remote sensing

MDPI

Article

A Hierarchical Deep-Learning Approach for Rapid Windthrow Detection on PlanetScope and High-Resolution Aerial Image Data

Wolfgang Deigele [1,2], Melanie Brandmeier [1,*] and Christoph Straub [3]

[1] Esri Germany and Switzerland, Ringstr. 7, 85402 Kranzberg, Germany; wolfgang.deigele@tum.de
[2] TUM Department of Aerospace and Geodesy, Technical University of Munich, Arcisstraße 21, 80333 München, Germany
[3] Department of Information Technology, Bavarian State Institute of Forestry, Hans Carl-von-Carlowitz-Platz 1, 85354 Freising, Germany; Christoph.Straub@lwf.bayern.de
* Correspondence: m.brandmeier@esri.de

Received: 19 May 2020; Accepted: 28 June 2020; Published: 2 July 2020

Abstract: Forest damage due to storms causes economic loss and requires a fast response to prevent further damage such as bark beetle infestations. By using Convolutional Neural Networks (CNNs) in conjunction with a GIS, we aim at completely streamlining the detection and mapping process for forest agencies. We developed and tested different CNNs for rapid windthrow detection based on PlanetScope satellite data and high-resolution aerial image data. Depending on the meteorological situation after the storm, PlanetScope data might be rapidly available due to its high temporal resolution, while the acquisition of high-resolution airborne data often takes weeks to a month and is, therefore, used in a second step for more detailed mapping. The study area is located in Bavaria, Germany (ca. 165 km^2), and labels for damaged areas were provided by the Bavarian State Institute of Forestry (LWF). Modifications of a U-Net architecture were compared to other approaches using transfer learning (e.g., VGG19) to find the most efficient architecture for the task on both datasets while keeping the computational time low. A custom implementation of U-Net proved to be more accurate than transfer learning, especially on medium (3 m) resolution PlanetScope imagery (intersection over union score (IoU) 0.55) where transfer learning completely failed. Results for transfer learning based on VGG19 on high-resolution aerial image data are comparable to results from the custom U-Net architecture (IoU 0.76 vs. 0.73). When using both architectures on a dataset from a different area (located in Hesse, Germany), however, we find that the custom implementations have problems generalizing on aerial image data while VGG19 still detects most damage in these images. For PlanetScope data, VGG19 again fails while U-Net achieves reasonable mappings. Results highlight the potential of Deep Learning algorithms to detect damaged areas with an IoU of 0.73 on airborne data and 0.55 on Planet Dove data. The proposed workflow with complete integration into ArcGIS is well-suited for rapid first assessments after a storm event that allows for better planning of the flight campaign followed by detailed mapping in a second stage.

Keywords: forest damage assessment; windthrow; convolutional neural networks; GIS; remote sensing

1. Introduction

Over the past years, the number of storms that caused damage to forests has been increasing due to climate change [1]. The storm "Kolle", for example, was responsible for 2.3 million cubic meters of thrown or broken trees in Bavaria (Germany) in 2017 [2]. Fallen or damaged trees are considered as a loss of timber if not removed in time. Leaving these trees in the forest brings other risks such as an infestation by bark beetles. The European spruce bark beetle Ips typographus, for example, can cause

severe subsequent damage, especially to spruce trees. Therefore, a quick and reliable method to detect the damaged areas is required for forest management.

