSNs trained on heterogeneous datasets, such as UAVid and ISPRS. During training, the accuracy of the validation set was the highest for PSPNet. The performance of CSN was lower when it was trained solely on the SDD using U-Net, except when it was trained on the SDD and UAVid dataset simultaneously in CSN Case 2. This is because the weights of CSN considered the loss from both datasets, and it is less efficient in terms of training compared to when only the loss from the SDD was considered. However, when the performance was evaluated using a test set with artificially added transformations to account for the variations in environmental factors and the flight height of UAVs, it was observed that the prediction results more accurately reflected the real-world situation for some classes.

With respect to datasets, training with the UAVid dataset, which was acquired from the UAV platform similar to the SDD, resulted in higher segmentation accuracy compared to training with the ISPRS dataset. Furthermore, the results indicated that the structure of CSN sharing all encoding layers was more effective than that of only initial convolutional layers. Although the two datasets had different class types and were constructed in different regions, they had similar spatial resolution because they were captured at similar flight heights. However, the UAVid dataset contained side-view images, and the objects' shapes and sizes differed from those in the SDD. CSN could not clearly recognize the classes influenced by shapes such as vehicles, buildings, and rocks. However, it could predict the classes influenced by spectral information, such as vegetation and trees, when training with the SDD and UAVid datasets. Thus, in terms of numerical accuracy, training with a single dataset using well-known networks, such as PSPNet, is effective; however, when CSN is trained on heterogeneous UAV datasets, the actual information that was not reflected even in the ground truth map can be better represented, particularly for specific classes.

6. Conclusions

This study explored the challenges and opportunities associated with the semantic segmentation of UAV imagery. The main conclusions include the following: (1) The limited availability of large-scale UAV image datasets is a significant challenge in training accurate segmentation models. (2) The impact of shared layers in the CSN architecture was evaluated, and the results revealed that CSN sharing all encoding layers was more effective than sharing only initial convolutional layers, particularly when dealing with the UAVid–SDD dataset. (3) While some traditional models, such as PSPNet, achieved high segmentation accuracy when trained on a single dataset (SDD), CSN, despite its lower numerical accuracy, had the potential to represent real-world situations better. This was particularly evident in cases wherein classes exhibited shape, size, and spectral characteristics variations. (4) The challenges in UAV semantic segmentation are caused by factors such as varying spatial resolutions, viewing angles, and environmental conditions. Additionally, certain classes, particularly those influenced by shape, may not benefit significantly from training with

multiple datasets. In summary, the segmentation model and training strategy should be carefully considered based on the specific characteristics of the dataset and the classes of interest. While traditional models may excel in high-accuracy scenarios, CSN, with its ability to learn from heterogeneous datasets, holds promise in capturing the variations in real-world situations that may not be adequately represented in ground truth annotations. The results of this study will contribute to the research aimed at improving semantic segmentation in UAV imagery while highlighting the potential for more accurate and robust segmentation in diverse environmental conditions.

