International Journal of Advanced Engineering Research and Science (IJAERS)
Peer-Reviewed Journal
ISSN: 2349-6495(P) | 2456-1908(O)
Vol-8, Issue-8; Aug, 2021
Journal Home Page Available: https://ijaers.com/
Article DOI: https://dx.doi.org/10.22161/ijaers.88.29

A Review of Remote Sensing Applications on Very High-Resolution Imagery Using Deep Learning-Based Semantic Segmentation Techniques

Philipe Borba[1,2], Edilson de Souza Bias[2], Nilton Correia da Silva[3], Henrique Llacer Roig[2]

[1]Brazilian Army Geographic Service, Brazil
[2]Geosciences Institute, University of Brasília, Brazil
[3]Campus Gama, University of Brasília, Brazil

Received: 07 Jul 2021,

Received in revised form: 05 Aug 2021,

Accepted: 12 Aug 2021,

Available online: 22 Aug 2021

***Keywords*—Remote Sensing, Deep Learning, Semantic Segmentation, Convolutional Neural Networks, State-of-the-art, Review.**

***Abstract*—Semantic Segmentation is a technique in Computer Sciences (CS) to extract information from images. Recent advances in Artificial Intelligence, particularly in Deep Learning, Semantic Segmentation combined with techniques such as convolutional neural networks, have presented better results and exciting results. Due to its power and better results than classical approaches, there has been an increase in research articles in Remote Sensing that propose using deep learning-based semantic Segmentation to extract information from satellite or airborne imagery. In this paper, we surveyed the state-of-the-art of Semantic Segmentation in Remote Sensing from 2010 until 2020 by identifying the research topics and the number of publications and citations. Furthermore, we also pointed out the fundamental algorithms, the main convolutional neural network architectures, backbones, and the most used evaluation metrics. In addition, some datasets were highlighted, as well as some frameworks that can be used to train semantic segmentation deep neural networks. Finally, we have shown some applications of the showcased techniques and concluded the paper by pointing out some research opportunities of Remote Sensing Semantic Segmentation, concerning some bleeding-edge scientific papers published in 2020 in CS.*

I. INTRODUCTION

The extraction of information from remote sensing images has been an active research field, with essential applications for urban planning, urban dynamics modeling, and disaster damage assessment. Semantic Segmentation is the process of assigning a label to each pixel of an image and decompose a scene into semantically meaningful regions [1]. Traditionally, semantic Segmentation is performed either pixel-wise or with object-based approaches. The latter is known as Geographic Object-Based Image Analysis (GEOBIA) [2] and usually outperforms the former. These approaches typically consist of two separate steps: Segmentation followed by classification. Because the second step's accuracy usually relies on the first step's quality, image segmentation is critical for GEOBIA.

However, image segmentation is not a trivial task, given that most algorithms rely on subjective and arbitrary parameters setting. The incorrect choice of parameters may lead to undesired results, such as under-segmentation and over-segmentation, which may impact the classification accuracy. Moreover, segmentation techniques' generalization capability is limited because they cannot deal with the objects' complexity present in an image. For example, a given set of parameters can provide good segmentation results at homogeneous regions (e.g., agricultural fields) and unsatisfactory results in heterogeneous areas like urban environments.

Thus, image analysts usually try several parameter combinations to achieve a suitable outcome for an entire scene, a time-consuming task. Adaptive segmentation algorithms were proposed to deal with the diversity of image objects [3, 4] or automatic tuning of segmentation parameters [5, 6]. However, these methods are complex, rely on human-made reference images, and are designed for specific applications.

Recently, improvements in computation power and parallel processing algorithms using graphics processing units (GPUs) favored the development of deep learning (DL) [7, 8], particularly convolutional neural networks (CNNs), a type of DL method introduced by [9], have become exceedingly popular for classification, object localization, and semantic segmentation of remote sensing images [10]. CNNs are designed to automatically extract spatial patterns (e.g., shapes, edges, texture) of images using a set of convolutions and pooling operations, hence learning object-specific characteristics in an end-to-end fashion.

Particularly in the context of semantic Segmentation, neural networks have achieved outstanding results [11, 12, 13, 14, 15, 16, 17, 18]. Unlike traditional pixel-wise classification, semantic Segmentation using CNNs can preserve the object boundaries producing sharp, fine-scale Segmentation. Fully convolutional networks (FCNs) were the first approach that employed deep networks for semantic Segmentation. The rationale behind FCNs relies on transforming the fully connected layers into upsampling or transposed convolutional layers [19] to perform dense pixel predictions. The pioneering work of [19] adapted well-known CNNs models such as AlexNet for semantic segmentation tasks.

In semantic Segmentation, the smallest segment can be a single pixel, which is not adequate for most applications of information extraction using high-resolution remote sensing images because, in these images, it is improbable to find a target with the dimensions of a single pixel. To overcome this problem, instance segmentation combined object detection and semantic segmentation can be used to classify an object at the pixel level and outline its exact shape [20]. Both semantic Segmentation and instance segmentation networks provide the opportunity to simultaneously detect and classify building footprints without the need for a previous segmentation step, thus vanquishing the limitations of GEOBIA.

This paper will cover the latest state-of-the-art (SOTA) of semantic Segmentation in very high-resolution remote sensing, focusing only on methods that use convolutional neural networks (CNNs). We also want to identify research opportunities in RS by briefly analyzing the latest trends on CS. To fulfill this goal, this review is organized as follows: in section 2, we show the SOTA of semantic Segmentation in RS and CS papers; in section 3, we cover the basic concepts of DL and semantic segmentation techniques, the primary neural network architectures, the available datasets and frameworks and finally some raster to vector methods; and in section 4 we sum up the concepts presented in this paper, as well as cover the opportunities of research in geosciences based on the comparison of the SOTA semantic segmentation methods.

II. LITERATURE REVIEW

We conducted a literature review on remote sensing to identify the most relevant deep learning techniques and methods employed to extract information from remote sensing imagery, presented in section 2.1.

Moreover, to identify possible new techniques from computer sciences, we carried out a brief literature survey on review articles and also pointed out the best results on popular benchmarks showcased on Papers With Code [21], shown in section 2.2.

2.1. Literature Review on Remote Sensing

To perform our literature review, we searched the knowledge database SciELO Citation Index (Web of Science) to investigate further what are the main research topics, the number of publications per year, and the most cited papers. This information was used to try to delineate the most relevant papers so that we could further analyze them so that we could extract more helpful information, such as the most popular methods employed.

The term” Semantic Segmentation” was searched using the time range 2010-2020 as the filter, and there were 10,145 results, then were filtered once more, considering only the” Remote Sensing” field, yielding 718 results. To identify the main research topics, we built a word cloud, shown in figure 1, with the keywords of these results. Analyzing the picture, we can infer that the research conducted from 2010 until 2020 has used neural networks, particularly convolutional neural networks (CNNs), to extract or identify features using high-resolution satellite or aerial imagery. Common ground features extracted by the considered papers are roads and buildings.

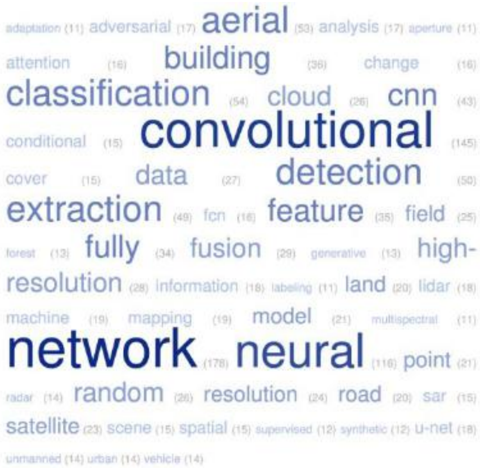

Fig. 1: Word cloud built with the keywords of the results of the search Semantic Segmentation on the Web of Science database, from 2010 to 2020, considering only papers in Remote Sensing. Larger words mean more recurring terms in the research papers' keywords.

During the considered time range, there has been a nearly exponential growth in the number of papers in remote sensing that covers semantic Segmentation that can be visualized in figure 2. The years 2015 and 2016 have presented a slight increase in the number of publications that might be a consequence of the papers published in CS, such as [22, 23, 24]. From 2017 until 2019, there has been a significant increase in the number of research papers, peaking at 140 in 2019. Since 2020 is not over yet, we can expect an even more substantial number than 2019, since the number of research papers published in 2020 is much higher than 2018's and only 40% smaller than2019's.

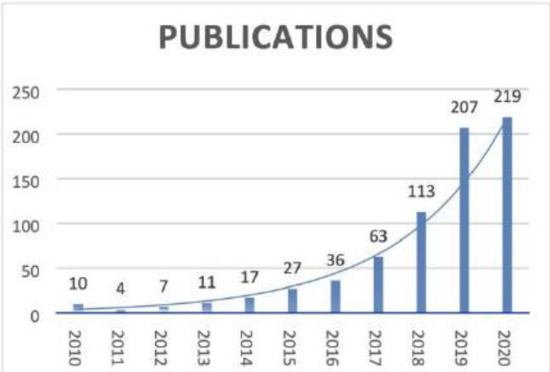

Fig. 2: Number of publications in Remote Sensing with the subject Semantic Segmentation from 2010 to 2020 registered on Web of Science.