Manual storm damage detection and mapping based on remote sensing data require a great amount of time and effort, and the automation of the process has been attempted in numerous research articles in the past years. Traditional methods of forest damage assessment utilize a multitude of different sensor types and approaches. A method for autonomous, object-based change detection of storm damages as described by [3] uses high-resolution multispectral images. Satellite images with resolutions of 5 m and 10 m from before and after a storm event in the southwest of France were combined in the study. To achieve the best results, the authors report the degree of damage but also consider how much damage was present prior to the storm. Using a mean shift segmentation algorithm, an overall of 87.8% correctly classified pixels was achieved. This method yields better results than an approach described by [4] to which it was compared. The latter operates on a purely pixel-based detection with an overall accuracy of 78.68%. Using Landsat Thematic Mapper, [5] proposes a method to detect storm damage in rugged terrain in northern Europe. The study area is located in the northern part of Norway with low sun angles that cause particularly long shadows of trees. The data consists of resampled satellite images with a resolution of 25 m. Correction models for the sun elevation are applied to the data to eliminate the effects of long shadows. The vegetation is dominated by spruces that grow on the steep slopes of the area. By using linear and non-linear regression models and by comparison with digital surface models (DSMs), various forms of damage could be detected, ranging from slight defoliation at the treetops to entirely dead trees. The pixel accuracy of damage detection in flat terrain was 80% while only 72% could be achieved in rugged terrain. The study highlights that change detection can lead to good results in areas with rugged terrain and low sun angles (and related shadows) using satellite image data. A two-step approach to forest storm damage classification was presented by [6] for an area around the city of Bern, Switzerland. The data consists of satellite images from the IKONOS and the SPOT4 missions with resolutions of 4 and 10 m, respectively. The different elevation levels of the mountains in the area (555 m to 2,060 m), lead to many different vegetation types and, therefore, tree species compositions. Damage assessment was performed by using a pixel-wise classification that uses the minimum and maximum values for each image band to assign a class to each pixel using a parallelepiped regression algorithm, as well as a rule-based object-oriented classification approach. Labeling of the entire forest area is available and was used together with visual interpretations of the images as ground truth. The achieved classification accuracy is 98% (forest class), 74% (damage class) on the IKONOS images, and 98% (forest class) and 71% (damage class) on the SPOT4 images. A method proposed by [7] utilizes C-band Synthetic Aperture Radar (SAR) data and change detection to determine the differences in back scatter images from before and after the storm event with a minimal operating area of 0.5 ha. With an accuracy of 88% for various test areas, the method is very promising, especially since SAR technology is independent of atmospheric conditions, and data can be generated very quickly after a storm event. Another approach that combines both passive and active methods utilizes unmanned Aircraft system (UAS) together with ALS data for a study area in Slovakia (200 ha) in the Kremnica mountain range [8]. The ALS sensor operates in the near-infrared spectrum with about 5 points per m^2. The UAS data is taken from approximately 700 m height in short flight campaigns. The total detected areas of windthrow is compared to each other, and the reference data that was collected via a Global Navigation Satellite System (GNSS). Windthrows were detected using a comparison of the NVDI computed before and after the event. The overall matching percentage between reference and the UAS based approach is 82%, and in combination with ALS data, the overall matching percentage is 88%.

One major downside of the described methods is their dependency on data before as well as after the storm event. As storms are rather unpredictable and affected areas can be large, it is difficult to keep data from before the event in storage. In [9] a first approach using Deep Learning for damage detection was presented that only requires one post-storm image. Deep Learning and Convolutional Neural Networks (CNNs), in particular, have seen increasing popularity in the field of remote sensing.

The advantage of CNNS in comparison to traditional methods is their capability of learning complex features and to adapt that knowledge to new scenes. In a recent paper [10], an example of landcover mapping using Deep Learning is provided. By using convolutional filters in every single processing step, fully convolutional networks (FNCs) can provide highly accurate results and are widely used in image classification and segmentation tasks (e.g., [11]).

This study is a follow-up project to [9] and further investigates the capacity of convolutional neural networks for satellite and high-resolution data. We use only one image collected after the storm to automatically detect damaged areas based on the texture and color information of the image and investigate the transferability of the final models. Thus, the main focus of this paper is to determine whether PlanetScope data can be used to predict storm damage using advanced CNNs in a first step as it might take a long time to aquire high resolution data from a flight campaign after a storm and this canhamper a rapid response.

2. Materials and Methods

2.1. Data

Our study area is located in the eastern part of the state of Bavaria, Germany (Figure 1). It covers an area of 160 km^2 and consists of forests, fields, and villages. Remote sensing data was collected after a storm event on the 18th of August 2017. The data were available in two different sets, one containing satellite images provided by the PlanetLabs corporation, and one containing aerial images provided by the Bavarian State Institute of Forestry (LWF). The satellite images were collected by the PlanetScope mission that currently consists of 130 small cube satellites with a daily revisit time or lower, resulting in one image of every part of the landmass of the earth at least once a day. The spatial resolution is 3 m, and the spectral resolution is 4 bands (RGB-NIR) [12]. The aerial images have a spatial resolution of 0.2 m and the same spectral bands as the satellite data (RGB-NIR) and were acquired using a digital mapping camera (DMC). During the flight campaign, a forward overlap of 80% and a side overlap of 50% were used. Orthophotos were computed using an ALS-DTM of the Bavarian Surveying Administration for orthorectification. The aerial images are split into 45 tiles with 10,000 × 10,000 pixels for file size reasons.

