forests

MDPI

Article

Attention-Based Semantic Segmentation Networks for Forest Applications

See Ven Lim [1], Mohd Asyraf Zulkifley [1,*], Azlan Saleh [1], Adhi Harmoko Saputro [2] and Siti Raihanah Abdani [3]

[1] Department of Electrical, Electronic & Systems Engineering, Faculty of Engineering & Built Environment, Universiti Kebangsaan Malaysia, Bangi 43600, Malaysia; azlansaleh@ukm.edu.my (A.S.)
[2] Faculty of Mathematics and Natural Science, Universitas Indonesia, Depok 16424, Indonesia
[3] School of Computing Sciences, College of Computing, Informatics and Mathematics, Universiti Teknologi MARA, Shah Alam 40450, Malaysia
* Correspondence: asyraf.zulkifley@ukm.edu.my

Abstract: Deforestation remains one of the key concerning activities around the world due to commodity-driven extraction, agricultural land expansion, and urbanization. The effective and efficient monitoring of national forests using remote sensing technology is important for the early detection and mitigation of deforestation activities. Deep learning techniques have been vastly researched and applied to various remote sensing tasks, whereby fully convolutional neural networks have been commonly studied with various input band combinations for satellite imagery applications, but very little research has focused on deep networks with high-resolution representations, such as HRNet. In this study, an optimal semantic segmentation architecture based on high-resolution feature maps and an attention mechanism is proposed to label each pixel of the satellite imagery input for forest identification. The selected study areas are located in Malaysian rainforests, sampled from 2016, 2018, and 2020, downloaded using Google Earth Pro. Only a two-class problem is considered for this study, which is to classify each pixel either as forest or non-forest. HRNet is chosen as the baseline architecture, in which the hyperparameters are optimized before being embedded with an attention mechanism to help the model to focus on more critical features that are related to the forest. Several variants of the proposed methods are validated on 6120 sliced images, whereby the best performance reaches 85.58% for the mean intersection over union and 92.24% for accuracy. The benchmarking analysis also reveals that the attention-embedded high-resolution architecture outperforms U-Net, SegNet, and FC-DenseNet for both performance metrics. A qualitative analysis between the baseline and attention-based models also shows that fewer false classifications and cleaner prediction outputs can be observed in identifying the forest areas.

Keywords: forest; remote sensing; deep learning; attention mechanism; artificial intelligence

check for updates

Citation: Lim, S.V.; Zulkifley, M.A.; Saleh, A.; Saputro, A.H.; Abdani, S.R. Attention-Based Semantic Segmentation Networks for Forest Applications. *Forests* **2023**, *14*, 2437. https://doi.org/10.3390/f14122437

Academic Editor: Qiaolin Ye

Received: 13 November 2023
Revised: 5 December 2023
Accepted: 12 December 2023
Published: 14 December 2023

1. Introduction

Forests provide significant ecological importance, including but not limited to biodiversity maintenance, habitats for various flora and fauna, the mitigation of climate change effects, watershed protection [1], erosion protection, carbon sequestration [2], and precipitation level maintenance. Forests also provide economic benefits, with raw materials such as timber [3], food, and medicine, which can drive commercial activities, contribute to people's livelihoods, and lead the development of the national economy. Malaysia's tropical rainforests constitute one of the twelve mega-diverse ecosystems in the world, housing approximately 152,000 fauna species and 15,000 flora species [4]. However, forests nowadays are subjected to uncontrolled deforestation due to various reasons, such as commodity extraction, agricultural expansion, and urbanization, and deforestation remains one of the most concerning issues around the world. Based on the public data from Global Forest Watch created by a collaboration between the University of Maryland, Google USGS,

and NASA, for the years 2000 to 2020, Malaysia experienced a net change of −1.12 Mha in tree cover. From 2016 to 2021, the total tree cover loss in Malaysia was 2.43 Mha, where 91.4% of the total loss (2.22 Mha) was caused by commodity-driven deforestation [5].

The Ministry of Energy and Natural Resources (KeTSA) [6] has been monitoring the forest status for many decades through a national forest monitoring program. The Ministry has deployed remote sensing technology to speed up the survey process while reducing the human labor needed for forest monitoring efforts. Initially, panchromatic aerial photographs were used as a remote sensing imaging source, before being replaced by satellite imagery starting from 1991 due to advancements in satellite technology and its effectiveness in remote sensing applications. Previously, the main approach for remote-sensing-based monitoring systems relied on a combination of elementary spectral bands—for example, the normalized difference vegetation index (NDVI) to distinguish green vegetation's spectral features [6].

There has been growing interest in applying automated techniques to remote sensing to rapidly identify the forest cover area, especially by using conventional machine learning techniques. Several studies have implemented conventional machine learning algorithms in the field of forestry-related remote sensing tasks, such as decision trees, random forest classification, and support vector machines. However, conventional machine learning algorithms are dependent on the type of information or features set by the algorithm's designer and have a limited capability to extract complex and deep features by themselves. The performance of conventional machine learning algorithms can also be very case-specific, which limits the scalability of such machine learning models to other applications. Therefore, many researchers prefer to employ deep learning methods instead of conventional machine learning algorithms, even though they are usually only effective for large datasets [7].

Deep learning methods, which have been applied in several forest-related applications, are a subtype of machine learning that enables feature learning from a set of large data. It is a type of representation learning method that can learn features directly from raw data for accurate detection and classification results. Usually, its architecture consists of composite multilevel representations, using non-linear functions to transform the representations from one-level to higher-level and more abstract representations, enabling the model to achieve complex feature learning [8]. There are various types of deep learning models available, such as convolutional neural networks and recurrent neural networks. Convolutional neural networks (CNNs) are built specifically to handle images in the form of multiple arrays. In terms of accuracy, flexibility, and rapid processing, CNNs perform better than conventional methods [9]. The classic CNN architecture uses multiple convolution and pooling operations to extract useful features from images before being passed to the fully connected layers. Furthermore, the fully connected layers can be replaced with upsampling operations for image segmentation purposes, creating fully convolutional neural networks. Further research has produced various improvised state-of-the-art architectures based on CNNs, such as the Siamese neural network, DeepLab, and U-Net. According to Elizar et al. [10], with the development of fully convolutional neural networks, the segmentation performance based on deep learning methods has improved dramatically in the past few years, especially when compared to the conventional machine learning approach.

The majority of the studies conducted in remote-sensing-based forestry applications use a deep learning approach, specifically convolutional neural networks, to effectively learn features for the accurate classification of forest images. The most used architecture is U-Net, due to its success in performing various segmentation tasks relating to semantic alienation. In general, the CNN architecture for segmentation tasks follows the encoder–decoder topology, whereby the encoder extracts information from low-resolution representations, while the decoder reconstructs high-resolution representations from the extracted low-resolution feature maps. However, very few studies have used high-resolution and multi-resolution fusion networks for remote sensing tasks, which is expected to yield semantically rich and spatially precise segmentation performance [11]. The minimum number of input bands required for effective forest classification also varies across stud-

ies, resulting in varying performance. Certain satellite spectral bands may not always be publicly available, and the creation of vegetation indices also requires additional data pre-processing. However, CNNs may be able to achieve sufficient classification quality without needing a large number of input bands. They can extract the necessary information for forest mapping not only based on the pixel color but also based on the pixel context [12].

In this paper, an optimal forest monitoring system based on the novel architecture of attention-based HRNet is proposed using Landsat-8 satellite imagery. The proposed approach uses HRNet as the base model due to its capability to preserve high-resolution representations and fuse representations of different resolutions together, producing good segmentation performance. The study focuses on applying the model to classify forest and non-forest areas in satellite images across several years, namely 2016, 2018, and 2020. Then, the performance of the optimal baseline HRNet is further improved by embedding an attention mechanism into the network.

2. Literature Review and Related Works

There are two main types of image classification used in remote sensing applications, which are pixel-based classification and object-based classification. Pixel-based classification assigns a class to every pixel in an image. It can be further divided into unsupervised and supervised methods. On the other hand, object-based classification groups pixels into objects that have representative vector shapes in terms of size and geometry.

A CNN is a popular deep learning model used for image processing and object detection, and it has multiple hidden layers [10,13]. Activation functions make the CNN more nonlinear and improve its expression ability. Pooling layers in CNNs perform downsampling and extract important features while reducing the dimensions of hidden layers. The fully connected layer connects the last few layers of the CNN and acts as a classifier to determine the probability of a pixel belonging to a certain class. The output layer consists of one neuron per class category, and all neurons are connected to the fully connected neurons [14].

Dong et al. [15] developed a fusion model of a CNN and random forest (RF) to classify subtropical areas in Taihuyuan, China, using satellite imagery. The model replaces the fully connected layer of the CNN with the RF classifier, resulting in improved performance. However, the model is computationally expensive due to its size and the number of inputs. Khan et al. [16] used a CNN to detect forest changes in Melbourne, Australia, using satellite imagery. They used bounding boxes to label the changes and found that the deep CNN model had higher accuracy and mean IoU compared to other methods. However, accurately producing bounding boxes for forest regions can be difficult, and the method cannot predict forest cover areas.

Fully convolutional networks (FCN) are used to obtain high-level representations from low-level representations by substituting the fully connected layer of the CNN with locally connected layers [17]. This replaced section forms the decoder part of the FCN, creating an encoder–decoder topology. Some improved variants of FCN-based networks are Siamese neural networks, DeepLab, and U-Net.

Siamese neural networks (SNN) consist of two identical CNNs that share weights during encoding. An SNN can also identify similarities and differences between inputs by computing distance metrics. Therefore, SNNs have been applied in visual and change detection tasks, including video target tracking [18] and landscape change detection [19]. Guo et al. [20] utilized a fully convolutional SNN to identify forest changes in Nanning and Fuzhou, China, using Landsat-8 satellite imagery. They introduced a modified version of Caye Daudt et al.'s model [21], called Siamese, which employs concatenation weight-sharing and subtraction weight-sharing methods. This modification prioritizes change information in different layers while preserving detailed image information. The results showed that the method achieved accurate deforestation and afforestation detection and good IoU scores.