We further narrowed our chosen papers by cross-referencing our search results with data from a GitHub repository (https://github.com/thho/DLinEO_review), which is under the license CC-BY-4.0 and contains data used in [1, 25]. Using this info, we have only considered semantic Segmentation, resulting in 261 papers to analyze. Then, we built the graph in figure 3 to find out the most popular architecture. We concluded that the most famous architecture in RS papers is the U-Net, followed by custom architectures and then Fully Convolutional Networks (FCNs).

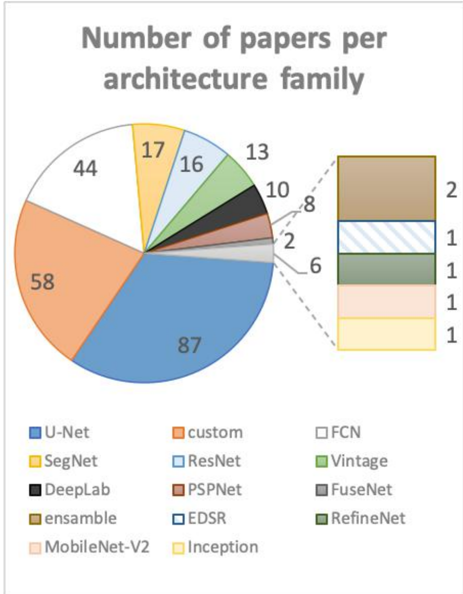

Fig. 3: Papers grouped by architecture family.

Then, to evaluate the backbone usage, we built a word cloud shown in figure 4 to find out the most popular backbones, and we found out that ResNets, VGG-16, and the Inception series are very popular.

Fig. 4: Family architectures used in Semantic Segmentation papers in Remote Sensing in the considered papers. Larger names represent more popular family architecture.

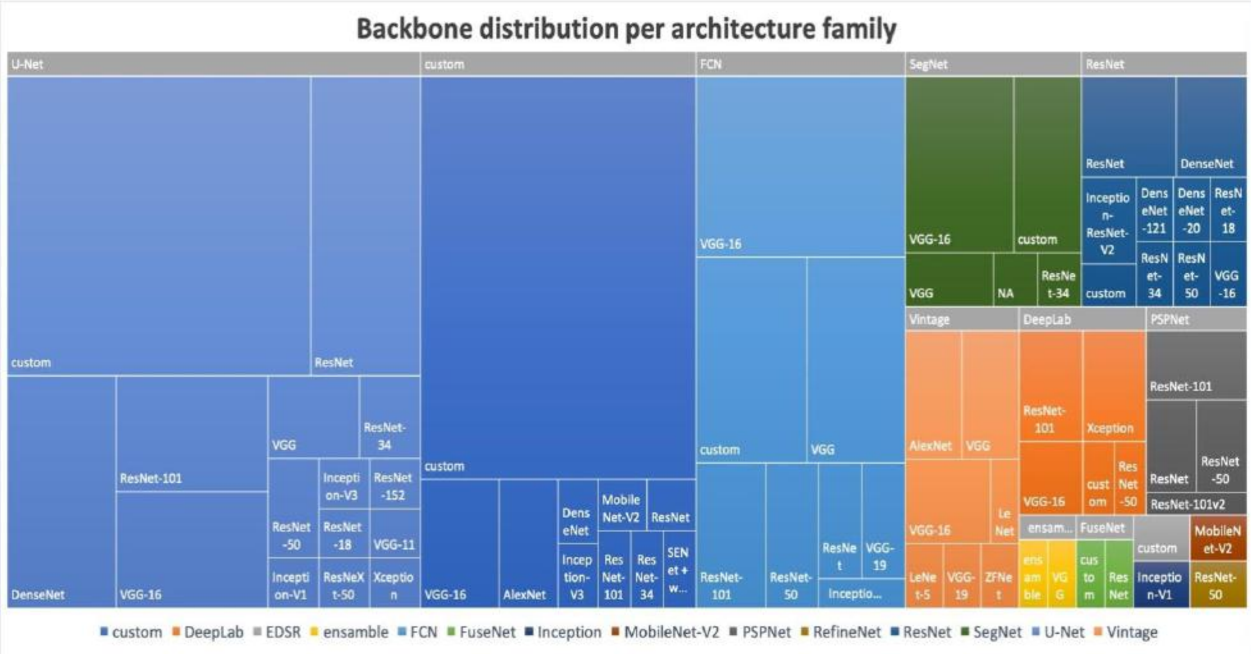

Fig 5: Tree Map representing the backbone distribution for each type of convolutional neural network architecture used in the considered papers.

To understand the relationship between the backbones and the architectures chosen in each paper and presented in the data here analyzed, we built a tree map shown in figure 5, which leads us to conclude that U-Nets with custom and ResNet backbones are very popular, followed by custom backbone and custom architecture, then by VGG-16 backbone with FCN architecture, and finally, VGG-16 backbone with SegNet architecture.

2.2. Brief Literature Review on Computer Science

There are several review articles in Computer Sciences [26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42] that portray the evolution of deep learning-based semantic segmentation methods. Common research fields on CS that use the mentioned techniques are research on self-driving vehicles [43, 44], pedestrian detection [45, 46] and computer aided diagnosis using medical images [47, 48].

The surveyed papers cover similar architectures and backbones already listed on 2.1. The novel backbones that were not identified in section 2.1 are the ones from the EfficientNet family, ResNeSt [49], and SE-ResNet family [50]. The training datasets used in CS applications are one of the main differences from RS studies. As examples of common datasets used in CS, we can cite the Cityscapes dataset [51], the PASCAL VOC (PASCAL Visual Object Classes Challenge) [52], and its extension, the PASCAL Context [39].

There is a platform called Papers With Code [21] that gathers results of several papers, as well as codes that are available online to reproduce such study considered papers. On this website, the results of each benchmark are ranked, and the best models are presented. Some of the models with the best results on the previously mentioned datasets are shown in table 1:

Table 1: Best models on some available datasets, according to Papers With Code [21].

Dataset	Best Model	Paper Title	mIoU
Cityscapes test	HRNet-OCR	Hierarchical MultiScale Attention for Semantic Segmentation [53]	85.1%
PASCAL VOC 2012 test	EfficientNet-L2+NAS-FPN	Rethinking Pretraining and Self-training [54]	90.5%
PASCAL Context	Channelized Axial Attention (CAA) with Simple decoder (Efficientnet-B7)	Channelized Axial Attention for Semantic Segmentation [55]	60.5%
Cityscapes val	HRNetV2-OCR+PSA	Polarized SelfAttention: Towards High-quality Pixelwise Regression [56]	86.95%

Other worth mentioning techniques found on the cited review papers and the research shown in table 1 are self-training [57], Channelized Axial Attention [55], and

Polarized Self-Attention [56].

III. MAIN CONCEPTS AND METHODOLOGIES IN SEMANTIC SEGMENTATION

From the SOTA review carried out in section 2, we identified some of the main concepts and techniques that we need to understand when studying semantic segmentation techniques applied to remote sensing.

Furthermore, considering the selected papers and regarding the ideas highlighted in the SOTA review, we will present some basic concepts in section 3.1, some training improving techniques in section 3.2, the main convolutional neural network backbones in section 3.3, the main architectures on section 3.4, some applications on RS and examples of some available datasets on section 3.5, and finally, some frameworks and tools on section 3.6.

3.1. Main Concepts of Convolutional Neural Networks

The convolution layer is one of the building blocks of Deep Learning. It can be defined as a combination of linear and nonlinear operations such as convolution and activation functions [58].

Convolution is a mathematical operation that applies an array of numbers (kernel) to the input, enabling feature extraction operations [58]. On the other hand, the activation function is a mathematical resource to introduce nonlinearities in the convolutional neural networks. Some examples of them are the sigmoid function, the hyperbolic tangent function, the rectified linear unit (ReLU) [58], the leaky rectified linear unit (Leaky ReLU) [59], the exponential linear unit (ELU) [60], the scaled exponential linear unit (SELU) [61], the gaussian error linear unit (GELU) [62], the Mish [63] and the Softmax [64]. Their mathematical definitions can be seen, respectively, on equations 1, 2, 3, 4, 5, 6, 7, 8, and 9. It is worth mentioning that Softmax is often used as an output function on convolutional neural networks.

$$sigmoid(x) = \frac{1}{1+e^{-x}} \tag{1}$$

$$tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2}$$

$$ReLU(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{3}$$

$$Leaky\_ReLU(x) = \begin{cases} 0.01x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{4}$$

$$ELU(x) = \begin{cases} \alpha(e^x - 1) & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases} \tag{5}$$

$$SELU(x) = 1.597 \begin{cases} 1.67326(e^x - 1) & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \tag{6}$$

$$GELU(x) = 0.5x\left(1 + tanh\left(\sqrt{\frac{2}{\pi}}\left(x + 0.044715x^3\right)\right)\right) \tag{7}$$

$$Mish(x) = x \cdot ln(1 + e^x) \tag{8}$$

$$\text{Softmax}(x_i) = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \tag{9}$$

The difference between filters that use convolutions (common in image processing tasks) and the convolutional layers of CNNs is that, instead of applying a pre-determined kernel to the input, it learns the best parameters of the kernel to extract features due to the training process [33, 39, 34].