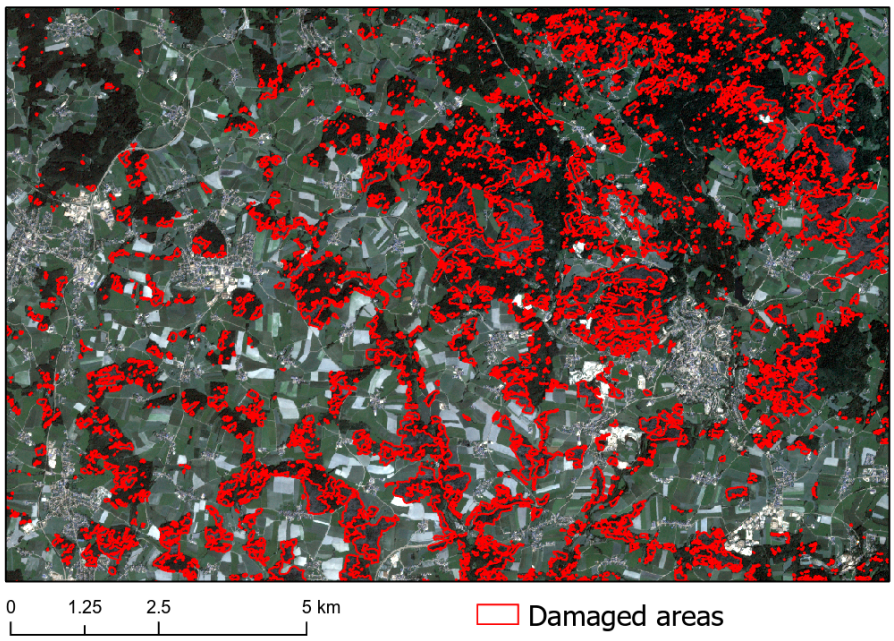

Figure 1. Full extent of the study area with aerial images and respective labels for damaged areas overlain as red polygons. Due to confidentiality agreements, we are not able to add location detail.

Labels of the damage were provided by LWF for both data sets. These labels were originally produced using both remote sensing datasets (PlanetScope and aerial images). First, the PlanetScope data was used to delineate the damaged forest areas in a two-step process. In the first step, an object-based image classification was applied using the software Trimble eCognition. In the second step, all resulting polygons of the object-based classification were visually checked and manually revised. For this mapping, a minimum mapping area of 200 m^2 was defined. After that, a second manual mapping was conducted based on the aerial images. Here, a minimum mapping area of 100 m^2 was used to delineate the damaged areas (Figure 2).

As one of the greatest challenges in remote sensing is the transferability of classification or model from one area to another area, we used a second dataset to assess the performance of our final models. This second dataset was provided by HessenForst and was used as an independent test dataset to validate the model's generalization performance on new data with slightly different characteristics. It includes both, Planet satellite data with 3.5 m resolution and aerial images with 0.2 m resolution as well as corresponding labels. However, the forest structure was not the same as in Bavaria (more deciduous trees), the quality of the labels was not ideal (compared to high-resolution images often not digitized in detail and incomplete), and the state of labeling was not complete as only the state forest was labeled while private forests were not labeled. Thus, we do not have a comprehensive set of ground-truth labels and can only assess the performance visually after the prediction.