Funding: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea NRF, funded by the Ministry of Education (2022R1F1A1063254).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author, Ahram Song, upon reasonable request.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Audebert, N.; Le Saux, B.; Lefèvre, S. Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images. *Remote Sens.* **2017**, *9*, 368. [CrossRef]
2. Xu, P.; Tang, H.; Ge, J.; Feng, L. ESPC_NASUnet: An end-to-end super-resolution semantic segmentation network for mapping buildings from remote sensing images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 5421–5435. [CrossRef]
3. Song, A.; Kim, Y. Semantic Segmentation of Remote-Sensing Imagery Using Heterogeneous Big Data: International Society for Photogrammetry and Remote Sensing Potsdam and Cityscape Datasets. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 601. [CrossRef]
4. Neupane, B.; Horanont, T.; Aryal, J. Deep Learning-Based Semantic Segmentation of Urban Features in Satellite Images: A Review and Meta-Analysis. *Remote Sens.* **2021**, *13*, 808. [CrossRef]
5. Lateef, F.; Ruichek, Y. Survey on semantic segmentation using deep learning techniques. *Neurocomputing* **2019**, *338*, 321–348. [CrossRef]
6. Yuan, X.; Shi, J.; Gu, L. A review of deep learning methods for semantic segmentation of remote sensing imagery. *Expert Syst. Appl.* **2021**, *169*, 114417. [CrossRef]
7. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
8. Azad, R.; Aghdam, E.K.; Rauland, A.; Jia, Y.; Avval, A.H.; Bozorgpour, A.; Karimijafarbigloo, S.; Cohen, J.P.; Adeli, E.; Merhof, D. Medical image segmentation review: The success of u-net. *arXiv* **2022**, arXiv:2211.14830.
9. Diakogiannis, F.I.; Waldner, F.; Caccetta, P.; Wu, C. ResUNet-a: A DL framework for semantic seg-mentation of remotely sensed data. *ISPRS J. Photogramm. Remote Sens.* **2020**, *162*, 94–114. [CrossRef]
10. Li, H.; Qiu, K.; Chen, L.; Mei, X.; Hong, L.; Tao, C. SCAttNet: Semantic Segmentation Network with Spatial and Channel Attention Mechanism for High-Resolution Remote Sensing Images. *IEEE Geosci. Remote Sens. Lett.* **2020**, *18*, 905–909. [CrossRef]
11. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
12. 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 (ECCV), Munich, Germany, 8–14 September 2018; pp. 801–818.
13. Panboonyuen, T.; Jitkajornwanich, K.; Lawawirojwong, S.; Srestasathiern, P.; Vateekul, P. Semantic Segmentation on Remotely Sensed Images Using an Enhanced Global Convolutional Network with Channel Attention and Domain Specific Transfer Learning. *Remote Sens.* **2019**, *11*, 83. [CrossRef]
14. Cui, B.; Chen, X.; Lu, Y. Semantic Segmentation of Remote Sensing Images Using Transfer Learning and Deep Convolutional Neural Network With Dense Connection. *IEEE Access* **2020**, *8*, 116744–116755. [CrossRef]
15. Gerke, M. Use of the Stair Vision Library within the ISPRS 2D Semantic Labeling Benchmark (Vaihingen). 2014. Available online: https://www.researchgate.net/publication/270104226_Use_of_the_Stair_Vision_Library_within_the_ISPRS_2D_Semantic_Labeling_Benchmark_Vaihingen (accessed on 12 September 2023).
16. Meletis, P.; Dubbelman, G. Training of convolutional networks on multiple heterogeneous datasets for street scene semantic segmentation. In Proceedings of the IEEE Intelligent Vehicles Symposium, Changshu, Suzhou, China, 26–30 June 2018; pp. 1045–1050.

17. Ghassemi, S.; Fiandrotti, A.; Francini, G.; Magli, E. Learning and Adapting Robust Features for Satellite Image Segmentation on Heterogeneous Data Sets. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 6517–6529. [CrossRef]
18. Cordts, M.; Omran, M.; Ramos, S.; Rehfeld, T.; Enzweiler, M.; Benenson, R.; Franke, U.; Roth, S.; Schiele, B. The Cityscapes Dataset for Semantic Urban Scene Understanding. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 3213–3223.
19. Lee, H.; Eum, S.; Kwon, H. Cross-Domain CNN for Hyperspectral Image Classification. In Proceedings of the IGARSS 2018—2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 3627–3630.
20. Lyu, Y.; Vosselman, G.; Xia, G.-S.; Yilmaz, A.; Yang, M.Y. UAVid: A semantic segmentation dataset for UAV imagery. *ISPRS J. Photogramm. Remote Sens.* **2020**, *165*, 108–119. [CrossRef]
21. Semantic Drone Dataset. Available online: http://dronedataset.icg.tugraz.ati (accessed on 12 September 2023).
22. 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.