Another critical concept in CNN theory is the pooling layer, which replaces a small neighborhood of a feature map with some statistical information, such as mean or max [39]. This process is vital because it sub-samples images, reducing the dimensionality of the feature maps by introducing a translation invariance to small shifts and distortions and decreasing the number of learnable parameters [58].

The combination of convolutional layers, activation functions, and pooling operations is usually called Convolutional Backbone, and its role is to extract high-level features [1].

Usually, a CNN used to classify an image is composed of input, the convolutional backbone, and a classifier head. This last one is typically composed of fully connected artificial neural networks (ANN), which have several perceptrons connected among each other.

The process of finding the best weights of the neural network has two steps: a forward stage and a backward stage [27]. According to [27], the first step uses the current weights and biases of the network to process the input and calculate a prediction. Then this prediction is compared to the expected output (ground truth) with a function called loss. After determining the loss, the gradients of each parameter are updated in the backward stage using the chain rule, a method called backpropagation [9].

The objective of the training process is to minimize the loss function, which means that the outputs of the trained neural networks are similar to the ground truth. To carry out the training, the weights of the neural network need to be initialized, and the way they are set can impact the training time.

According to [65], two popular initialization methods are Glorot (a.k.a. Xavier initialization) [66] and He (a.k.a. Kaiming initialization) [67]: the first has as its primary goal achieve faster convergence and better accuracy by scaling the neural network weights so that the variance of the input is equal to the conflict of the output [65]; the second aims to achieve depth independent performance by modifying the scaling factor to account rectifier nonlinearities [65]. The weights of a neural network can also be initialized from a previously trained network, a technique that is known as transfer learning. [68] defines four types of transfer learning: instance-based, mapping-based, network-based, and adversarial-based.

To achieve convergence faster during the training process, some algorithms with adaptative learning rates can be used. In neural networks studies, these algorithms are usually gradient-based and are called optimizers [69]. Some examples of them are Stochastic Gradient Descend (SGD) [70], AdaGrad [71], Nesterov Accelerated Gradient (NAG) [72], Adaptative Moment Estimation (Adam) [73], Rectified Adam (RAdam) [74], Adaptative and Momental Bound (AdaMod) [75] and Adaptative Second Order (AdaHessian) [76].

Regarding loss functions, [77] summarizes some of the available ones that are usually chosen for semantic segmentation tasks. Among those, it is worth mentioning the ones that are commonly used in semantic segmentation papers: the Cross-Entropy (CE) [78], the Weighted Cross-Entropy (WCE) [79], the Dice [80], the IoU/Jaccard [81], the Tversky [82] and the Focal Tversky [83]. The mathematical formulation of each cited loss function is described respectively in the equations 10, 11, 12, 13, 14, and 15, where N is the number of pixels, g_i^c is the binary indicator of whether the class label c is correctly classified for pixel i, $s^c{}_i$ is the corresponding predicted probability, α and β are hyperparameters used to control the balance between false positives and false negatives, and γ is a coefficient in the interval [1,3].

Some metrics can be used to evaluate the quality of the trained neural networks. According to [84], overall accuracy (OA), precision, recall, and the F_1 index are helpful for evaluating the quality of the training, and they are defined by the following equations:

$$OA = \frac{TP + TN}{FP + FN} \tag{16}$$

$$precision = \frac{TP}{TP + FP} \tag{17}$$

$$recall = \frac{TP}{TP + FN} \tag{18}$$

$$F_1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{19}$$

where TP, TN, FP, and FN are, respectively, the true positives, the true negatives, the false positives, and the false negatives.

According to [31], the Jaccard Index, also known as intersection over union (IoU), can be defined by:

$$IoU = J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{20}$$

where A e B are, respectively, the ground truth and the predicted data.

$$L_{CE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} g_i^c \log s_i^c \tag{10}$$

$$L_{WCE} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} w_c g_i^c \log s_i^c \tag{11}$$

$$L_{Dice} = 1 - \frac{2\sum_{i=1}^{N}\sum_{c=1}^{C} g_i^c s_i^c}{\sum_{i=1}^{N}\sum_{c=1}^{C} g_i^{c2} + \sum_{i=1}^{N}\sum_{c=1}^{C} s_i^{c2}} \tag{12}$$

$$L_{IoU} = 1 - \frac{\sum_{i=1}^{N}\sum_{c=1}^{C} g_i^c s_i^c}{\sum_{i=1}^{N}\sum_{c=1}^{C} \left(g_i^c + s_i^c - g_i^c s_i^c\right)} \tag{13}$$

$$L_{Tversky} = \frac{\sum_{i=1}^{N}\sum_{c=1}^{C} g_i^c s_i^c}{\sum_{i=1}^{N}\sum_{c=1}^{C} \left(g_i^c s_i^c\right) + \alpha\sum_{i=1}^{N}\sum_{c=1} \left(1 - g_i^c\right) s_i^c + \beta\sum_{i=1}^{N}\sum_{c=1} g_i^c \left(1 - s_i^c\right)} \tag{14}$$

$$L_{FT} = \left(1 - L_{Tversky}\right)^{\frac{1}{\gamma}} \tag{15}$$

Also, according to [31], the mean intersection over union index (mIoU) can be defined by:

$$mIoU = \frac{1}{m} \sum \frac{A_{pred} \cap A_{true}}{A_{pred} \cup A_{true}} \quad (21)$$

where m is the number of expected classes, A_{pred} is the prediction set, and A_{true} is the ground truth set.

3.2. Convolutional Neural Networks Training Improving Techniques

Convolutional Neural Networks usually take a long time to train, even when using a GPU. This occurs due to the fact of the large number of weights that have to be adjusted in the process of backpropagation: the larger the number of parameters of the model, the longer it will take to train. This can be overcome using distributed training on several GPUs and increasing the batch size.

In addition, the time spent on the training process also depends on the number of samples that the training dataset has. On the one hand, if there are not enough images on the training dataset, the neural network will not" see" a significant number of patterns to learn and perform poorly on the training dataset. This below-average learning is known as underfitting. On the other hand, if the number of images is not high enough, the neural network can memorize the data and perform well on the training dataset, but poorly on the test dataset, known as overfit [64, 85].

Moreover, the performance on test datasets can be improved by using regularization techniques, which are defined by [64] as any modification made to a learning algorithm that is intended to reduce its generalization error but not its training error. Some examples of regularization techniques are weight decay, label smoothing, early stopping, dropout, batch normalization, and data augmentation. Each of these is described below:

- Weight decay (a.k.a. L2 Regularization) is a method that modifies the weights of a neural network in such a way that the loss to be minimized is added a penalty of the L_2 norm of the weights [64].

- Label smoothing [86, 64] is a technique that adds noise to the label, mitigating the effect of some incorrect label that the dataset may have. It also has the advantage of preventing the pursuit of hard probabilities without discouraging correct classification [64].

- Early stopping consists of stopping the training when the neural network stops learning, in other words, when the validation metrics stop improving [64].

- Dropout [87] is a technique used to reduce the dependency of some neurons on neural networks. At each training step, it is calculated a probability of the neuron to be shut down, and if it is larger than the set threshold, this element is turned off (outputs zero). This has a regularizing effect since it forces the network to learn patterns with other connected neurons.

- Batch Normalization [88] is a model reparameterization technique that introduces both additive and multiplicative noise on the hidden units at training time by normalizing the inputs to outputs with zero mean and unit variance [64].

- Data augmentation is a technique that uses image manipulation to create new training samples [64, 89]. Common data augmentation operations are random crop, random flip, and random color jitters. Furthermore, a novel data augmentation technique that has been recently employed in CS papers is Mixup [90], which consists of building synthetic images composed of a weighted sum of random pairs of the training data. According to [64, 89], data augmentation also has a regularizing effect, and it may contribute to avoid overfitting. One step further on data augmentation is using self-supervised techniques to learn from data the augmentation procedures that can achieve better metrics. As examples of such methods, we can cite AutoAugment [91], Faster AutoAugment [92], and RandAugment [93].

Furthermore, there is another approach to training optimization, which is the usage of Learning Rate Scheduling [94]. This technique changes the value of the learning rate according to some heuristic to try to improve the neural network accuracy and reduce training time [95, 96]. Some examples are Time Based Exponential Decay [97], Exponential Decay [98], Linear Warmup, Cosine Annealing [96], Cosine Power Annealing [99], and One-Cycle Learning Rate Scheduling Policy [100].

Finally, the last training improving technique that we will cover is Stochastic Weight Averaging (SWA) [101, 102], which is a procedure used to optimize the neural network that averages multiple points along the trajectory of Stochastic Gradient Descent (SGD), with specific learning rate procedures, that can be either cyclical or constant. The usage of this technique can help the optimizer to find a better optimization landscape, which might lead to better optimization results.

3.3. Main Convolutional Neural Network Backbones used on Semantic Segmentation Tasks

In this subsection, we will briefly present the key ideas regarding the main convolutional neural networks used to perform semantic segmentation tasks in RS. From our bibliographic research carried out in 2.1, we analyzed the results shown in figures 3 and 5, and then we identified key backbones to be explained in this section. The chosen backbones were AlexNet [22], ZFNet [23], GoogLeNet [24], VGG-19 [24], the ResNet family [103], Inception [86, 104], XCeption [105] and MobileNet [106, 107, 108]. From the bibliographic research done in Computer Sciences, we came across the following worth mentioning backbones: ResNeXt, ResNeSt, and EfficientNet.