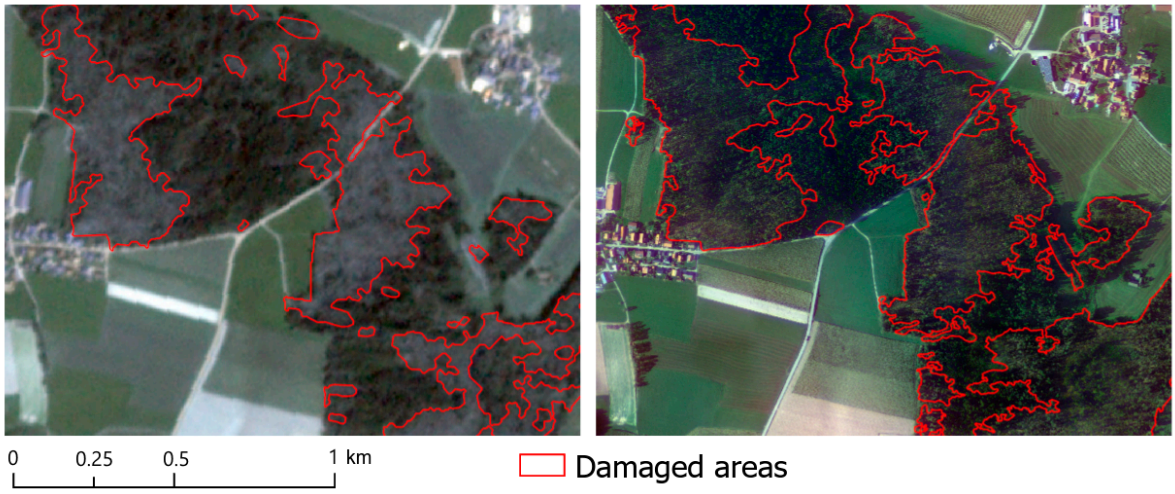

Figure 2. Comparison of the two different sets of labels used in the study overlain on PlanetScope data (**left**) and aerial images (**right**).

2.2. Methods

Figure 3 shows the overall workflow of the study. In the following, we will describe the respective sections in detail.

2.2.1. Data Preprocessing

Prior to feeding the data into the network, it was split into a training and a test set with a ratio of ca. 80:20. This was done by randomly splitting the satellite data into 2 parts, and by selecting 8 out of the 45 ortho tiles. The training data was used to train the convolutional neural networks. The test data was used to assess the performance of the final networks. To feed the data into the network, it was then split into tiles of 256 × 256 × 4 pixels for images and 256 × 256 × 1 pixels for the labeling. The labeling tiles were binary images with the value 0 representing damage and the value 1 representing no damage. This was done for both datasets, satellite, and aerial images, using the ArcGIS Pro (2.5) tool 'Export data for Deep Learning' that, among others, creates RCNN mask tiles from image and labeling data. In total, 2000 satellite tiles and 14,600 ortho tiles were created that contained at least one pixel of

damage (tiles without damage were not exported as this would even increase class imbalance) in their respective labeling tile (Figure 4). The independent test data set provided by HessenForst was not split into tiles as it was only used inside ArcGIS Pro to validate the already trained networks.