Chen et al. [22] introduced the DeepLab series of advanced deep learning segmentation models in 2016. The latest version, DeepLabv3+, has a similar design to previous models but with some differences. Andrade et al. [23] utilized DeepLabv3+ to detect deforestation and found that this model outperformed other models. DeepLabv3+ also performs well with limited dataset sizes, demonstrating its superior generalization capacity. However, they found that there was a potential bias in the trained models due to the large imbalance between deforested and non-deforested areas. Ferreira et al. [24] applied deep learning to map Brazil nut trees in the Amazonian rainforest using WorldView-3 satellite imagery by adopting the DeepLabv3+ architecture with three different encoder backbones: ResNet-18, ResNet-50, and MobileNetV2. In their study, DeepLabv3+ with all three different backbones achieved almost similar accuracy in mapping Brazil nut trees. The authors noted that the shadows of Brazil nut trees were important features for proper mapping.

U-Net is a popular deep learning architecture designed for semantic segmentation tasks. It was designed by Ronneberger et al. [25] originally for biomedical image segmentation. Since then, U-Net has shown tremendous success in various image segmentation tasks for various applications, including forestry-related tasks. One modification in the U-Net architecture is the use of a large number of feature channels in the upsampling part, allowing context information propagation to higher-resolution layers. Abdani et al. [26] enhanced the multi-scale capability of U-Net by incorporating an SPP module. Bragagnolo et al. [27] used U-Net to map forest cover changes in the Amazon rainforest and compared its performance with that of other deep learning architectures. The results showed that U-Net and ResNet50-SegNet had high accuracy and F1 scores for forest cover mapping segmentation. U-Net also had the lowest training time among the benchmarked architectures. One advantage of the authors' approach is the model's ability to tolerate some misclassification without significantly affecting the performance.

A study by Wagner et al. [12] examined Amazon forest cover and deforestation from 2015 to 2022 using U-Net with the pixel-based approach. Images were obtained from the Planet API with the PlanetNICFI R package v1.0.4. Forest cover masks were created with the K-textures algorithm. U-Net achieved high accuracy and F1-scores in forest cover segmentation, validated using airborne LiDAR data. However, the model faced difficulties in detecting deforestation, particularly in identifying burnt areas.

3. Methodology

3.1. Study Area

The regions chosen to prepare the model's dataset were Malaysian rainforests. Malaysian rainforests display vast biodiversity in terms of fauna and flora and is one of the twelve mega-diverse ecosystems in the world. There are approximately 306 species of wild mammals, 742 species of birds, 242 species of amphibians, more than 449 freshwater species, and more than 150,000 estimated species of invertebrates. Malaysia's flora biodiversity also constitutes approximately 15,000 species of vascular plants [4].

3.2. Dataset Preparation

Satellite images are downloaded at an eye altitude of approximately 8000 ft so that forest features can be differentiated clearly from others. The satellite images are from the Landsat-8 satellite, captured from February 2013 to September 2021. A set of ten regions of land parcels from the years 2016, 2018, and 2020 are selected, downloaded, and saved in Portable Network Graphics (PNG) format at the highest image resolution possible, constituting a total of 30 land plots. High-resolution satellite imagery helps to provide sufficient contextual information of pixels for the efficient training of deep learning models. The temporal resolution is kept constant for all land plots to maintain uniformity and control the variations in spectral distribution across the years. The forest and non-forest areas of the chosen regions are approximately balanced with a ratio of 50:50. Some examples of non-forest areas that are included in the dataset are deforested areas, plantations, human-

made buildings, and roads and highways. The chosen images are verified to contain little or no cloud cover, to prevent noise from being introduced in the training of the model.

Masks are created manually for every plot of land using the GNU Image Manipulation Program 2 (GIMP-2) software 2.10.22 (GIMP Team, Kernersville, NC, USA, https://www.gimp.org/) and under the supervision of a supervisor. The annotation of masks is performed by overlaying a mask layer on top of a satellite image, labeling all image pixels belonging to forest areas as high and labeling all image pixels belonging to non-forest areas as low, and saving the final mask layer as a separate mask image. Final masks consist of two classes based on their labels: white-colored pixels represent forest areas, and black-colored pixels represent non-forest areas. The annotated land plots and their masks are then divided into smaller patches of 224 pixels × 224 pixels, creating a dataset of 17,280 patches to facilitate processing and allow parallel processing, in addition to computer memory limitations. The OpenCV library is used to process the land sub-image patches into an array of three vector values per pixel, constituting the RGB channels. The RGB values are in the range of 0 to 255, which are then normalized to the range of −1 to 1. Mask patches are also processed into arrays of 0 s (non-forest) and 1 s (forest). Hence, the model has three input channels from the normalized RGB sub-image and the respective binary mask.

3.3. HRNet Architecture

The concept behind the HRNet architecture is straightforward, as it involves maintaining a high resolution throughout the process, conducting parallel sampling and fusion, and ultimately producing feature maps with various resolutions [28]. These feature maps can be selectively combined and utilized based on the specific requirements of different tasks. The feature map maintains its high resolution during the entire process, which is the primary characteristic of the HRNet model [29]. The low-resolution feature map subnetworks are added to the primary high-resolution feature map network in order to perform the multi-scale fusion and feature extraction of several networks and to realize the model [29]. HRNet is one of the latest novel architectures published by Wang et al. [11]. This model's unique feature, distinguishing it from other state-of-the-art architectures, is the connection of high-to-low-convolution streams, arranged in parallel, which preserve high-resolution representations throughout the entire process, and the repetitive fusion of representations from multi-resolution streams to generate reliable high-resolution representations with strong position sensitivity.

As shown in Figure 1, the main body of the network consists of four stages, beginning with a high-resolution convolution stream in the first stage. For every new stage, one new high-to-low-resolution stream with half the resolution of the previous higher-resolution stream is added, and the multi-resolution streams are connected in parallel. Consequently, every later stage in the parallel stream consists of resolutions from the previous stage and a new lower one.

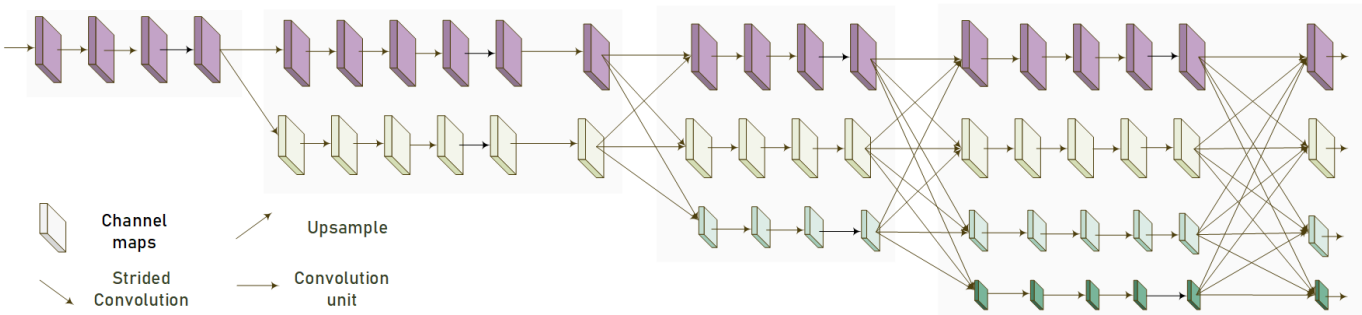

Figure 1. Architecture of HRNet.

Information sharing across multi-resolution representations is achieved by fusing the modules repeatedly at the end of every stage, as shown in Figure 2. Each output representation is the sum of the transformed representations of the input representations.

The transformation of input representations is dependent on the resolution between the input and the output representations. The authors tested the performance against other state-of-the-art architectures in various applications, including human pose estimation, semantic segmentation, and object detection. In their study, HRNet outperformed other tested state-of-the-art algorithms in terms of average precision for human pose estimation and object detection applications and the mean of the class-wise intersection over union for semantic segmentation applications.

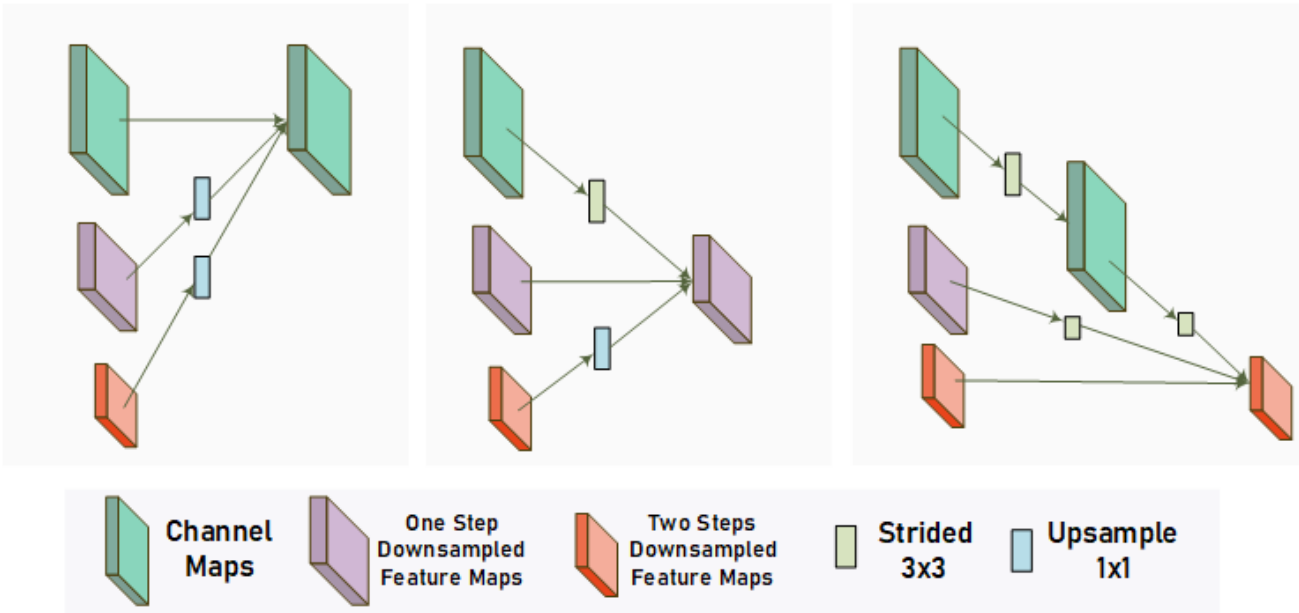

Figure 2. Multi-resolution representation fusion module in HRNet.