According to [1, 109], convolutional neural networks (CNNs) were introduced by [9] and in 2012, [110] used them in a model called AlexNet to win the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [22]. According to [8], in 2013 and 2014, ILSVRC were also won by CNNs, with models respectively called ZFNet [23], GoogLeNet [24]. [1] define the architectures AlexNet [110], ZFNet [23] and VGG-19 [24] as Vintage Architectures.

In 2015, the family of architectures called ResNets [103] introduced skip connections to address the vanishing/exploding gradient [66, 111], which prevented deep neural networks from having a large number of layers. Due to this idea, deeper models were possible, and then the 2015's ILSVRC was won by a ResNet-152. The ResNet family has the ResNet blocks as its basic building blocks, a series of convolutions and activations stacked. There is a concatenation operation by the end of the block (also called skip connections) to preserve some of the input information.

To further push the boundary regarding the performance of the ResNet family-based algorithms, [86, 104] developed a family of architectures called Inception, which has as its basic block the inception block. Different from ResNet blocks that only concatenate the input of the block with the output, the inception block has several outputs: each output is the result of a different stacking of convolutions and pooling operations. Further advances on such idea were also proposed by the XCeption family [105] and the MobileNet family [106].

Thus, [112] evolved the idea of the Inception Block by proposing a backbone called ResNeXt: in this method, a cardinality value to the blocks is proposed, which widens the block with more branches of stacked convolutions, enabling further representation learning. Other backbone architectures that are worth mentioning are the SE-ResNet [50] and the ResNeSt [49]. The first method proposes the usage of an attention mechanism at the beginning and the end of the ResNet block, composing the Squeeze and Excite block, which performs dynamic channel-wise feature recalibration, to improve the representational power of the network. The latter method proposes the usage of Split-Attention Block, which adds the same idea of cardinality to the SE-Net-Block proposed by [50].

Recently there have been some breakthrough architectures using Neural Architecture Search (NAS) [113, 114, 115], which is a reinforcement learning technique to find out the best architecture to perform tasks on object detection and semantic segmentation [1]. Using NAS techniques, in late 2019, researchers at Google have created a series of backbones called EfficientNet [116]. In 2020, another group from Google had published a paper called EfficientDet: Scalable and Efficient Object Detection [117], in which they improved EfficientNets and proposed a weighted bi-directional feature pyramid network (BiFPN). According to [117], with these improvements, the research team achieved 4x smaller networks that used 13x fewer FLOPs, with a gain of 0.2% of mean average precision (mAP) of state-of-the-art mAP on the COCO dataset.

3.4. Main Convolutional Neural Network Architectures Used on Semantic Segmentation Tasks

In neural network applications, the convolutional backbone is often combined with other structures depending on the task that we want to perform. It can be used with a design such as fully convolutional layers to perform classification. In the case of semantic Segmentation, there are some approaches, as using naïve encoders and encoder-decoder structures [1]. There are also Generative Adversarial Networks (GAN) [39, 118, 119] and Recurrent Neural Networks (RNNs) with Long Short-Term Memory (LSTM) [30] approaches to perform semantic segmentation tasks, but we will not cover those techniques in this paper. More information on those techniques can be found on [1, 30, 42].

Naïve decoders normally use a convolutional backbone and trained deconvolutional layers to perform the upsampling task to generate the segmentation mask, combined with some interpolation method such as bilinear. Some examples of this type of architecture are Fully Convolutional Networks (FCN) [120], DeepLabV1 [121], DeepLabV2 [122], ParseNet [123], PSPNet [124] and DeepLabV3 [125].

Encoder-decoder models, in contrast to naïve decoder, instead of using an interpolation method to upsample the feature maps, use a more complex decoder, with shortcuts or skip connections to maintain information from the encoder to the decoder and gradually perform the upsampling [1]. Some examples of this type of model are the DeconvNet [126], the SegNet [127], the U-Net [79], the U-Net++ [128], the DoubleU-Net [129], the MultiResUNet [130], the RefineNet [131] and the DeepLabV3+ [132]. The architecture of an encoder-decoder architecture called U-Net is shown in figure 6.

A novel type of encoder-decoder architecture is the HRNet (or High-Resolution Net) [133] and the HR-Net OCR[53], both of which are featured on top positions of the Cityscapes benchmark, as shown in table 1. This method aims to maintain high-resolution images at every stage of the process by combining different parallel chains of convolutions and strided convolutions. Object-Contextual Representations (OCR) is an attention mechanism [134] that considers the context of the considered pixel instead of it alone. OCR can be combined with different backbones such as ResNet-101 and Xception and different architectures such as DeepLabV3+ to improve segmentation results, as shown by [135]. When OCR is combined with HR-Net, we have the HR-Net OCR architecture.

Another type of attention mechanism that can be combined with HR-Net is the Polarized Self-Attention (PSA) [56], which has two main operations in its design: the polarized filtering and enhancement component. This type of attention mechanism not only looks at spatial features but also channel representations.

Finally, another worth mentioning set of techniques is the usage of EfficientNet backbones with Feature Pyramid Networks (FPN), combined with self-training techniques such as noisy student, which is a semi-supervised learning technique that improves the training results [57]. Table 1 shows that the best method on PASCAL VOC 2012 test dataset is the usage of EfficientNet trained with noisy student technique (a.k.a. EfficientNet-L2) with FPN architecture and Neural Architecture Search (NAS) [54]. On the other hand, the best model on PASCAL Context is the combination of a plain EfficientNet-B7 with an attention mechanism called Channelized Axial Attention (CAA) [55].

3.5. Applications on Remote Sensing and Examples of Available Datasets

Deep Learning (DL) plays an important role in nowadays science is particularly geosciences. There are several RS research papers such as [136], [137], and [138] that compare classical computer vision techniques to DL techniques, and they show that DL can achieve better accuracies.

DL-based techniques can solve several problems in Geosciences. Among those problems we can cite object detection [139, 140], hyperspectral image classification [10, 141], super-resolution [142, 143, 144], change detection [145, 146] and semantic segmentation.

Regarding Semantic Segmentation [84, 147, 148, 149], there are some use cases, such as building footprint extraction [11, 12, 150, 13, 14, 15, 16, 17, 18], road extraction [151, 152, 153] and land use and land cover (LULC) analysis [154, 155].

To train neural networks that can solve LULC problems, data from the ISPRS Potsdam and Vaihingen [156, 157] can be used. This is a dataset with airborne photogrammetric imagery of Potsdam, covering six classes (impervious surfaces, building, low vegetation, tree, car, and clutter/background).

Moreover, to perform training of deep convolutional neural networks that can extract building footprints, some of the open datasets available online are listed below, and the details are shown in table 2:

- SpaceNet [158, 159]: dataset with satellite imagery of the following cities: Rio de Janeiro, Las Vegas, Paris, Khartoum, and Shanghai.
- Massachusetts [160]: dataset with satellite imagery of the city of Boston.
- WHU building [161]: dataset with airborne photogrammetric imagery of New Zealand.
- INRIA aerial [162]: dataset with satellite imagery from the following cities: Austin, Chicago, Kitsap County, Western Tyrol, and Vienna.
- LandCover.ai [163]: dataset with satellite imagery of Poland.
- AIRS [164]: dataset with satellite imagery of Christchurch City in New Zealand.
- CrowdAI [165]: a simplified version of the SpaceNet Dataset, with only RGB images.

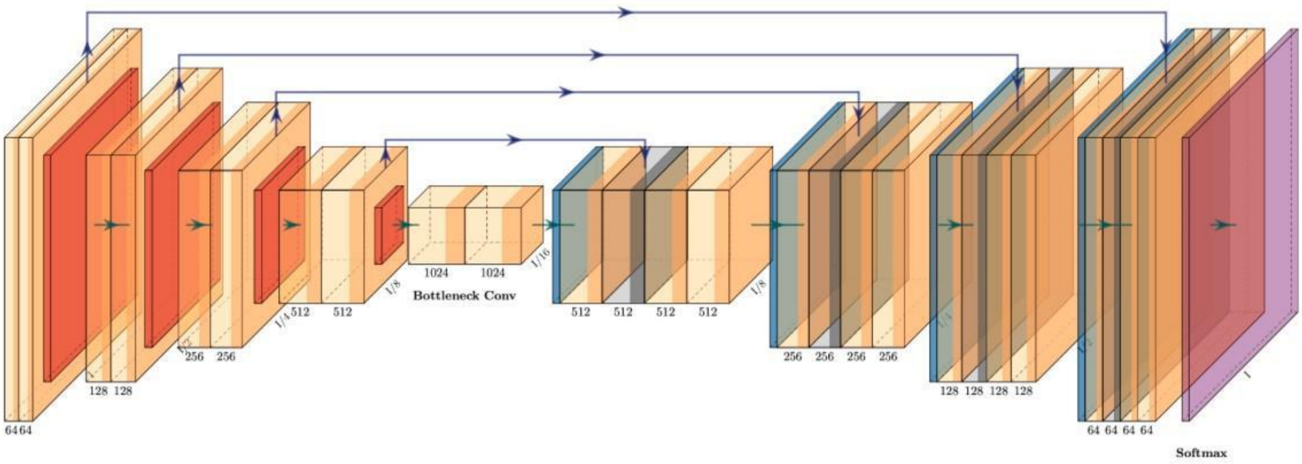

Fig. 6: Basic structure of a U-Net. Figure built using [https://github.com/HarisIqbal88/PlotNeuralNet.](https://github.com/HarisIqbal88/PlotNeuralNet)