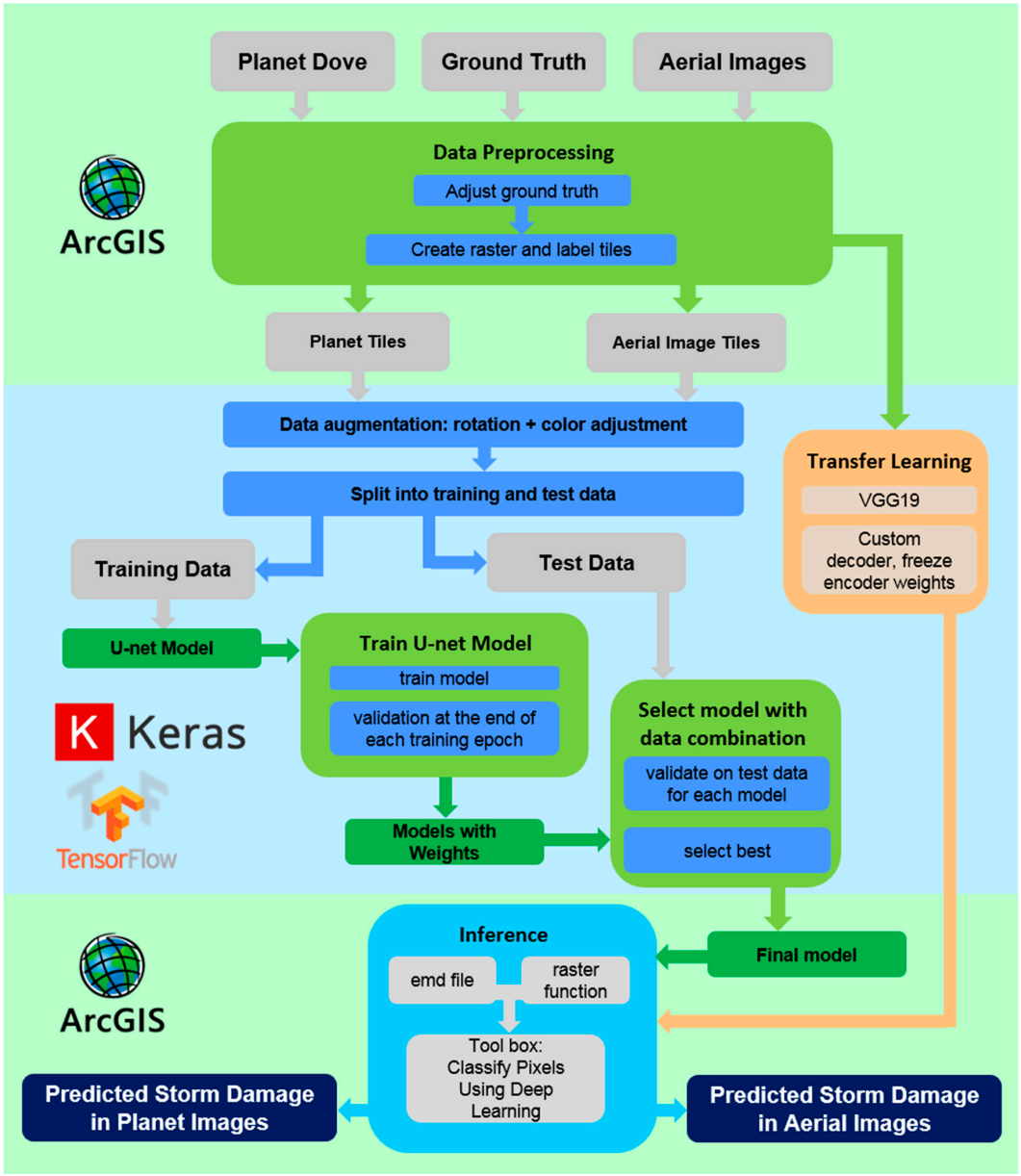

Figure 3. Flowchart showing the analytical workflow of the study.

After the export, extensive data augmentation was performed to increase the set of available training data and to help the model to generalize better. This included rotations, mirroring, as well as adding random noise to the individual image bands. Close attention was paid not to create artificial data that was unlikely or impossible to occur naturally in Germany, like sun angles from the North. After the augmentation step, 24,000 satellite and 60,000 ortho tiles were available for training. Figure 4 shows example images and label tiles for both the satellite and the aerial image data. To additionally increase the data set and improve the ability of the model to distinguish between damaged and undamaged forest, 600 tiles of undamaged forest were added to the data. This was only done for orthophotos since the resolution of 3 m for satellite data left almost no tiles with entirely undamaged forest in the first place.

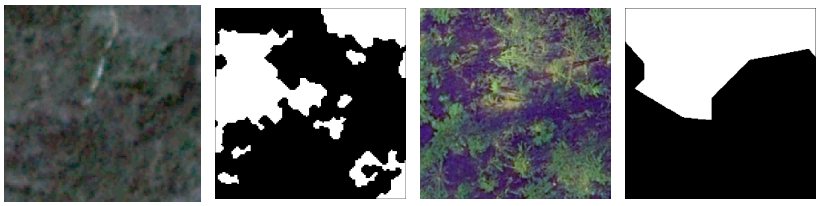

Figure 4. Image and label tiles for satellite (**a**,**b**) and aerial image (**c**,**d**) data. White represents damaged, black not damaged. For scale: 256 × 256 pixels with 3 m (**a**,**b**) and 0.2 m (**c**,**d**) spatial resolution.

2.2.2. Model Architectures

As one of the challenges of this study was the limited amount of training data, even after data augmentation (compared to, for example, the ImageNet dataset (ImageNet, 2016)), we based our main approach on U-Net modifications, since it can be trained from scratch on small data sets. U-Net was originally developed for biomedical image segmentation at the University of Freiburg by Olaf Ronneberger [13] and won first place at the ISBI cell tracking competition 2015 [14]. The name U-Net was derived from the resemblance of the letter U of the encoding and decoding parts (Figure 5). The encoder gradually decreases the input dimensions by a factor of 2 using maximum pooling operations. This keeps only the most significant features at the bottleneck with the cost of losing their exact location and shape. The encoder is a mirrored version of the decoder with upconvolutions instead of max pooling operations, which increases the data dimensions again by a factor of 2 in every step. The unique feature of U-Net are the concatenation steps in between each encoder and decoder block that transferred the extracted features in every sampling step. This leads to a highly detailed segmentation map with relatively low memory usage and inference time. The architecture of U-Net is shown in Figure 5 as it was proposed by Ronneberger et al. We tested several modifications during our study that are described in the next section.