3.4. Performance Metrics

To evaluate the quality of the model's forest segmentation ability after training, three performance metrics are chosen: accuracy, mean intersection over union, and categorical cross-entropy loss. Accuracy is defined by the number of correctly classified pixels divided by the total number of pixels in a single image patch.

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

where *TP* is the pixels correctly classified as positive, *TN* is the pixels correctly classified as negative, *FP* is the pixels falsely classified as positive, and *FN* is the pixels incorrectly classified as negative.

Accuracy alone is not a sufficient performance metric as it can be affected by class imbalances in the dataset [30]. Hence, the intersection over union is introduced to complement the accuracy. The intersection over union (*IoU*), also known as the Jaccard index, is defined as the number of pixels correctly classified as true divided by the total number of pixels correctly classified as true and the number of falsely classified pixels. *IoU* shows the degree of overlap between the ground truth and the prediction. In the case of binary classification between forest and non-forest areas, two different IoUs from forest and non-forest areas, respectively, are measured and averaged to show the overall mean *IoU* performance of the segmentation network, known as mIoU. This study uses mIoU as the main decision factor to select the best-performing models.

$$IoU = \frac{TP}{TP + FP + FN} \tag{2}$$

Another performance metric used in the study is the categorical cross-entropy loss, also known as softmax loss. Loss functions are used in optimization algorithms in deep learning to train the neural network weights using stochastic gradient descent. In the context of performance evaluation, categorical cross-entropy loss shows how well the neural network predicts a class, usually in the probability range between 0 and 1. The lower the loss, the better and more consistent the classification capability of the neural network. Categorical cross-entropy loss is suitable to be used in binary class classifications, and it is identical to binary cross-entropy loss. The formula for categorical cross-entropy loss is given by

$$Loss = -\sum_{i=1}^{i=N} y_i.ln\hat{y}_l \quad (3)$$

where N is the number of classes, y_i is the true value for the Nth class, and $\hat{y}_i$ is the predicted value for the Nth class. In this study, N = 2 since it is a binary classification task.

3.5. Baseline Model Hyperparameter Tuning

The dataset is split at a ratio of 4:1 on a region basis, with 80% (8 regions, 13,824 patches) for training and 20% (2 regions, 3456 patches) for testing. Various hyperparameters have been optimized in different HRNet model training sessions to identify the set of hyperparameters that yield the best performance. The tested hyperparameters are as follows:

i. Optimizers: Adam, Nadam, SGD, RMSprop;
ii. Learning rate: 0.00005, 0.0001, 0.0005, 0.001;
iii. Batch size: 8, 16, 24, 32.

Training is conducted using the Tesla P100 GPU provided by the Kaggle platform. Due to the limitations in the GPU memory provided and the large dataset size, the training dataset needs to be split further into two halves, and each training dataset subgroup is used to train HRNet for an equal number of epochs until convergence occurs.

The greedy approach strategy is adopted to identify a global optimal hyperparameter setting by evaluating each hyperparameter's effect on the segmentation performance locally. The default hyperparameter settings are the Adam optimizer, a learning rate of 0.0001, and a batch size of 8. The upper limit that is set is 0.0001 and, by using this option, possible convergence at local extrema is mitigated without exceeding the limit. In addition, the minimum learning rate of 0.00005 is selected to maintain a balance with the computing efficiency and avoid training excessively slowly without significant performance improvements. Batch sizes smaller than 8 are not examined because they are believed to contain less information, such as batches with 4 images. It is anticipated that smaller batch sizes will result in less informative updates during the training of the model. Furthermore, the limitation of the GPU VRAM size to 16 GB restricts our testing to a maximum of 32 images per batch.

Hyperparameter tuning using the greedy approach starts with the optimizer first, followed by the learning rate, and then finally the batch size. At every stage, only the hyperparameter of interest is varied, while the rest are kept at default values, and the hyperparameter of interest that produces the best mIoU metric is selected for each stage. The procedure is repeated until no more hyperparameters are left to test. Each locally selected optimal hyperparameter forms the globally optimal hyperparameter combination. The greedy approach reduces the minimum number of testing sessions required for tuning.

The hyperparameter combination that yields the best mIoU performance in the original HRNet network is chosen as the optimal baseline HRNet model used for forest segmentation tasks.

3.6. Baseline Model Modification Using Attention Mechanism

The baseline HRNet model is embedded with an attention mechanism with the aim to improve the baseline performance. The attention mechanism is inspired by the human visual system and its capability to observe and identify important and meaningful features in complex scenes. It diverts attention to the most important regions of an image and

disregards irrelevant parts. This allows the network to dynamically select these features by adjusting the weights adaptively based on the importance of the input.

The selected attention mechanism to be used in the baseline HRNet network is the convolutional block attention module (abbreviated as CBAM), as shown in Figure 3. Introduced by Woo et al. [31], CBAM is a simple and lightweight attention module that combines the strengths of both channel and spatial attention mechanisms together. The CBAM module has two sequential submodules, with the channel attention module placed first, followed by the spatial attention module.

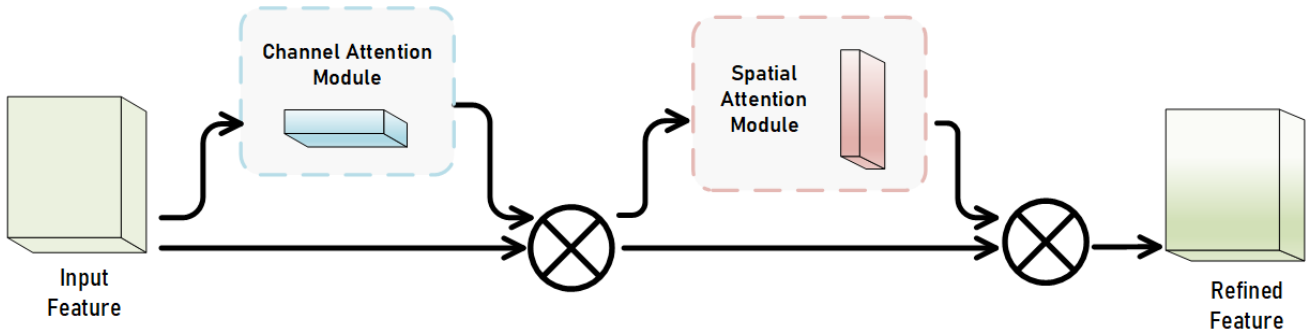

Figure 3. Architecture of a CBAM module.

The channel attention submodule in the CBAM module, as shown in Figure 4, identifies which channels (feature maps) contain the most important and relevant information in order to perform the classification task effectively. In the channel attention submodule, the spatial dimension of the input feature map is squeezed using two different pooling methods: average pooling and max pooling. Average pooling is used to learn the context of an entire channel, while max pooling is used to capture the distinctive features of a channel. Both of these pooled maps are fed to a shared MLP network with a hidden layer of a reduced channel size by reduction ratio r. The two resulting output features are summed and fed to the sigmoid activation function to produce the channel attention map, which is multiplied with the input feature map to produce a final output channel-enhanced feature map.

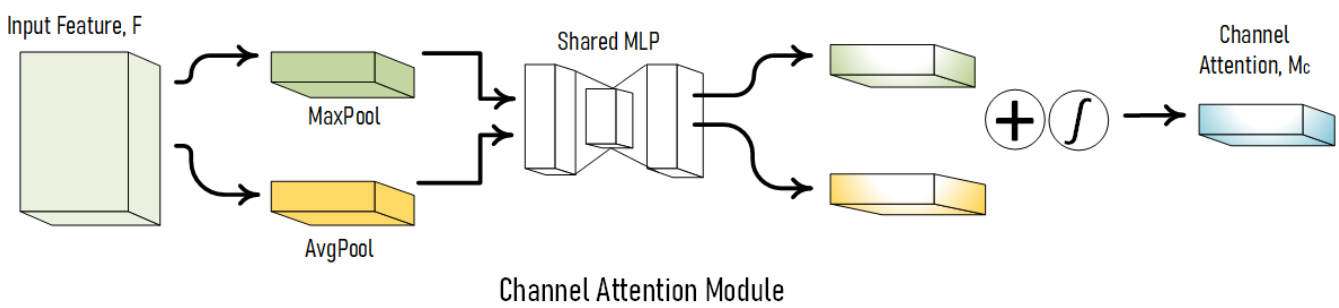

Figure 4. Channel attention submodule in CBAM.

The spatial attention submodule in the CBAM module, as shown in Figure 5, locates the points in the feature maps that are more important and have distinctive features that can be used for effective classification. In the spatial attention submodule, the input feature maps are both average-pooled and max-pooled along the channel axis to highlight the locations of important features, followed by the concatenation of both pooled maps. The concatenated feature maps then pass through a convolutional layer and are fed to a sigmoid activation function to produce the spatial attention map, which is multiplied with the input feature maps to produce the final output spatially enhanced feature map.