Table 2: Comparison between building footprint datasets

Dataset	# of buildings	# of tiles	Tile Size	Spatial Resolution
LandCover.ai	12,788	41	33 tiles with the size 9000 x 9500 px and eight tiles with size 4200 x 4700 px	25cm and 50 cm
INRIA	216,418	360	5000 x 5000 px	30 cm
Massachusetts Buildings	310,425	151	1500 x 1500 px	1 m
Spacenet	462,091	17,533	512 x 512 px	35 cm
WHU building dataset	220,000	25,577	512 x 512 px	7.5 cm and 2.7 cm
AIRS	220,000	1,047	10,000 x 10,000 px	7.5 cm
CrowdAI	Unknown	280,741 training images, 60,317 validation images and 60,697 test images	300 x 300 px	Unknown

3.6. Available Frameworks and Tools

The two most famous deep learning frameworks are Tensorflow [166] and PyTorch [167]. Both are open source, have large communities, are very well documented, and have outstanding performance. Tensorflow has an underlying library called Keras [168], enabling a higher level and more readable code. On the PyTorch side, PyTorch Lightning [169], FastAI [170], and Catalyst [171], among others, are frameworks that provide similar improvements given by Keras.

Considering segmentation models tools openly available, there are two frameworks developed in Python that use Tensorflow and PyTorch, respectively segmentation models [172] and segmentation models PyTorch [173]. To train segmentation models without coding skills, users can build a JSON file with the parameters of the training and use a Python package called segmentation models trainer [174], which was built using Tensorflow, Keras, and segmentation models. [175] has also created a training framework using PyTorch and PyTorch Lightning called PyTorch segmentation models trainer, which instead of using a JSON to fill the hyperparameters, uses a YAML file using configuration composition, which enables users to reuse settings. To build training masks from vector data, a QGIS [176] plugin called DeepLearningTools [177] can be used.

There are also tools to help to build and to inspect datasets, such as FiftyOne [178]. With this tool, data scientists can visualize the labels overlapped to the images and calculate image similarity indexes to assess the quality of the dataset and identify missing labels.

Concerning data augmentation, each library has built-in operations. As external options, we can cite Albumentations [179], a Python package that is framework agnostic and works only on CPU. Another option on the PyTorch ecosystem is Kornia [180], a package that works on either CPU or GPU.

IV. CONCLUSION

In this paper, we presented the SOTA of Semantic Segmentation in Remote Sensing, an ever-growing field of research, with an almost exponential increase in the number of publications, as shown in section 2.1. We identified that the most used backbones on RS tasks are the ResNet family, VGG-16, Inception-V3, and AlexNet. Furthermore, we identified that the most famous architectures used in RS are the U-Net, DeepLabV3+, FCN, and SegNet. We also briefly showed the main theories, algorithms, and neural networks architectures and backbones.

This paper has also briefly presented how convolutional neural networks work and the techniques used for training such structures, like weight initialization, popular optimizers, some of the loss functions available, and the often-used metrics in RS papers. We also showed some of the existing regularizing techniques such as weight decay, label smoothing, early stopping, dropout, batch normalization, and data augmentation.

Then, we also presented some learning rate scheduling methods and stochastic weight averaging. We also listed the most famous backbones and architectures found on the RS papers surveyed and presented some applications of such techniques on RS. We also showed some available datasets and popular frameworks and packages to train deep learning convolutional neural networks.

There are many research papers in CS that propose several neural architectures, and some have been used in RS applications. Deep Learning is an ever-growing field, and in 2020 there have been many promising and exciting new backbones, such as the EfficientNet family, the ResNeSt-269 [49], and the SE-ResNet family [50].

Moreover, we have identified a research opportunity in RS to combine the mentioned backbones with popular architectures such as U-Net, FPNs, and PSPNet. Another research opportunity is the usage of HRNet-OCR [53], HRNetV2-OCR+PSA [56], EfficientNet-B7+CAA [55], and EfficientNet-L2+NAS-FPN [54], which are in the leader board of Papers With Code [21], but was not observed in the surveyed papers regarding remote sensing applications.

In addition, another research opportunity that we identified is to perform an extensive comparison of the accuracy of trained models with several combinations of neural networks architectures and backbones to define the best method to extract information from very-high remote sensing images. We can also highlight other research opportunities, such as determining the best loss function to be used in training and the best inference method to improve validation data accuracy. The suggested loss function for such a study is the Focal Tversky [83] since it handles class imbalance problems, a common problem in remote sensing datasets, especially building footprint extraction datasets.

Additionally, even though new optimizers such as RAdam, AdaMod, and AdaHessian have been proposed, few papers in remote sensing have tested them. The same principle can be applied to activation functions such as Leaky-ReLU, ELU, SELU, GELU, and Mish. So, we also identify research opportunities of the influence of optimizers and activation functions in the training time and the test metric scores.

Finally, other aspects that we did not find in the surveyed papers and that can be researched is the usage of stochastic weight averaging [101, 102], novel augmentation techniques such as Mixup [90], AutoAugment [91], Faster AutoAugment [92] and RandAugment [93].

ACKNOWLEDGEMENTS

We would like to acknowledge the Brazilian Army Geographic Service and the Institute of Geosciences of the University of Brasília for their support in our research.

REFERENCES

[1] Thorsten Hoeser and Claudia Kuenzer, “Object detection and image segmentation with deep learning on Earth observation data: A review-part I: Evolution and recent trends,” *Remote Sensing*, vol. 12, no. 10, 2020.

[2] Thomas Blaschke, Geoffrey JHay, Maggi Kelly, Stefan Lang, Peter Hofmann, Elisabeth Addink, Raul Queiroz Feitosa, Freek Van der Meer, Harald Van der Werff, Frieke Van Coillie, et al., “Geographic object-based image analysis–towards a new paradigm,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 87, pp. 180–191, 2014.

[3] Gang Li and Youchuan Wan, “Adaptive watershed segmentation of remote sensing image based on wavelet transform and fractal dimension,” in *Proceedings of the 2011, International Conference on Informatics, Cybernetics, and Computer Engineering (ICCE2011) November 19–20, 2011, Melbourne, Australia*. Springer, 2011, pp. 57–67.

[4] Hua Jiang, GuiLin Xu, and Jing Qin, “Research on adaptive model of remote sensing image segmentation based on graph theory,” in *Computer Application and*

System Modeling (ICCASM), 2010 International Conference on. IEEE, 2010, vol. 6, pp. V6–445.

[5] Bo Peng, Xingzheng Wang, and Yan Yang, "Region based exemplar references for image segmentation evaluation," *IEEE Signal Processing Letters*, vol. 23, no. 4, pp. 459–462, 2016.

[6] MadodomziMafanya, Philemon Tsele, Joel Botai, PhetoleManyama, Barend Swart, and Thabang Monate, "Evaluating pixel and object based image classification techniques for mapping plant invasions from uav derived aerial imagery: Harrisiapomanensis as a case study," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 129, pp. 1–11, 2017.

[7] Jim X. Chen, "The Evolution of Computing: AlphaGo," *Computing in Science and Engineering*, vol. 18, no. 4, pp. 4–7, 2016.

[8] John E. Ball, Derek T. Anderson, and Chee Seng Chan, "Comprehensive survey of deep learning in remote sensing: theories, tools, and challenges for the community," *Journal of Applied Remote Sensing*, vol. 11, no. 04, pp. 1, 2017.

[9] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation Applied to Handwritten Zip Code Recognition," *Neural Computation*, vol. 1, no. 4, pp. 541–551, dec 1989.

[10] Xiao Xiang Zhu, DevisTuia, LichaoMou, Gui Song Xia, Liangpei Zhang, Feng Xu, and Friedrich Fraundorfer, "Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources," dec 2017.

[11] K. Bittner, S. Cui, and P. Reinartz, "Building extraction from remote sensing data using fully convolutional networks," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, vol. 42, no. 1W1, pp. 481–486, 2017.

[12] Qi Chen, Lei Wang, Yifan Wu, Guangming Wu, Zhiling Guo, and Steven L. Waslander, "Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, pp. 42–55, jan 2019.

[13] Shengsheng Wang, Xiaowei Hou, and Xin Zhao, "Automatic Building Extraction from High-Resolution Aerial Imagery via Fully Convolutional EncoderDecoder Network with Non-Local Block," *IEEE Access*, vol. 8, pp. 7313–7322, 2020.

[14] Kang Zhao, Muhammad Kamran, and Gunho Sohn, "Boundary Regularized Building Footprint Extraction From Satellite Images Using Deep Neural Network," 2020.

[15] Wei Guo, Weihong Li, Weiguo Gong, and Jinkai Cui, "Extended Feature Pyramid Network with Adaptive Scale Training Strategy and Anchors for Object Detection in Aerial Images," *Remote Sensing*, vol. 12, no. 5, pp. 784, mar 2020.