In addition to U-Net, a modification of the much deeper VGG19 was used and compared to U-Net. VGG19 is a version of the VGG model developed by the Oxford Visual Geometry Group. It consists of 19 layers (16 convolution layers, 3 fully connected layers, 5 pooling layers, and a softmax layer) [15]. It was pre-trained on the ImageNet database [16]. Using weights from previous training on a different problem instead of randomly initializing weights and training from scratch is called transfer learning and can be useful for some tasks that are similar, as computation times can be reduced. To adapt the network to semantic segmentation we added a custom decoder consisting of the right half of U-Net with 6 upconvolution blocks. The custom decoder was trained in 5 epochs using our training data while the encoder weights were frozen and thus not updated. Transfer learning was mostly tested on aerial image data, since the histogram curves of satellite data were completely different from those that VGG19 expects, which led to a failure of the model.

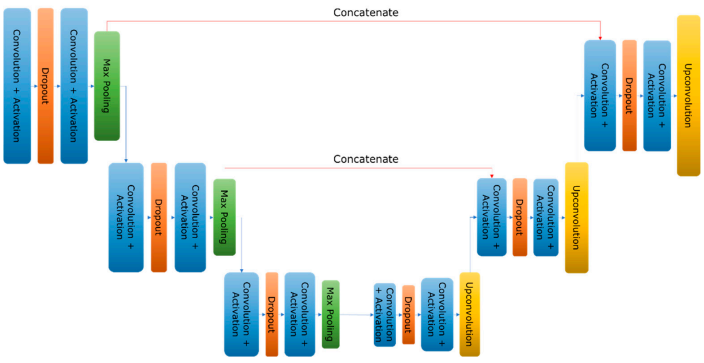

Figure 5. The general architecture of U-Net. We tested different configurations and modifications that are described in the following sections. Modified from Ronneberger et al. (2015) [13].

2.2.3. Evaluation Metrics

A commonly used metric for measuring the performance of a neural network is the accuracy (Equation (1), [17]) where TP, TN, FP, and FN represent true positive, true negative, false positive, and false negative. It states how many pixels were classified correctly in total. There was, however, a large class imbalance in the dataset with up to 95% no-damage pixels in the labels generated based on the PlanetScope data and 65% no-damage pixels in the labels generated based on the aerial image data (compare Figure 2), which made the accuracy not an ideal metric. The Intersection over Union (IoU, Equation (2), [18]) shows how many pixels are correctly classified as damaged to the sum of actual and classified damaged pixels. It is, therefore, the more reliable metric and is further used. Since the outcome of the network was a pseudo probability of each pixel containing damage or not, a threshold of 0.5 was introduced to acquire binary values for validation during training. This threshold can be modified by the user depending on whether to maximize the TPs or the overall accuracy. In our scenario, the forestry department might want to maximize the TPs in order to avoid bark beetle infection in areas that would have been overseen otherwise. Another measure would have been the area under the curve (AUC), but as it is not widely used in computer vision, we opted for the two measures described below. For a discussion of the ROC curve and setting the threshold, we refer the reader to the previous paper by Hamdi [9].

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

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \tag{2}$$

2.2.4. Hyperparameters and Experiments

Hyperparameters were only tested and tuned for the Planet satellite data and labels. As this paper was a follow up of a study by Hamdi et al. [9] that presents extensive research on the same ortho data, we do not focus on the U-Net architecture for aerial images. The respective hyperparameters were adopted with slight adjustments regarding the model architecture as well as an improved loss function. For all experiments on satellite data as well as aerial data, the Rectified Linear Unit (ReLU [19]) was used as activation function and Adam [20] as the optimizer. To find the best modification of the architecture and the ideal hyperparameters for our task, we conducted several experiments. During the training of the models, the dataset was randomly split into a training and validation set (20%) to assess their performance. The hardware used was a Tesla P100 GPU with 16 GB of memory alongside 488 GB of main memory at the facility of the Leibniz Supercomputing Centre (LRZ).