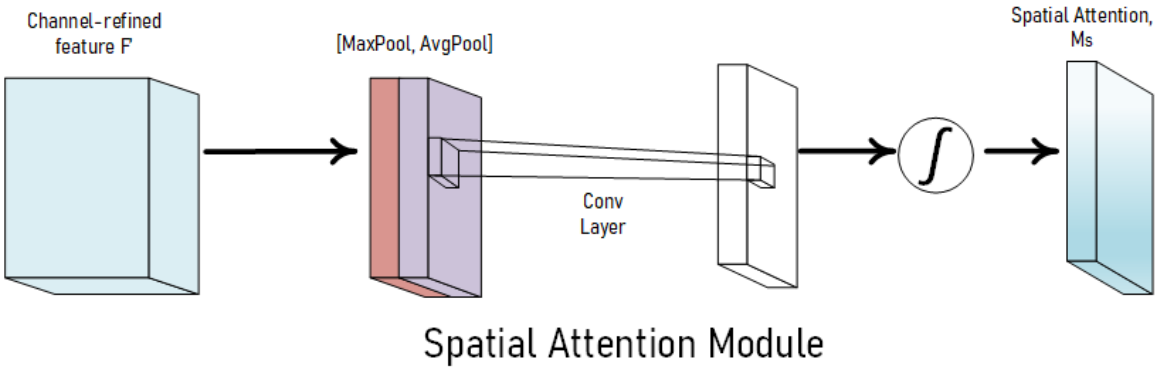

Figure 5. Spatial attention submodule in CBAM.

The embedding of the base HRNet network with the attention module begins with experimenting with a few different placement locations to insert the CBAM module. Seven different candidate locations are chosen, as shown in Figure 6, whereby the last convolutional unit for each stage in a resolution stream is replaced with the CBAM module. There are four CBAM module placement locations at the same resolution stream, and different stages of the HRNet network (denoted by red arrows) are experimented with to investigate the effect of the stage-wise placement of the CBAM module in the HRNet network, whether they affect the overall segmentation performance or not, and how early or late the CBAM module should be inserted into the baseline model. Another four sets of CBAM module placement locations at different resolution streams and different stages of the HRNet network (denoted by red arrows) are also experimented with to investigate the effect of the CBAM module's feature map's resolutions on the overall HRNet segmentation performance.

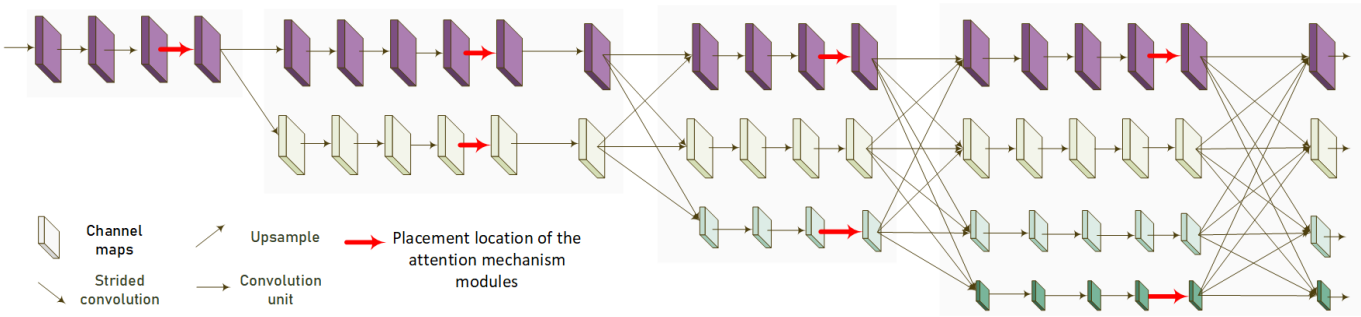

Figure 6. Candidate locations for placement of CBAM module in HRNet. The red arrow signifies the placement of the embedded attention mechanism into the HRNet.

After identifying the optimal placement location for the CBAM module in HRNet, the reduction ratio of the shared MLP network's hidden layer in the channel attention submodule is varied with values of 1, 2, 4, 8, 16, and 32. This is applied to observe the reduction in the number of channels at the hidden layer that is optimal for the shared MLP network to learn and capture channel features effectively, to produce better overall segmentation results. Finally, by taking the identified optimal reduction ratio of the channel attention submodule, the kernel size of the convolutional layer in the spatial attention module is varied with sizes of 1 pixel × 1 pixel, 3 pixels × 3 pixels, 5 pixels × 5 pixels, 7 pixels × 7 pixels, and 9 pixels × 9 pixels. These experiments are performed to investigate the optimal convolutional kernel size to effectively capture the spatial context of the feature maps without losing too much information or fitting too much information into a kernel.

The hyperparameters of the CBAM module that yield the best mIoU performance are selected as the optimal attention-embedded HRNet model used for forest segmentation tasks. This study suggests that some areas and channels of the forest can be emphasized

through a dual attention mechanism. Some of the feature maps, either in temporal or spatial channels, are more important for forest detection and vice versa. Therefore, more weights should be allocated to the feature maps that have a strong correlation with the object of interest, particularly the forest in this case. On the other hand, the background information, consisting of lakes, rivers, roads, and residential areas, should be allocated less weight, so that the model can distinguish forest and non-forest areas better. Moreover, some of the locations, such as the edges of the forest, should be given less attention so that the stitching of the image results in more accurate mapping.

3.7. Benchmarking Study

Benchmarking studies, comparing the chosen optimal baseline HRNet and attention-embedded HRNet model with other chosen candidate state-of-the-art models, are conducted to evaluate the potential of the proposed attention-embedded model as a viable choice to be used in conducting forest segmentation tasks. The same set of performance metrics used in this work so far are also applied to the candidate models.

3.8. Qualitative Analysis between Baseline and Attention-Embedded Model

After the quantitative analysis to select the optimal baseline and attention-embedded HRNet models, a qualitative analysis between the baseline and attention-embedded models is also performed by stitching the output image patches together to form a complete image of a tested land plot. Any changes or improvements to the segmentation quality between the baseline and the best attention-embedded version are analyzed and discussed. The models' effectiveness in the application of forest change monitoring is also analyzed and discussed.

4. Results and Discussion

4.1. Dataset Creation

The chosen satellite to obtain the satellite images is the Landsat-8 satellite, which was operational between the years 2013 and 2021, with a high spatial resolution of 30 m for the visible spectral band and more accurate sensors than previous Landsat generations, making it a good choice for usage in remote sensing applications. A set of ten locations across Malaysia with equally spaced years of 2016, 2018, and 2020 is chosen, with the coordinates shown in Table 1.

Table 1. Coordinates of the chosen locations for dataset preparation.

Dataset Type	Coordinate
Training	4°27′50.64″ N, 101°13′3.91″ E 4°28′20.89″ N, 101°11′0.90″ E 4°27′3.97″ N, 101°13′2.38″ E 4°25′55.53″ N, 101°11′38.52″ E 4°0′2.74″ N, 101°23′24.27″ E 4°14′16.67″ N, 101°19′30.57″ E 4°48′44.67″ N, 103°21′2.03″ E 3°49′30.93″ N, 103°18′52.84″ E
Testing	3°8′24.90″ N, 101°29′42.08″ E (Test Site A) 4°0′11.39″ N, 101°20′36.26″ E (Test Site B)

The raw satellite images are obtained at an eye altitude of about 7000 ft (2133.6 m) and have a resolution of 8064 pixels × 3584 pixels. The forest and non-forest areas are approximately distributed equally in a 50/50 ratio. The forest class consists of mainly tropical rainforests around Malaysia. Due to substantial spatial variations in their environmental conditions, tropical rainforests are regarded as the most intricate terrestrial ecosystems [32]. The measurement of canopy height is a crucial factor in assessing the functional aspects of forest ecosystems [33]. According to the Forest Survey of India [34], there are five categories of forest: very dense forest, encompassing lands with tree cover, including mangroves,

exhibiting a canopy density of 70% and above; moderately dense forest, including areas with tree cover showcasing a canopy density ranging from 40% to 70%; open forest, covering lands with a canopy density between 10% and 40%; scrub, representing forested lands with poor tree growth, primarily featuring small or stunted trees, with a canopy density less than 10%; and non-forest, encapsulating areas not falling within the specified forest classifications. The tropical rainforests in Malaysia can be divided into two classes: mixed forest and single dominant forest [35]. Mixed forests, such as the *Dipterokarpa* forest, are characterized by a diverse array of plant species from different families. On the other hand, single dominant forests, exemplified by the lime single dominant forest (*Dryobalanops aromatica*), are dominated by a single species.

The non-forest class has larger variation, consisting of oil palm plantations, water bodies, roads and highways, clear-cut forests, and man-made buildings. A set of corresponding binary masks as the ground truths are manually annotated as accurately and consistently as possible for every raw satellite image using the Program GIMP-2 software version 2.10.22 (GIMP Team, Kernersville, NC, USA, https://www.gimp.org/) (white: forest, bit 1; black: non-forest, bit 0), so that the validation and testing process is effective. The images and masks are all sliced into smaller patches of 224 pixels × 224 pixels resolution to be fed into the tested networks, as the models cannot process large-sized images at one time, leading to extraordinarily high GPU memory consumption. For the training dataset, the raw satellite images have three spectral channels consisting of red, blue, and green, while the ground truth masks have 2 channels representing forest (white) and non-forest (black) classes. The training dataset is split at an 80/20 ratio, with 80% for training and 20% for testing, with the detailed dataset division shown in Table 1. Due to the large size of the annotated dataset and the limited GPU memory of the Tesla P100 GPU, the training dataset is split further into two smaller, equally sized groups, where each training group is trained separately with an equal number of epochs and the model weights of one training group carried forward to the next training group.

4.2. Baseline Default Hyperparameters

The optimization of the baseline HRNet model's hyperparameters is conducted using the greedy approach, with the optimizers varied first, followed by the learning rate, and finally the batch size. The default hyperparameter settings are the Adam optimizer, a 0.0001 learning rate, and a batch size of 8.