[16] Guang Yang, Qian Zhang, and Guixu Zhang, "EANet: Edge-Aware Network for the Extraction of Buildings from Aerial Images," *Remote Sensing*, vol. 12, no. 13, pp. 2161, 2020.

[17] L. Hang and G. Y. Cai, "Cnn Based Detection of Building Roofs From High Resolution Satellite Images," *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-3/W10, no. November 2019, pp. 187–192, 2020.

[18] Jingjing Ma, Linlin Wu, Xu Tang, Fang Liu, Xiangrong Zhang, and Licheng Jiao, "Building Extraction of Aerial Images by a Global and Multi-Scale EncoderDecoder Network," *Remote Sensing*, vol. 12, no. 15, pp. 2350, 2020.

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 3431–3440.

[20] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross' Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[21] Facebook, "Papers with code," 2018.

[22] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211– 252, 2015.

[23] Matthew D. Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8689 LNCS, no. PART 1, pp. 818–833, 2014.

[24] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich, "Going deeper with convolutions," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 07-12-June-2015, pp. 1–9, 2015.

[25] Thorsten Hoeser, Felix Bachofer, and Claudia Kuenzer, "Object Detection and Image Segmentation with Deep Learning on Earth Observation Data: A Review—Part II: Applications," *Remote Sensing*, vol. 12, no. 18, pp. 3053, 2020.

[26] Jurgen Schmidhuber, "Deep Learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[27] Yanming Guo, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu, and Michael S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, 2016.

[28] Bo Zhao, Jiashi Feng, Xiao Wu, and Shuicheng Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *International*

Journal of Automation and Computing, vol. 14, no. 2, pp. 119–135, 2017.

[29] Yanming Guo, Yu Liu, Theodoros Georgiou, and Michael S. Lew, "A review of semantic segmentation using deep neural networks," *International Journal of Multimedia Information Retrieval*, vol. 7, no. 2, pp. 87–93, 2018.

[30] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, Pablo MartinezGonzalez, and Jose Garcia-Rodriguez, "A survey on deep learning techniques for image and video semantic segmentation," *Applied Soft Computing Journal*, vol. 70, pp. 41–65, 2018.

[31] Hongshan Yu, Zhengeng Yang, Lei Tan, Yaonan Wang, Wei Sun, Mingui Sun, and Yandong Tang, "Methods and datasets on semantic segmentation: A review," *Neurocomputing*, vol. 304, pp. 82–103, 2018.

[32] Arne Schumann, Lars Sommer, KrassimirValev, and Jurgen Beyerer, "A systematic evaluation of recent deep learning architectures for fine-grained vehicle classification," p. 1, 2018.

[33] Farhana Sultana, Abu Sufian, and Paramartha Dutta, "Advancements in image classification using convolutional neural network," *Proceedings - 2018 4th IEEE International Conference on Research in Computational Intelligence and Communication Networks, ICRCICN 2018*, pp. 122–129, 2018.

[34] Md ZahangirAlom, Tarek M. Taha, Chris Yakopcic, Stefan Westberg, PahedingSidike, MstShamima Nasrin, Mahmudul Hasan, Brian C. Van Essen, Abdul A.S.Awwal, and Vijayan K. Asari, "A state-of-the-art survey on deep learning theory and architectures," *Electronics (Switzerland)*, vol. 8, no. 3, 2019.

[35] Swarnendu Ghosh, Nibaran Das, Ishita Das, and Ujjwal Maulik, "Understanding deep learning techniques for image segmentation," *ACM Computing Surveys*, vol. 52, no. 4, 2019.

[36] Xiaolong Liu, Zhidong Deng, and Yuhan Yang, "Recent progress in semantic image segmentation," *Artificial Intelligence Review*, 2018.

[37] Asifullah Khan, AnabiaSohail, UmmeZahoora, and Aqsa Saeed Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artificial Intelligence Review*, pp. 1–70, 2020.

[38] Shijie Hao, Yuan Zhou, and Yanrong Guo, "A Brief Survey on Semantic Segmentation with Deep Learning," *Neurocomputing*, vol. 406, pp. 302–321, 2020.

[39] Shervin Minaee, Yuri Boykov, FatihPorikli, AntonioPlaza, Nasser Kehtarnavaz, and Demetri Terzopoulos, "Image Segmentation Using Deep Learning: A Survey," pp. 1–23, 2020.

[40] SaeidAsgariTaghanaki, Kumar Abhishek, Joseph PaulCohen, Julien Cohen-Adad, and Ghassan Hamarneh, "Deep semantic segmentation of natural and medical images: a review," *Artificial Intelligence Review*, 2020.

[41] Yuzhu Ji, Haijun Zhang, Zhao Zhang, and Ming Liu, "Cnn-based encoder-decoder networks for salient object detection: A comprehensive review and recent advances," *Information Sciences*, vol. 546, pp. 835–857, 2021.

[42] Shijie Hao, Yuan Zhou, and Yanrong Guo, "A brief survey on semantic segmentation with deep learning," *Neurocomputing*, vol. 406, pp. 302–321, 2020.

[43] Baojun Li, Shun Liu,Weichao Xu, and Wei Qiu, "Real-time object detection and semantic segmentation for autonomous driving," in *MIPPR 2017: Automatic Target Recognition and Navigation*. International Society for Optics and Photonics, 2018, vol. 10608, p. 106080P.

[44] Yu-Ho Tseng and Shau-Shiun Jan, "Combination of computer vision detection and segmentation for autonomous driving," in *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2018, pp. 1047–1052.

[45] Fabian Flohr, DariuGavrila, et al., "Pedcut: an iterative framework for pedestrian segmentation combining shape models and multiple data cues.," in *BMVC*, 2013.

[46] Garrick Brazil, Xi Yin, and Xiaoming Liu, "Illuminating pedestrians via simultaneous detection & segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4950–4959.

[47] Xiaofeng Zhu, Heung-Il Suk, Seong-Whan Lee, and Dinggang Shen, "Subspace regularized sparse multitask learning for multiclass neurodegenerative disease identification," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 3, pp. 607–618, 2015.

[48] Xiaofeng Zhu, Heung-Il Suk, and Dinggang Shen, "A novel matrix-similarity based loss function for joint regression and classification in ad diagnosis," *NeuroImage*, vol. 100, pp. 91–105, 2014.

[49] Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, Mu Li, and Alexander Smola, "ResNeSt: Split-Attention Networks," *arXiv*, 2020.

[50] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu, "Squeeze-and-Excitation Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 8, pp. 2011–2023, 2020.

[51] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, RodrigoBenenson, Uwe Franke, Stefan Roth, and Bernt Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[52] Longlong Jing and Yingli Tian, "Self-supervised visual feature learning with deep neural networks: A survey," 2019.

[53] Andrew Tao, Karan Sapra, and Bryan Catanzaro, "Hierarchical Multi-Scale Attention for Semantic Segmentation," 2020.

[54] Barret Zoph, GolnazGhiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D. Cubuk, and Quoc V. Le, "Rethinking pre-training and self-training," 2020.

[55] Ye Huang, Wenjing Jia, Xiangjian He, Liu Liu, Yuxin Li, and Dacheng Tao, "Channelized axial attention for semantic segmentation," 2021.

[56] Huajun Liu, Fuqiang Liu, Xinyi Fan, and Dong Huang, "Polarized self-attention: Towards high-quality pixelwise regression," 2021.

[57] QizheXie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le, "Self-training with noisy student improves imagenet classification," 2020.

[58] Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do, and Kaori Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into Imaging*, vol. 9, no. 4, pp. 611–629, 2018.

[59] Andrew L Maas, Awni Y Hannun, and Andrew YNg, "Rectifier nonlinearities improve neural network acoustic models," *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, vol. 28, 2013.

[60] Djork Arne Clevert, Thomas Unterthiner, and Sepp' Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," *4th International Conference on Learning Representations, ICLR 2016 Conference Track Proceedings*, pp. 1–14, 2016.

[61] Gunter¨ Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter, "Self-normalizing neural networks," *Advances in Neural Information Processing Systems*, vol. 2017-December, pp. 972–981, 2017.

[62] Dan Hendrycks and Kevin Gimpel, "Gaussian Error Linear Units (GELUs)," pp. 1–9, 2016.

[63] DigantaMisra, "Mish: A self-regularized nonmonotonic neural activation function," *arXiv*, 2019.

[64] Ian Goodfellow, YoshuaBengio, and Aaron Courville, *Deep learning*, MIT press, 2016.

[65] Meenal V. Narkhede, Prashant P. Bartakke, and Mukul S. Sutaone, *A review on weight initialization strategies for neural networks*, Number 0123456789. Springer Netherlands, 2021.

[66] Xavier Glorot and YoshuaBengio, "Understanding the difficulty of training deep feedforward neural networks," *Journal of Machine Learning Research*, vol. 9, pp. 249–256, 2010.

[67] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Delving deep into rectifiers: Surpassing humanlevel performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[68] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu, "A survey on deep transfer learning," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11141 LNCS, pp. 270–279, 2018.

[69] Ian Goodfellow, YoshuaBengio, and Aaron Courville, *Deep Learning*, 2017.