The first parameter we tested was the tile size of the training data. We compared results for sizes of 128 × 128 pixels and 256 × 256 pixels using multiple test scenarios with the other parameters being randomly selected. The second experiment was to find the optimal number of U-Net blocks as well as the individual number of convolutional filters in each block. Additionally, we performed tests to find the optimal learning rate as it was a critical parameter for good results as well as for efficient backpropagation [21]. The introduction of batch normalization layers into our model allowed for slightly higher learning rates, resulting in fewer epochs to reach the same accuracy and thus decrease the likelihood of overfitting. The training itself was done in batches of 80 image and label tiles with the optimum being reached at around 25 epochs for satellite data and 3 epochs for ortho data. The satellite data were heavily unbalanced in favor of "no damage" pixels. Class imbalance is a general problem in many machine learning tasks (e.g., [22–24]) and needed to be addressed. To compensate, a weighted loss function was used (Equation (3)). The weighted binary cross-entropy added weights according to the proportion of pixels that contained damage (5%) and pixels that contained no damage (95%). The imbalance for ortho data was less pronounced, with 35% damage to 65% no damage pixels.

The equation iterates over N pixels with y_i as the true class and $\hat{y}_i$ as the prediction, as well as the additional weights w_1 and w_2.

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i)w_1 + (1 - y_i)\log(1 - \hat{y}_i)w_2] \quad (3)$$

Since the labels derived from aerial images were much more precise in terms of spatial resolution, the network for satellite data was trained not only on the less precise satellite labels but also on the aerial image labels in two different model instances. The testing was also performed with both sets of labels to investigate differences and to find the best setting for future predictions.

3. Results

3.1. Fine-tuning Results

Results from our experiments are summarized in Tables 1 and 2. With respect to tile sizes, 256 × 256 pixels performed better than the smaller tile size for both satellite and ortho data and will thus be used in the final models (Table 1) even though differences were only in the third decimal place of the IoU parameter and thus almost negligible. Testing different modifications of U-Net showed that a block configuration of [8,16,32,64,128] delivered good results, with [64,64,64,64] being slightly better, but considerably slower (Table 2). This configuration represents 5 blocks in total with 8 filters in the first block, followed by blocks of 16, 32, 64, and 128 filters. Finally, the optimal learning rate was found at around 0.002 (Table 1).

Table 1. Comparison of different training scenarios regarding the optimal tile size.

				128 × 128		**256 × 256**	
Scenario	**Learning Rate**	**Number of Blocks**	**Number of Filters Per Block**	**IoU**	**Seconds**	**IoU**	**Seconds**
1	0.001	4	[32,32,32,32]	0.4573	290	0.4588	475
2	0.0015	5	[8,16,32,64]	0.4566	260	0.4574	445
3	0.001	3	[16,32,64]	0.4461	370	0.4512	530
4	0.002	4	[16,16,32,32]	0.4481	310	0.4577	420
5	0.001	5	[8,16,32,64,128]	0.4632	410	0.4658	610

Table 2. Comparison of different training scenarios regarding the optimal block configuration.

Scenario	**Number of Blocks**	**Number of Filters Per Block**	**IoU**	**Seconds Per Epoch**
1	4	[64,64,64,64]	0.4666	880
2	5	[8,16,32,64,128]	0.4658	610
3	4	[16,32,64,128]	0.4640	760
4	3	[32,64,128]	0.4629	830
5	6	[4,8,16,32,64,128]	0.4527	600
6	5	[4,8,16,32,64]	0.4365	410
7	5	[16,16,16,16,16]	0.4457	430
8	4	[32,32,32,32]	0.4576	480
9	5	[64,32,16,8,4]	0.4382	600
10	4	[64,32,16,8]	0.4538	560

3.2. Prediction Results for PlanetScope and Airborne Data

Table 3 shows the final results of our U-Net implementation for satellite data as well as both architectures (U-Net and VGG19) on airborne data. We found that the network worked best when trained and tested using the more accurate labels form the orthophotos and not the less accurate labels form the data itself. This is not surprising as the first set of labels is a much better ground-truth information than the labels for the satellite data.

Results for aerial images were already reported in detail by Hamdi et al. 2019. With our slightly modified architecture that included batch normalization and the weighted binary cross-entropy loss function described in the methods section, final test accuracies were now 86%/0.73 IoU (vs. 92%/ less than 0.42 IoU reached by Hamdi et al., but with different and now much more complete and accurate labeling). Note that compared to the study of Hamdi et al. 2019, who used 12 images, we also used a much larger dataset with 45 images.