Table 2 shows the performance summary of the different optimizers used while keeping the learning rate and batch size at the default settings. The chosen number of epochs for this training is 220 due to the slow convergence behavior of the SGD optimizer. The results show that the RMSprop optimizer provides the best performance, with a mIoU score of 83.73% and an accuracy score of 91.16%. The descending order of optimizer performance in terms of mIoU is RMSprop, Nadam, Adam, and SGD.

Table 2. Baseline HRNet performance using different optimizers.

Optimizer	Accuracy (%)	mIoU (%)	Loss
Adam	87.91	78.43	0.8799
Nadam	90.93	83.35	0.7287
SGD	85.80	75.13	0.3500
RMSprop	91.16	83.73	0.7970

The SGD optimizer performs gradient updates on the network's weights after each randomly selected minibatch based on a fixed learning rate [36]. The gradient updates in the SGD optimizer may not be effective in certain points of the data samples and may have difficulty in reaching the global optimum due to the fixed learning rate. One advantage of SGD to note is its lower loss score, indicating that the model could generalize better and

make predictions consistently with fewer errors; however, the SGD optimizer also requires a greater number of epochs to reach convergence compared to the other studied optimizers.

The Adam and Nadam optimizers are adaptive learning rate algorithms for the updating of network weights [36]. They require little tuning and have faster convergence. Nadam is a modification of Adam with Nesterov's accelerated gradient. Both outperform SGD, with Nadam performing better. However, Nadam has a longer training time due to the small batch size. Occasionally, Adam and Nadam show spikes in the learning curve, making consistent performance difficult. The RMSprop optimizer adapts the learning rates using a decaying average of squared gradients, resulting in a smoother learning curve compared to Adam and Nadam [36]. It also converges faster with fewer epochs and outperforms other optimizers.

The RMSprop optimizer outperforms all the optimizers studied and is chosen as the optimal optimizer for the first stage of hyperparameter tuning. Table 3 shows the performance summary with the use of different learning rates while using RMSprop as the optimal optimizer and the default batch size. Training from this point on reveals that the model converges at around only 100 epochs. Results show that the learning rate of 0.0005 performs better, with a mIoU score of 83.90% and an accuracy score of 91.26%. The descending order of learning rates in terms of mIoU performance is 0.0005, 0.001, 0.0001, and 0.00005.

Table 3. Baseline HRNet performance using different learning rates.

Learning Rate	Accuracy (%)	mIoU (%)	Loss
0.00005	90.53	82.66	0.7407
0.0001	90.55	82.72	0.7009
0.0005	91.26	83.90	0.7250
0.001	90.83	83.16	0.6504

The RMSprop optimizer adjusts the learning rates, but a suitable starting rate must be strategically chosen. Large rates lead to fast convergence but may lead to a suboptimal solution. Small rates require more epochs but allow for smoother gradient updates. Rates lower than 0.0005 result in a lower mIoU and higher losses, indicating insufficient updates. Rates higher than 0.0005 also result in a lower mIoU but lower loss, suggesting convergence to a suboptimal solution. In this study, a learning rate of 0.001 is found to be less effective compared to 0.0005. This suggests that, given the attention mechanism, a lower learning rate per parameter update is necessary for fine tuning. The reduced step size in the update process can be attributed to the challenges associated with learning diverse environmental features, encompassing not only forests but also plantations and bushy areas, commonly found in Southeast Asian countries. An optimal learning rate of 0.0005, identified from the results, is selected for the second stage.

Table 4 shows the performance summary for the use of different batch sizes while using the optimal RMSprop optimizer and an optimal learning rate of 0.0005. Training the baseline model using a batch size of 32 yields the best outcome, with a mIoU score of 84.84% and an accuracy score of 91.81%. The decreasing order of batch size in terms of mIoU performance is 32, 24, 16, and 8. A batch size of 24 appears to be the least effective for our specific case, primarily due to the dependence on the dataset used for our South East Asia forest mapping objective. It is important to note that a larger batch size does not necessarily lead to improved performance, as evidenced by the findings of Wu and He in their study on group normalization [37]. Therefore, it is crucial to carefully select the batch size based on the specific application at hand.

Table 4. Baseline HRNet performance using different batch sizes.

Batch Size	Accuracy (%)	mIoU (%)	Loss
8	91.26	83.90	0.7250
16	91.78	84.78	0.7890
24	91.62	84.51	0.7054
32	91.81	84.84	0.6142

The batch size determines the number of training samples used for the estimation and updating of gradients in the model. Larger batch sizes generally result in better performance by using more data samples at once to estimate error gradients and update model weights, leading to a better fit. However, larger batch sizes require more memory and more predictions before reaching the final estimate. Smaller batch sizes require fewer data samples at once to update the model weights, resulting in low memory consumption of the CPU/GPU. However, they may yield less accurate error gradient estimates and require more frequent and noisy updates to model weights. The best performance is observed with a batch size of 32, indicating that larger data samples help to make more accurate error gradient estimations. Therefore, a batch size of 32 is selected for this study.

4.3. Optimal Baseline Model's Hyperparameters

From the results obtained using the greedy approach, the optimal combination of hyperparameters for the baseline HRNet model is the RMSprop optimizer, a 0.0005 learning rate, and a batch size of 32.

4.4. Location Placement of CBAM Module

Table 5 shows the segmentation performance with four different stage placements of the CBAM module in the HRNet model. These four locations are located at the highest resolution stream of 56 pixels × 56 pixels. The last convolutional unit of every HRNet stage is configured to be replaced with the CBAM module, since the feature maps at this point will undergo multi-resolution fusion with other resolution streams before entering a new stage, boosting the feature maps from different resolutions with the attention-enhanced feature maps. The CBAM module used has a reduction ratio of 8 for the channel attention's shared MLP network and a kernel size of 7 pixels × 7 pixels for the spatial attention's convolutional layer. Placing the CBAM module at stage 2 of the HRNet model yields the best performance, with an accuracy score of 92.24% and mIoU score of 85.58%, followed by stage 1, stage 3, and stage 4. Placing the CBAM module too early in the baseline model might cause the attention-enhanced feature map to be lost as it propagates through the later stages of HRNet, creating a vanishing gradient problem. Alternatively, placing the CBAM module too late might cause the model not to be able to extract sufficient information needed to improve the segmentation performance, as the attention-enhanced feature map is fused too late and unable to distribute the attentive features across the hidden layers of the HRNet model at a sufficient depth.

Table 5. HRNet + CBAM performance when placing CBAM module at different stages of HRNet.

Stage Location	Accuracy (%)	mIoU (%)	Loss
1	91.77	84.77	0.7258
2	92.24	85.58	0.6770
3	91.37	84.09	0.6908
4	91.25	83.89	0.7528

Table 6 shows the segmentation performance with four different resolution stream placements of the CBAM module in the HRNet model. The results show that placing the

CBAM module at a 28 pixels × 28 pixels resolution stream at stage 2 HRNet produces better segmentation performance, with an accuracy score of 91.87% and a mIoU score of 84.94%. The performance when the CBAM module is placed at the 14 pixels × 14 pixels (stage 3 HRNet) and 7 pixels × 7 pixels (stage 4 HRNet) resolution streams shows lower scores when compared to the 28 pixels × 28 pixels resolution stream. However, when it is compared with the previous results of placing the CBAM module at stage 3's and stage 4's resolution streams, respectively, they perform better than their counterparts. This means that boosting the lower-resolution feature maps with an attention mechanism at the later stages of HRNet is more effective than boosting the higher-resolution ones. The reason that placing the CBAM module at the 28 pixels × 28 pixels resolution stream leads to higher performance compared to the 56 pixels × 56 pixels resolution stream could be due to the optimal stage placement, based on the previous results, since the 28 pixels × 28 pixels resolution stream only begins at stage 2 of HRNet.

Table 6. HRNet + CBAM performance when placing CBAM module in different resolution streams of HRNet.

Resolution Stream	Accuracy (%)	mIoU (%)	Loss
56 pixels × 56 pixels	91.77	84.77	0.7258
28 pixels × 28 pixels	91.87	84.94	0.7421
14 pixels × 14 pixels	91.81	84.84	0.7047
7 pixels × 7 pixels	91.84	84.89	0.6838

Placing the CBAM module at stage 2's 56 pixels × 56 pixels resolution stream, the one with the highest resolution in the HRNet model, produces the best performance and it is chosen as the optimal architecture for the attention-embedded HRNet model (HRNet + CBAM).

4.5. Reduction Ratio Tuning of Channel Attention Submodule's Shared MLP Network

Table 7 shows the performance results when varying the reduction ratio r of the channel attention submodule's shared MLP network in the CBAM module of the optimal HRNet + CBAM architecture. A reduction ratio of eight gives the best results, with an accuracy score of 92.24% and a mIoU score of 85.58%, followed by 16, 4, and 2. One of the main purposes of reducing the number of channels at the shared MLP network's hidden layer is to reduce the computing overhead of the CBAM module. By reducing the number of channels with a reduction ratio, r, of the appropriate value, the channel attention submodule can effectively extract channels of high importance and learn the essential features from the selected channels. Selecting too few channels (high reduction ratio) could hinder the shared MLP network's ability to extract sufficient information from the vital channels. Alternatively, selecting too many channels (low reduction ratio) also causes the shared MLP network to learn information from too many channels and be unable to focus on and identify the vital ones. The reduction ratio $r = 2$ is chosen as the optimal hyperparameter for the CBAM's channel attention submodule.

Table 7. HRNet + CBAM performance using different reduction ratios, r, in channel attention submodule.