[70] Leon Bottou,' "Large-Scale Machine Learning with Stochastic Gradient Descent," *Proceedings of COMPSTAT'2010*, 2010.

[71] John Duchi, Elad Hazan, and Yoram Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *Journal of Machine Learning Research*, vol. 12, pp. 2121–2159, 2011.

[72] Y. NESTEROV, "A method for solving the convex programming problem with convergence rate O(1/kˆ2)," 1983.

[73] Diederik P Kingma and Jimmy Lei Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015, pp. 1–15.

[74] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han, "On the Variance of the Adaptive Learning Rate and Beyond," pp. 1–14, 2019.

[75] Jianbang Ding, Xuancheng Ren, Ruixuan Luo, and Xu Sun, "An adaptive and momental bound method for stochastic learning," *arXiv*, 2019.

[76] Zhewei Yao, Amir Gholami, Sheng Shen, Kurt Keutzer, and Michael W. Mahoney, "ADAHESSIAN: An Adaptive Second Order Optimizer for Machine Learning," *arXiv*, pp. 1–20, 2020.

[77] Jun Ma, "Segmentation Loss Odyssey," 2020.

[78] Kevin P Murphy, *Machine learning: a probabilistic perspective*, MIT press, 2012.

[79] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, vol. 9351, pp. 234–241.

[80] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*. IEEE, 2016, pp. 565–571.

[81] Md Atiqur Rahman and Yang Wang, "Optimizing intersection-over-union in deep neural networks for image segmentation," in *International symposium on visual computing*. Springer, 2016, pp. 234–244.

[82] SeyedRaein Hashemi, SeyedSadeghMohseni Salehi, Deniz Erdogmus, Sanjay P Prabhu, Simon K Warfield, and Ali Gholipour, "Asymmetric loss functionsand deep densely-connected networks for highlyimbalanced medical image segmentation: Application to multiple sclerosis lesion detection," *IEEE Access*, vol. 7, pp. 1721–1735, 2018.

[83] Nabila Abraham and NaimulMefraz Khan, "A novel focal tversky loss function with improved attention unet for lesion segmentation," 2018.

[84] Foivos I. Diakogiannis, Franc¸ois Waldner, Peter Caccetta, and Chen Wu, "ResUNet-a: a deep learning framework for semantic segmentation of remotely sensed data," 2019.

[85] Zeju Li, Konstantinos Kamnitsas, and Ben Glocker, "Overfitting of neural nets under class imbalance: Analysis and improvements for segmentation," in

International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, 2019, pp. 402–410.

[86] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[87] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014.

[88] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015.

[89] Connor Shorten and Taghi M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data*, vol. 6, no. 1, 2019.

[90] Hongyi Zhang, MoustaphaCisse, Yann N. Dauphin, and David Lopez-Paz, "mixup: Beyond empirical risk minimization," 2018.

[91] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le, "Autoaugment: Learning augmentation policies from data," 2019.

[92] Ryuichiro Hataya, Jan Zdenek, KazukiYoshizoe, and Hideki Nakayama, "Faster AutoAugment: Learning Augmentation Strategies Using Backpropagation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics),* vol. 12370 LNCS, pp. 1–16, 2020.

[93] Ekin D. Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V. Le, "Randaugment: Practical automated data augmentation with a reduced search space," 2019.

[94] Jieun Park, Dokkyun Yi, and Sangmin Ji, "A novel learning rate schedule in optimization for neural networks and it's convergence," *Symmetry*, vol. 12, no. 4, 2020.

[95] Leslie N. Smith, "Cyclical learning rates for training neural networks," *Proceedings - 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017*, no. April 2015, pp. 464–472, 2017.

[96] Ilya Loshchilov and Frank Hutter, "Sgdr: Stochastic gradient descent with warm restarts," 2017.

[97] YuriiNesterov, "Introduction to convex optimization: A basic course," 2004.

[98] Zhiyuan Li and Sanjeev Arora, "An exponential learning rate schedule for deep learning," *arXiv preprint arXiv:1910.07454*, 2019.

[99] Andrew Hundt, Varun Jain, and Gregory D. Hager, "sharpdarts: Faster and more accurate differentiable architecture search," 2019.

[100] Leslie N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay," 2018.

[101] Pavel Izmailov, DmitriiPodoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson, "Averaging weights leads to wider optima and better generalization," *arXiv preprint arXiv:1803.05407*, 2018.

[102] Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson, "There are many consistent explanations of unlabeled data: Why you should average," *arXiv preprint arXiv:1806.05594*, 2018.

[103] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and JianSun, "Deep residual learning for image recognition," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 770–778, 2016.

[104] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," 2016.

[105] Franc¸ois Chollet, "Xception: Deep learning with depthwise separable convolutions," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 1800–1807, 2017.

[106] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[107] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[108] Andrew Howard, Mark Sandler, Grace Chu, LiangChieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al., "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1314–1324.

[109] Dan Cires¸anCires¸an, Ueli Meier, and Jurgen Schmid-¨huber, "Multi-column Deep Neural Networks for Image Classification," Tech. Rep., 2012.

[110] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *COMMUNICATIONS OF THE ACM*, vol. 60, no. 6, 2017.

[111] YoshuaBengio, Patrice Simard, and Paolo Frasconi, "Learning Long-Term Dependencies with Gradient Descent is Difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994.

[112] Saining Xie, Ross Girshick, Piotr Dollar, Zhuowen' Tu, and Kaiming He, "Aggregated residual transformations for deep neural networks," *Proceedings 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 5987– 5995, 2017.

[113] Barret Zoph and Quoc V. Le, "Neural architecture search with reinforcement learning," *5th International Conference on Learning Representations, ICLR 2017 Conference Track Proceedings*, pp. 1–16, 2017.

[114] Chenxi Liu, Liang-Chieh Chen, Florian Schroff, Hartwig Adam, Wei Hua, Alan Yuille, and Li FeiFei, "Auto-

DeepLab: Hierarchical Neural Architecture Search for Semantic Image Segmentation," Tech. Rep.

[115] GolnazGhaisi, Tsung-Yi Lin, Ruoming Pang Quoc, and V Le Google Brain, "NAS-FPN: Learning Scalable Feature Pyramid Architecture for Object Detection," Tech. Rep.

[116] Mingxing Tan and Quoc V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," Tech. Rep., 2019.

[117] Mingxing Tan, Ruoming Pang, and Quoc V Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10781– 10790.

[118] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, SherjilOzair, Aaron Courville, and YoshuaBengio, "Sparse generative adversarial network," *Proceedings - 2019 International Conference on Computer Vision Workshop, ICCVW 2019*, pp. 3063–3071, 2019.

[119] Clint Sebastian, Raffaele Imbriaco, Egor Bondarev, and Peter H. N. de With, "Adversarial Loss for Semantic Segmentation of Aerial Imagery," 2020.

[120] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully Convolutional Networks for Semantic Segmentation," Tech. Rep.

[121] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille, "Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs," *ICLR 2015*, dec 2014.

[122] Liang Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, apr 2018.

[123] Wei Liu, Andrew Rabinovich, and Alexander C. Berg, "ParseNet: Looking Wider to See Better," 2015.

[124] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia, "Pyramid scene parsing network," *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 6230–6239, 2017.

[125] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," Tech. Rep., 2017.

[126] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han, "Learning deconvolution network for semantic segmentation," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 International Conference on Computer Vision, ICCV 2015, pp. 1520–1528, 2015.

[127] Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla, "Segnet: A deep convolutional encoderdecoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[128] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, NimaTajbakhsh, and Jianming Liang, "Unet++: A nested u-net architecture for medical image segmentation," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11045 LNCS, pp. 3–11, 2018.

[129] Debesh Jha, Michael A. Riegler, Dag Johansen, Pal˚ Halvorsen, and Havard D. Johansen, "DoubleU-Net: A˚ Deep Convolutional Neural Network for Medical Image Segmentation," jun 2020.

[130] Nabil Ibtehaz and M Sohel Rahman, "MultiResUNet : Rethinking the U-Net architecture for multimodal biomedical image segmentation," *Neural Networks*, vol. 121, pp. 74–87, 2020.

[131] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," *Proceedings 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 5168–5177, 2017.

[132] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam, "EncoderDecoder with Atrous Separable Convolution for Semantic Image Segmentation," Tech. Rep.

[133] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao, "Deep high-resolution representation learning for visual recognition," 2020.

[134] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and IlliaPolosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[135] Yuhui Yuan, Xiaokang Chen, Xilin Chen, and Jingdong Wang, "Segmentation transformer: Objectcontextual representations for semantic segmentation," 2021.

[136] Mabel Ortega Adarme, Raul Queiroz Feitosa, Patrick NigriNigriHapp, Claudio Aparecido De Almeida, and Alessandra Rodrigues Gomes, "Evaluation of Deep Learning Techniques for Deforestation Detection in the Brazilian Amazon and Cerrado Biomes From Remote Sensing Imagery," *Remote Sensing*, vol. 12, no. 6, pp. 910, mar 2020.

[137] NicholusMboga, Stefanos Georganos, Tais Grippa, Moritz Lennert, Sabine Vanhuysse, and Eleonore' Wolff, "Fully convolutional networks and geographic object-based image analysis for the classification of VHR imagery," *Remote Sensing*, vol. 11, no. 5, 2019.