Reduction Ratio, r	Accuracy (%)	mIoU (%)	Loss
2	91.75	84.73	0.6706
4	91.98	85.12	0.6713
8	92.24	85.58	0.6770
16	92.08	85.31	0.6171

4.6. Kernel Size Tuning of Convolutional Layer in Spatial Attention Submodule

Table 8 shows the performance results when varying the kernel sizes of the spatial attention submodule's convolutional layer in the CBAM module of the optimal HRNet + CBAM architecture and with the optimal reduction ratio $r = 2$. A kernel size of 7 pixels × 7 pixels yields the best segmentation performance, with an accuracy score of 92.24% and a mIoU score of 85.58%, followed by 9 pixels × 9 pixels, 5 pixels × 5 pixels, and 3 pixels × 3 pixels. The kernel sizes of a convolutional layer in a deep learning model determine the sizes of pixel values processed at a time during a single step of a convolution operation. Kernel sizes that are too small could hinder the spatial attention submodule's ability to view the larger spatial context of neighboring pixels. Alternatively, kernel sizes that are too large could render the spatial attention submodule unable to effectively pinpoint spatially where the important information is across the spatial dimensions of the feature map. A kernel of 7 pixels × 7 pixels is chosen as the optimal kernel size for the CBAM's spatial attention submodule.

Table 8. HRNet + CBAM performance using different kernel sizes in spatial attention submodule.

Kernel Size	Accuracy (%)	mIoU (%)	Loss
3 pixels × 3 pixels	91.10	83.63	0.7602
5 pixels × 5 pixels	91.28	83.94	0.7090
7 pixels × 7 pixels	92.24	85.58	0.6770
9 pixels × 9 pixels	92.14	85.40	0.6074

4.7. Optimal Attention-Embedded HRNet Model and Qunatitative Comparison with Baseline HRNet Model

The network diagram of the proposed optimal attention-embedded HRNet model using the CBAM module is shown in Figure 7, with the CBAM module replacing the last convolutional unit at the highest-resolution stream of HRNet's second stage before the fusion portion.

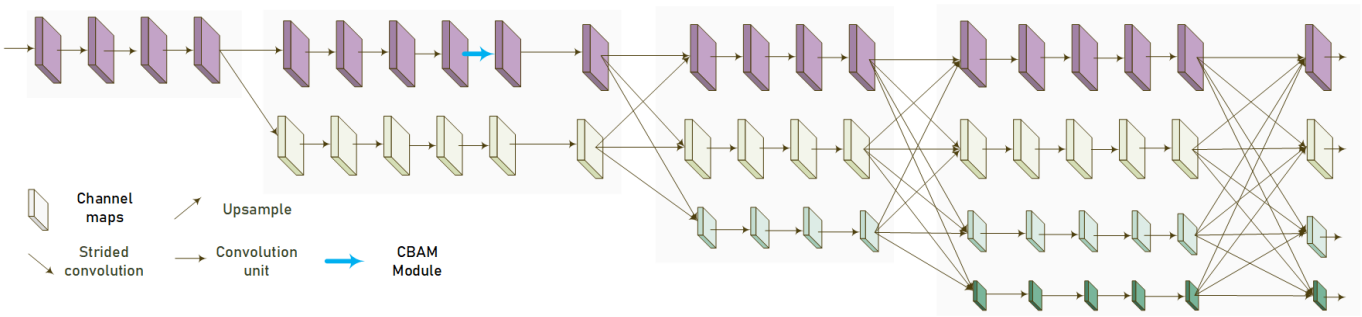

Figure 7. Proposed attention-embedded HRNet model using CBAM module.

The summarized settings for the optimal attention-embedded HRNet model using the CBAM module are as follows:

i. CBAM module placement: stage 2, 56 pixels × 56 pixels resolution stream;
ii. Reduction ratio r (channel attention): 8;
iii. Kernel size (spatial attention): 7 pixels × 7 pixels

Table 9 summarizes the performance results between the optimal baseline HRNet model and the optimal attention-embedded HRNet model.

Table 9. Performance comparison between optimal HRNet model and optimal HRNet + CBAM model.

Model	Accuracy (%)	mIoU (%)	Loss
HRNet	91.81	84.84	0.6142
HRNet + CBAM	92.24	85.58	0.6770

From Table 9, there is an improvement of 0.43% for the accuracy score and an improvement of 0.74% for the mIoU score between the HRNet and HRNet + CBAM models. Although the accuracy score improvement is lower, the larger mIoU improvement shows that the CBAM module embedded into the HRNet model increases the overlapping between the ground truth and the model's prediction, which should give better and more consistent segmentation quality. The minimal increase in loss of 0.0628 between the HRNet and HRNet + CBAM may imply that the replacement of the original HRNet convolutional unit's parameters with the CBAM module's parameter introduces slight difficulties in fitting the problem to the newly altered model. Nevertheless, the larger mIoU improvement of the mIoU score offsets the increment in loss.

4.8. Benchmarking Study

To evaluate the effectiveness of the proposed attention-embedded HRNet model, several candidate state-of-the-art semantic segmentation architectures are tested and compared quantitatively with the proposed model, namely U-Net, SegNet, and FC-FC-DenseNet. Table 10 summarizes the benchmarking tests that are performed using the optimal hyperparameter configurations.

Table 10. Benchmarking study of HRNet and HRNet + CBAM models with other state-of-the-art semantic segmentation networks.

Model	Accuracy (%)	mIoU (%)	Loss
HRNet	91.81	84.84	0.6142
FC-DenseNet	82.56	70.01	1.2159
SegNet	91.61	84.49	0.6071
U-Net	91.24	83.87	0.8283
HRNet + CBAM	92.24	85.58	0.6770

The tests show that SegNet and U-Net perform comparatively well with the baseline HRNet model, with accuracy and mIoU performance scores slightly lower than those of HRNet. The loss score for U-Net is higher than that for HRNet, while the loss score for SegNet is slightly lower than that of HRNet. FC-DenseNet performs the worst among the rest of the tested models, with an accuracy score of 82.56%, a mIoU score of 70.01%, and a loss score of 1.2159. FC-DenseNet also has the longest training time compared to the rest of the models, and the batch size has to be reduced to accommodate the limited GPU resources due to the huge number of feature maps produced in the hidden layers of the FC-DenseNet network. Nonetheless, based on the mIoU scores, the baseline HRNet model still outperforms other tested state-of-the-art models, and the addition of a CBAM module into the baseline model leads to a further improvement in the segmentation performance in terms of the accuracy and mIoU scores.

4.9. Qualitative Analysis between Baseline and Attention-Embedded Models

In addition to the quantitative analysis between models using performance metrics, the models are also analyzed qualitatively by observing the prediction outputs from both models of the test dataset. Figure 8 shows the selected samples of raw satellite images, its ground truth masks, its predictions from HRNet, and its predictions from HRNet + CBAM

from the test dataset, formed by piecing nine neighboring patches of three selected sample locations from each test location together.

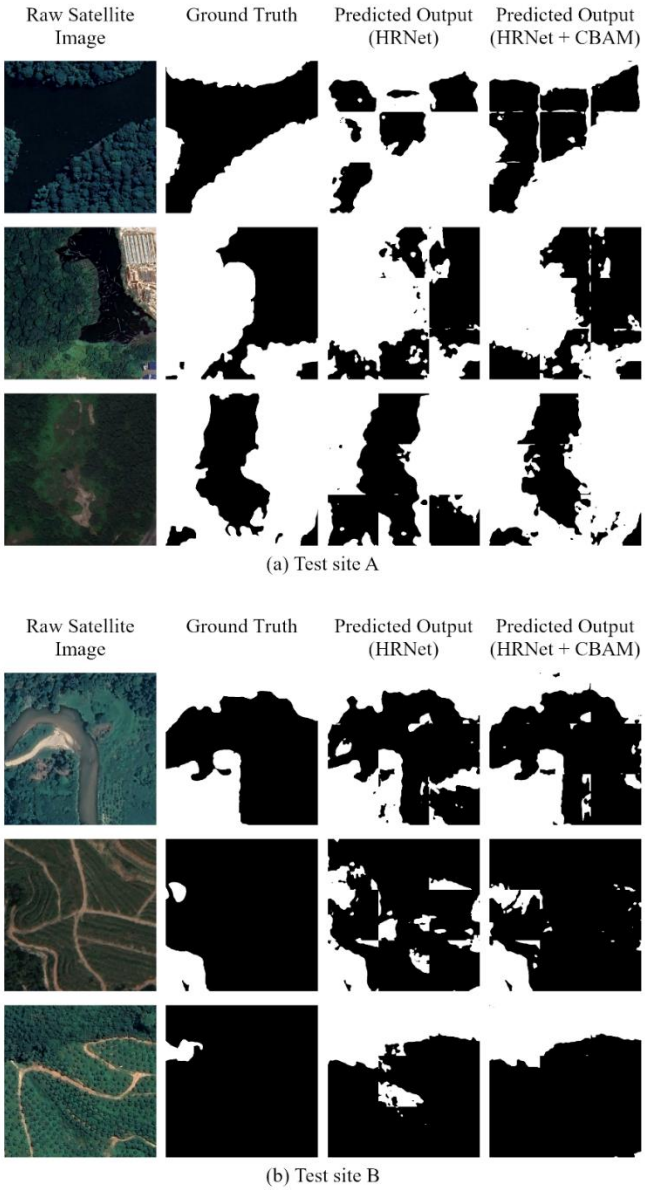

Figure 8. Raw satellite image, ground truth, and HRNet and HRNet + CBAM models' output predictions for selected location samples in testing dataset (nine 224 pixels × 224 pixels neighboring patches stitched together).

As observed from the samples, the HRNet + CBAM model produces a more accurate and cleaner segmentation output compared to the HRNet model, showing that the attention mechanism helps in the training of the network to perform forest segmentation tasks. From test site A, which is located around suburban areas, the HRNet model has some difficulty in properly segmenting the water bodies, generating many false positives and patch-like boundaries between forest and non-forest areas, while the HRNet + CBAM model produces a cleaner segmentation boundary for the water bodies and produces fewer false positives, although minimal line-shaped false positives are still present along the boundaries where the images are sliced. Some false negatives at certain forest areas in the HRNet model are also prevented with the addition of the CBAM module.

From test site B, which is located around rural areas, the HRNet model tends to generate more patches of false positives in non-forest areas; in this case, the forest areas

were cleared for plantation purposes. With the addition of the CBAM module into the HRNet model, the HRNet + CBAM model produces fewer false positive classifications at non-forested areas such as newly created plantations and green-colored flatlands. However, both models still have a slight difficulty in classifying certain non-forest areas in test site B with some vegetation present. Nonetheless, the cleaner segmentation output represents a performance improvement with the CBAM module. Overall, the addition of the attention mechanism into the baseline model yields a cleaner and more useful forest segmentation output with fewer salt-and-pepper-like false classifications.

The base model is HRNet, which is a model developed with the aim of producing good detection for high-resolution images. It has inherently multiscale capabilities, whereby the high-resolution scale is carried over throughout the network while adding a smaller scale for each stage of the network. Therefore, the model itself should be able to cater to various resolutions of forest imaging as long as it is trained with the selected resolution. At present, our dataset comes from countries in Southeast Asia that have lush green forests, which makes it suitable for this particular case. However, the model is capable of learning a variety of forest types, given that it is trained for such a situation. Nonetheless, the proposed model is not designed to segment non-tropical forest such as savanna and boreal forests.

4.10. Forest Area Change Detection

Figure 9 shows the raw satellite image outputs of two different test sites from the years 2016, 2018, and 2020, and their predicted segmentation outputs are taken from the HRNet + CBAM model. For test site A, the HRNet + CBAM model is able to detect a slight increase in non-forest areas between the years 2018 and 2020, most likely deforested to be repurposed for the construction of buildings and facilities. For test site B, the HRNet + CBAM model can detect a significant increase in non-forest areas between the years 2016 and 2018, deforested to be repurposed as plantation regions. The model also can differentiate and classify leftover greenery in the plantation areas as non-forest areas from forest areas, with small instances of false classifications as forest areas. By comparing the segmentation output between years and performing pixel-wise operations, a forest change map can be created between years, with bits 1 to 0 representing deforestation occurrences and bits 0 to 1 representing reforestation occurrences. Overall, the HRNet + CBAM model is proven to be effective in detecting forest changes using satellite images.

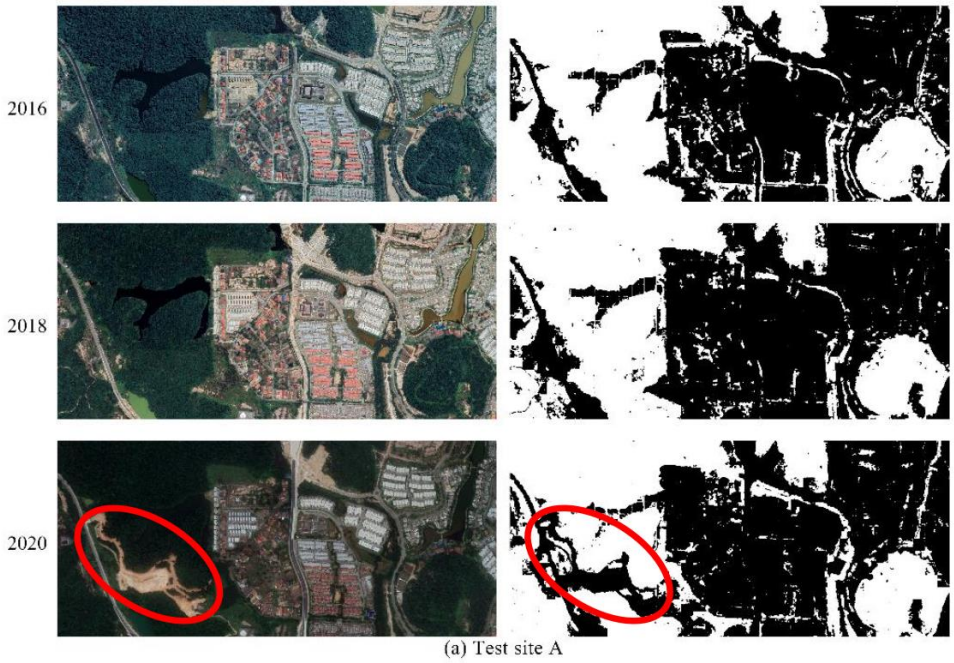

Figure 9. *Cont.*

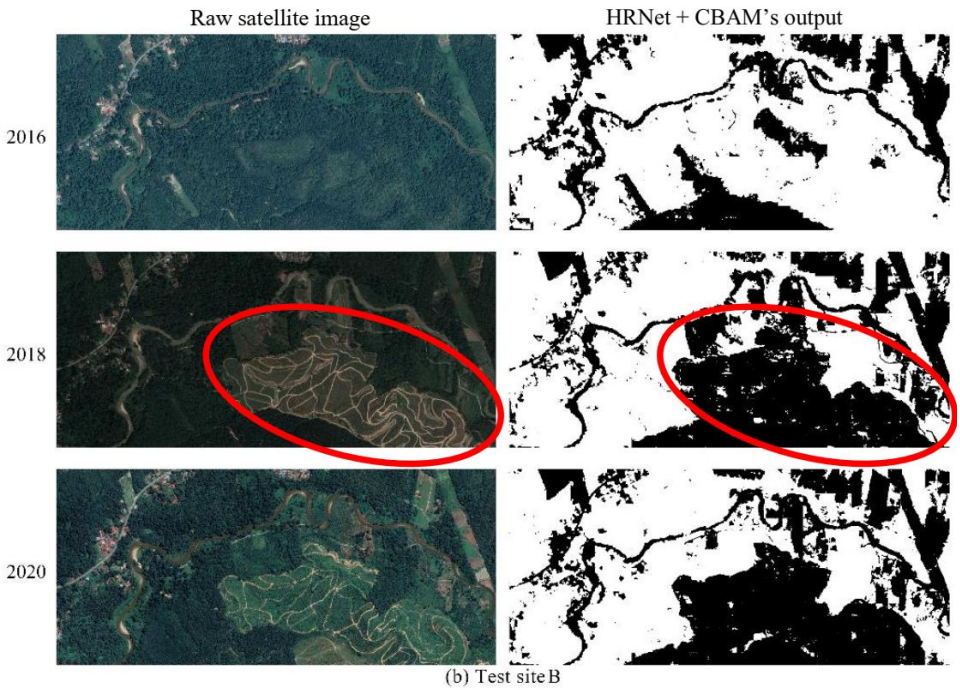

Figure 9. Raw satellite images and HRNet + CBAM model's output predictions for locations in testing dataset.

4.11. Study Limitations

There are several limitations identified in this work. The first limitation is that the annotated dataset may not represent the entire population of various forest and non-forest types, with the possibility that a few types of non-forest areas are absent from the dataset, such as tea plantations and burnt areas due to forest fires. The performance of the model, which works optimally with the curated dataset, might face degradation under new area types that are unseen during training, such as those mentioned earlier. Another limitation is that the HRNet + CBAM model still has some difficulties in classifying flat green landscapes as non-forest areas. Some types of vegetation in these areas might have a slight resemblance to forest trees, causing the model to mistakenly and incorrectly identify them as forest areas. A huge limitation is the limited GPU memory resources used in model training. The Tesla P100 GPU has a memory capacity of 16 GB VRAM, which is insufficient to fit the entire created dataset into the GPU. This was also noted in a previous study by Ru et al. [38], who used Google Colab for forest segmentation. They stated that the limited RAM and GPU runtime caused difficulties in the forest segmentation training process. The main objective of this study was to develop a forest monitoring system based on the novel architecture of the attention-based HRNet using Landsat-8 satellite imagery. However, due to GPU constraints, it is not possible to make significant extensions, such as to new forests, burned forests, and forest density monitoring systems. For this reason, this study only focuses on detecting forests, to avoid an excessive burden on the GPU. The training dataset must be split into two smaller groups and trained in two separate sessions with an equal number of epochs, with the final model weights from the first training group carried over to the second training group. This is equivalent to training the network in smaller batch sizes, although it differs slightly in the sense that the model trained using this method with one training group cannot see and learn another training group simultaneously. The performance would very likely be improved if the entire dataset could be fit into the GPU in a single training session.

5. Conclusions and Future Works

An optimal attention-embedded high-resolution segmentation network, HRNet + CBAM, has been developed and tested in performing classification tasks between forest and non-forest areas in Malaysia using Landsat-8 images with temporally spaced years, available online. A dataset comprising raw satellite images of ten locations within Malaysia for the years 2016, 2018, and 2020 has been manually annotated and has been shown to be effective in training the model to sufficiently learn and classify forest and non-forest areas. The dataset is further split at an 80/20 ratio, with 80% for training and 20% for testing. An optimal baseline HRNet model used for forest segmentation tasks has been determined using the greedy approach of tuning the following hyperparameters: optimizer, learning rate, and batch size. The performance of the baseline HRNet model is 91.81% for accuracy, 84.84% for mIoU, and 0.6142 for loss. An optimal attention-embedded HRNet model using the CBAM module has been developed and has improved the baseline performance to 92.24% accuracy, 85.58% mIoU, and 0.6770 loss. Benchmarking studies with other state-of-the-art models such as U-Net, SegNet, and FC-DenseNet have shown that both HRNet and HRNet + CBAM outperform other models in terms of accuracy and mIoU. A qualitative image comparison of the predicted outputs between HRNet and HRNet + CBAM has also shown improved segmentation quality for the HRNet + CBAM variant, with fewer false classifications between forest and non-forest areas. For future works in this field, a larger dataset with more variations in the non-forest class can be considered to improve the sample population of the dataset when compared to the entire population variation of geographical landscapes in Malaysia. Other forms and variations of attention mechanisms can also be embedded into several other state-of-the-art architectures to test their effectiveness in different types of image segmentation networks. A higher-end GPU can be considered to accommodate the large dataset size, or other alternative pipelining techniques of loading large datasets into the limited GPU memory can be investigated. Alternatively, a dataset with a slightly lower resolution can be considered to reduce the memory usage.