[138] Emilio Guirado, SihamTabik, Domingo AlcarazSegura, Javier Cabello, and Francisco Herrera, "Deeplearning Versus OBIA for scattered shrub detection with Google Earth Imagery: Ziziphus lotus as case study," *Remote Sensing*, vol. 9, no. 12, pp. 1–22, 2017.

[139] Musyarofah, Valentina Schmidt, and Martin Kada, "Object detection of aerial image using mask-region convolutional neural network (mask R-CNN)," in *IOP*

Conference Series: Earth and Environmental Science. jul 2020, vol. 500, Institute of Physics Publishing.

[140] Kun Li, Xiangyun Hu, Huiwei Jiang, Zhen Shu, and Mi Zhang, "Attention-Guided Multi-Scale Segmentation Neural Network for Interactive Extraction of Region Objects from High-Resolution Satellite Imagery," 2020.

[141] Hyperspectral Data, Yushi Chen, Zhouhan Lin, YushiChen, Zhouhan Lin, Xing Zhao, and Student Member, "Deep Learning-Based Classification of Hyperspectral Data," vol. 7, no. June 2014, pp. 1–14, 2015.

[142] Charis Lanaras, Jose' Bioucas-Dias, SilvanoGalliani, Emmanuel Baltsavias, Konrad Schindler, Remote Sensing, and Eth Zurich, "Super-resolution of Sentinel-2 images: Learning a globally applicable deep neural network," 2018.

[143] Mengjiao Qin, Sebastien' Mavromatis, Linshu Hu, Feng Zhang, Renyi Liu, Jean Sequeira, and Zhenhong Du, "Remote Sensing Single-Image Resolution Improvement Using A Deep Gradient-Aware Network with Image-Specific Enhancement," *Remote Sensing*, vol. 12, no. 5, pp. 758, feb 2020.

[144] Francesco Salvetti, Vittorio Mazzia, Aleem Khaliq, and Marcello Chiaberge, "Multi-Image Super Resolution of Remotely Sensed Images Using Residual Attention Deep Neural Networks," *Remote Sensing*, vol. 12, no. 14, pp. 2207, jul 2020.

[145] Yi Liu, Chao Pang, Zongqian Zhan, Xiaomeng Zhang, and Xue Yang, "Building Change Detection for Remote Sensing Images Using a Dual Task Constrained Deep Siamese Convolutional Network Model," 2019.

[146] Qing Wang, Xiaodong Zhang, Guanzhou Chen, Fan Dai, Yuanfu Gong, and Kun Zhu, "Change detection based on Faster R-CNN for high-resolution remote sensing images," *Remote Sensing Letters*, vol. 9, no. 10, pp. 923–932, oct 2018.

[147] Liangpei Zhang, Lefei Zhang, and Bo Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geoscience and Remote Sensing Magazine*, vol. 4, no. 2, pp. 22–40, 2016.

[148] Ronald Kemker, Carl Salvaggio, and Christopher Kanan, "Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 145, pp. 60–77, 2018.

[149] Hui Yang, Penghai Wu, Xuedong Yao, Yanlan Wu, Biao Wang, and Yongyang Xu, "Building extraction in very high-resolution imagery by dense-attention networks," *Remote Sensing*, vol. 10, no. 11, pp. 1–16, 2018.

[150] Lili Zhang, Jisen Wu, Yu Fan, Hongmin Gao, and Yehong Shao, "An efficient building extraction method from high spatial resolution remote sensing images based on improved mask R-CNN," *Sensors (Switzerland)*, vol. 20, no. 5, pp. 1–13, 2020.

[151] Yiheng Zhang, ZhaofanQiu, Ting Yao, Dong Liu, and Tao Mei, "Fully Convolutional Adaptation Networks for Semantic Segmentation," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 6810–6818, 2018.

[152] Ming Wu, Chuang Zhang, Jiaming Liu, Lichen Zhou, and Xiaoqi Li, "Towards Accurate High Resolution Satellite Image Semantic Segmentation," *IEEE Access*, vol. 7, pp. 55609–55619, 2019.

[153] Renbao Lian and Liqin Huang, "DeepWindow: Sliding Window Based on Deep Learning for Road Extraction from Remote Sensing Images," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, no. d, pp. 1–1, 2020.

[154] Shahab EddinJozdani, Brian Alan Johnson, and Dongmei Chen, "Comparing Deep Neural Networks, Ensemble Classifiers , and Support Vector Machine Algorithms," *Remote Sensing*, vol. 11, no. 1, pp. 1–24, 2019.

[155] Manuel Carranza-Garc´ıa, Jorge Garc´ıa-Gutierrez, and' Jose C. Riquelme,' "A framework for evaluating land use and land cover classification using convolutional neural networks," *Remote Sensing*, vol. 11, no. 3, 2019. [156] ISPRS, "2d semantic labeling contest," 2012.

[157] Ahram Song and Jaewan Choi, "Fully Convolutional Networks with Multiscale 3D Filters and Transfer Learning for Change Detection in High Spatial Resolution Satellite Images," *Remote Sensing*, vol. 12, no. 5, pp. 799, mar 2020.

[158] The SpaceNet Catalog, "Spacenet on amazon web services (aws)," 2018.

[159] Adam Van Etten, Dave Lindenbaum, and Todd Bacastow, "SpaceNet: A remote sensing dataset and challenge series," *arXiv*, 2018.

[160] Volodymyr Mnih, "Machine Learning for Aerial Image Labeling," *PhD Thesis*, p. 109, 2013.

[161] Yang Long, Gui Song Xia, Shengyang Li, Wen Yang, Michael Ying Yang, Xiao Xiang Zhu, Liangpei Zhang, and Deren Li, "DiRS: On creating benchmark datasets for remote sensing image interpretation," *arXiv*, pp. 1–22, 2020.

[162] Emmanuel Maggiori, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez, "Can semantic labeling methods generalize to any city? The inria aerial image labeling benchmark," *International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2017-July, pp. 3226–3229, 2017.

[163] Adrian Boguszewski, Dominik Batorski, Natalia Ziemba-Jankowska, Anna Zambrzycka, and Tomasz Dziedzic, "LandCover.ai: Dataset for Automatic Mapping of Buildings, Woodlands and Water from Aerial Imagery," pp. 1–14, 2020.

[164] Qi Chen, Lei Wang, Yifan Wu, Guangming Wu, Zhiling Guo, and Steven L. Waslander, "Aerial imagery for roof segmentation: A large-scale dataset towards automatic mapping of buildings," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, no. November 2018, pp. 42–55, 2019.

[165] Sharada Prasanna Mohanty, Jakub Czakon, Kamil A Kaczmarek, Andrzej Pyskir, Piotr Tarasiewicz, Saket Kunwar, Janick Rohrbach, Dave Luo, Manjunath Prasad,

Sascha Fleer, et al., “Deep learning for understanding satellite imagery: An experimental survey,” *Frontiers in Artificial Intelligence*, vol. 3, 2020.

[166] Mart´ın Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, SanjayGhemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, RafalJozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mane, Rajat Monga, Sherry Moore,’ Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viegas, Oriol Vinyals, Pete Warden, Martin’ Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng, *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, 2015, Software available from tensorflow.org.

[167] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, AlykhanTejani, SasankChilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and SoumithChintala, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alche-Buc, E. Fox, and R. Garnett, Eds., pp. 8024–’ 8035. Curran Associates, Inc., 2019.

[168] Francois Chollet et al., “Keras,” https://github. com/fchollet/keras, 2015.

[169] WA Falcon, “Pytorch lightning,” *GitHub. Note:https://github.com/PyTorchLightning/pytorchlightning*, vol. 3, 2019.

[170] Jeremy Howard and Sylvain Gugger, “Fastai: A layered api for deep learning,” *Information (Switzerland)*, vol. 11, no. 2, pp. 1–26, 2020.

[171] Sergey Kolesnikov, “Accelerated deep learning rd,” https://github.com/catalyst-team/ catalyst, 2018.

[172] Pavel Yakubovskiy, “Segmentation models,” https://github.com/qubvel/ segmentation_models, 2019.

[173] Pavel Yakubovskiy, “Segmentation models pytorch,” https://github.com/qubvel/ segmentation_models.pytorch, 2020.

[174] Philipe Borba, “phborba/segmentation models trainer: First Release,” sep 2020.

[175] Philipe Borba, “phborba/pytorch segmentation models trainer: Version 0.8.0,” July 2021.

[176] QGIS Development Team, *QGIS Geographic Information System*, Open-Source Geospatial Foundation, 2009.

[177] Philipe Borba, “phborba/DeepLearningTools: First release,” oct 2020.

[178] B. E. Moore and J. J. Corso, “Fiftyone,” *GitHub. Note: https://github.com/voxel51/fiftyone*, 2020.

[179] E. Khvedchenya V. I. Iglovikov A. Buslaev, A. Parinov and A. A. Kalinin, “Albumentations: fast and flexible image augmentations,” *ArXiv e-prints*, 2018.

[180] J. Shi D. Ponsa F. Moreno-Noguer E. Riba, D. Mishkin, and G. Bradski, “A survey on kornia: an open-source differentiable computer vision library for PyTorch,” 2020.