Author Contributions: S.V.L. and M.A.Z., conceptualization, formal analysis and methodology. S.V.L., M.A.Z., A.S., A.H.S. and S.R.A., writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Universiti Kebangsaan Malaysia under Dana Padanan Kolaborasi with grant number DPK-2023-006 and the Asia-Pacific Telecommunity under the Extra Budgetary Contribution from the Republic of Korea Fund with grant number KK-2022-026.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to the restriction by the our collaborator.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Worm, B.; Barbier, E.B.; Beaumont, N.; Duffy, J.E.; Folke, C.; Halpern, B.S.; Jackson, J.B.C.; Lotze, H.K.; Micheli, F.; Palumbi, S.R.; et al. Impacts of Biodiversity Loss on Ocean Ecosystem Services. *Science* **2006**, *314*, 787–790. [CrossRef] [PubMed]
2. Bonan, G.B. Forests and Climate Change: Forcings, Feedbacks, and the Climate Benefits of Forests. *Science* **2008**, *320*, 1444–1449. [CrossRef] [PubMed]
3. Schulze, K.; Malek, Ž.; Verburg, P.H. Towards Better Mapping of Forest Management Patterns: A Global Allocation Approach. *For. Ecol. Manag.* **2019**, *432*, 776–785. [CrossRef]
4. Ministry of Energy Natural Resource. *Sixth National Report of Malaysia to the Convention on Biological Diversity*; Ministry of Energy Natural Resource: Putrajaya, Malaysia, 2019. Available online: https://www.cbd.int/doc/nr/nr-06/my-nr-06-en.pdf (accessed on 15 January 2023).
5. Potapov, P.; Hansen, M.C.; Pickens, A.; Hernandez-Serna, A.; Tyukavina, A.; Turubanova, S.; Zalles, V.; Li, X.; Khan, A.; Stolle, F.; et al. The Global 2000–2020 Land Cover and Land Use Change Dataset Derived From the Landsat Archive: First Results. *Front. Remote Sens.* **2022**, *3*, 856903. [CrossRef]
6. Kementerian Sumber Asli Alam Sekitar Iklim dan Perubahan National Forest Monitoring System—REDD PLUS. Available online: https://redd.ketsa.gov.my/mrvframework/national-forest-monitoringsystem/ (accessed on 15 January 2023).
7. Nathiratul Athriyah, A.M.; Muhammad Amir, A.K.; Zaki, H.F.M.; Zulkifli, Z.A.; Hasbullah, A.R. Incremental Learning of Deep Neural Network for Robust Vehicle Classification. *J. Kejuruter.* **2022**, *34*, 843–850. [CrossRef] [PubMed]

8. LeCun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
9. Nafea, M.M.; Tan, S.Y.; Jubair, M.A.; Abd, M.T. A Review of Lightweight Object Detection Algorithms for Mobile Augmented Reality. *Int. J. Adv. Comput. Sci. Appl.* **2022**, *13*, 536–546. [CrossRef]
10. Elizar, E.; Zulkifley, M.A.; Muharar, R.; Hairi, M.; Zaman, M. A Review on Multiscale-Deep-Learning Applications. *Sensors* **2022**, *22*, 7384. [CrossRef]
11. Wang, J.; Sun, K.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X.; et al. Deep High-Resolution Representation Learning for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 3349–3364. [CrossRef]
12. Wagner, F.H.; Dalagnol, R.; Silva-Junior, C.H.; Carter, G.; Ritz, A.L.; Hirye, M.C.; Ometto, J.P.H.B.; Saatchi, S. Mapping Tropical Forest Cover and Deforestation with Planet NICFI Satellite Images and Deep Learning in Mato Grosso State (Brazil) from 2015 to 2021. *Remote Sens.* **2022**, *15*, 521. [CrossRef]
13. LeCun, Y.; Kavukcuoglu, K.; Farabet, C. Convolutional Networks and Applications in Vision. In Proceedings of the ISCAS 2010–2010 IEEE International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems, Paris, France, 30 May–2 June 2010; pp. 253–256.
14. Phung, V.H.; Rhee, E.J. A High-Accuracy Model Average Ensemble of Convolutional Neural Networks for Classification of Cloud Image Patches on Small Datasets. *Appl. Sci.* **2019**, *9*, 4500. [CrossRef]
15. Dong, L.; Du, H.; Mao, F.; Han, N.; Li, X.; Zhou, G.; Zhu, D.; Zheng, J.; Zhang, M.; Xing, L.; et al. Very High Resolution Remote Sensing Imagery Classification Using a Fusion of Random Forest and Deep Learning Technique—Subtropical Area for Example. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 113–128. [CrossRef]
16. Khan, S.H.; He, X.; Porikli, F.; Bennamoun, M. Forest Change Detection in Incomplete Satellite Images with Deep Neural Networks. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 5407–5423. [CrossRef]
17. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *39*, 640–651.
18. Tao, R.; Gavves, E.; Smeulders, A.W.M. Siamese Instance Search for Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 1420–1429.
19. Chen, H.; Wu, C.; Du, B.; Zhang, L.; Wang, L. Change Detection in Multisource VHR Images via Deep Siamese Convolutional Multiple-Layers Recurrent Neural Network. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 2848–2864. [CrossRef]
20. Guo, Y.; Long, T.; Jiao, W.; Zhang, X.; He, G.; Wang, W.; Peng, Y.; Xiao, H. Siamese Detail Difference and Self-Inverse Network for Forest Cover Change Extraction Based on Landsat 8 OLI Satellite Images. *Remote Sens.* **2022**, *14*, 627. [CrossRef]
21. Caye Daudt, R.; Le Saux, B.; Boulch, A. Fully Convolutional Siamese Networks for Change Detection. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 4063–4067.
22. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *arXiv* **2018**, arXiv:1802.02611.
23. Andrade, R.B.; Costa, G.A.O.P.; Mota, G.L.A.; Ortega, M.X.; Feitosa, R.Q.; Soto, P.J.; Heipke, C. Evaluation of Semantic Segmentation Methods for Deforestation Detection in the Amazon. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *XLIII-B3*, 1497–1505. [CrossRef]
24. Ferreira, M.P.; Lotte, R.G.; D'Elia, F.V.; Stamatopoulos, C.; Kim, D.-H.; Benjamin, A.R. Accurate Mapping of Brazil Nut Trees (*Bertholletia excelsa*) in Amazonian Forests Using WorldView-3 Satellite Images and Convolutional Neural Networks. *Ecol. Inform.* **2021**, *63*, 101302. [CrossRef]
25. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015; Part III 18. Springer International Publishing: Cham, Switzerland, 2015.
26. Abdani, S.R.; Zulkifley, M.A.; Mamat, M. U-Net with Spatial Pyramid Pooling Module for Segmenting Oil Palm Plantations. In Proceedings of the IEEE International Conference on Artificial Intelligence in Engineering and Technology, IICAIET 2020, Kota Kinabalu, Malaysia, 26–27 September 2020; pp. 1–5.
27. Bragagnolo, L.; da Silva, R.V.; Grzybowski, J.M.V. Amazon Forest Cover Change Mapping Based on Semantic Segmentation by U-Nets. *Ecol. Inform.* **2021**, *62*, 101279. [CrossRef]
28. Cheng, Z.; Fu, D. Remote Sensing Image Segmentation Method Based on HRNET. In Proceedings of the 2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 6750–6753. [CrossRef]
29. Li, M.; Zhu, M.; He, Y.; Shu, J.; Li, P.; Hou, A.; Zheng, Z.; Zhou, G.; Li, Z.; Wang, Z.; et al. Classification of Surface Natural Resources Based on Hr-Net and Dem. *Int. Geosci. Remote Sens. Symp.* **2021**, *2021*, 4988. [CrossRef]
30. Maxwell, A.E.; Warner, T.A.; Guillén, L.A. Accuracy Assessment in Convolutional Neural Network-Based Deep Learning Remote Sensing Studies—Part 1: Literature Review. *Remote Sens.* **2021**, *13*, 2450. [CrossRef]
31. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
32. Konishi, S.; Tani, M.; Kosugi, Y.; Takanashi, S.; Sahat, M.M.; Nik, A.R.; Niiyama, K.; Okuda, T. Characteristics of Spatial Distribution of Throughfall in a Lowland Tropical Rainforest, Peninsular Malaysia. *For. Ecol. Manag.* **2006**, *224*, 19–25. [CrossRef]
33. Adrah, E.; Wan Mohd Jaafar, W.S.; Omar, H.; Bajaj, S.; Leite, R.V.; Mazlan, S.M.; Silva, C.A.; Chel Gee Ooi, M.; Mohd Said, M.N.; Abdul Maulud, K.N.; et al. Analyzing Canopy Height Patterns and Environmental Landscape Drivers in Tropical Forests Using NASA's GEDI Spaceborne LiDAR. *Remote Sens.* **2022**, *14*, 3172. [CrossRef]

34. *Forest Survey of India India State of Forest Report 2021*, 17th ed.; Chapter 2; Forest Survey of India (Ministry of Environment Forest and Climate Change): Uttarakhand, India, 2021; pp. 15–60.
35. Department of Information. Malaysia Information: Flora and Fauna. Available online: https://www.malaysia.gov.my/portal/content/143 (accessed on 30 November 2023).
36. Soydaner, D. A Comparison of Optimization Algorithms for Deep Learning. *Int. J. Pattern Recognit. Artif. Intell.* **2020**, *34*, 2052013. [CrossRef]
37. Wu, Y.; He, K. Group Normalization. *Int. J. Comput. Vis.* **2020**, *128*, 742–755. [CrossRef]
38. Ru, F.X.; Zulkifley, M.A.; Abdani, S.R.; Spraggon, M. Forest Segmentation with Spatial Pyramid Pooling Modules: A Surveillance System Based on Satellite Images. *Forests* **2023**, *14*, 405. [CrossRef]