Table 3. Test results for different model types on different data and thresholds with aerial labeling.

Model	Data	Learning Rate	Threshold (Accuracy = max)	Accuracy	Threshold (IoU = max)	IoU
U-Net	Planet	0.002	0.26	92%	0.12	0.55
U-Net	Airborne Data	0.002	0.67	86%	0.58	0.73
VGG19	Airborne Data	0.002	0.39	83%	0.51	0.76

Figure 6 shows an example satellite data tile (A), the corresponding satellite label (B) the prediction(C) after setting the respective threshold. D, E, F show an example tile for the final model for orthophotos.

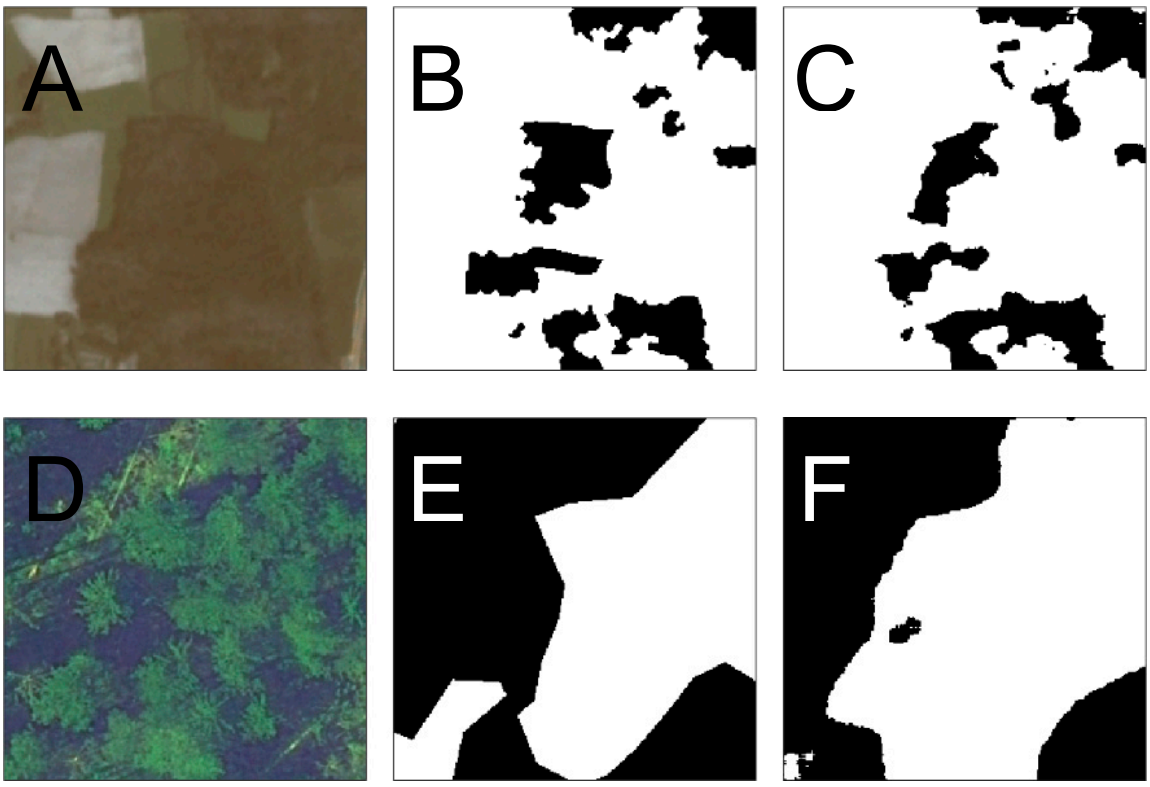

Figure 6. Image tile for Planet (**A**–**C**) and aerial data (**D**–**F**) together with the respective labeling and prediction prior to thresholding.

The final models were now imported into a custom ArcGIS toolbox for upscaling predictions to a large, continuous dataset. This has the advantage that every user can now use the model as a geoprocessing tool, and it allows for further analysis and processing of the results. We created a custom tool that incorporates several post-processing steps, such as smoothing or removal of artifacts. A schematic representation of the steps is shown in Figure 7. The vectorization function is used to create polygons that directly show the total area of damage. By adjusting the threshold, less or more area is included in the prediction. Figure 8 shows an example of the prediction on satellite data with the prediction as red polygons and the ortho labels as green polygons.

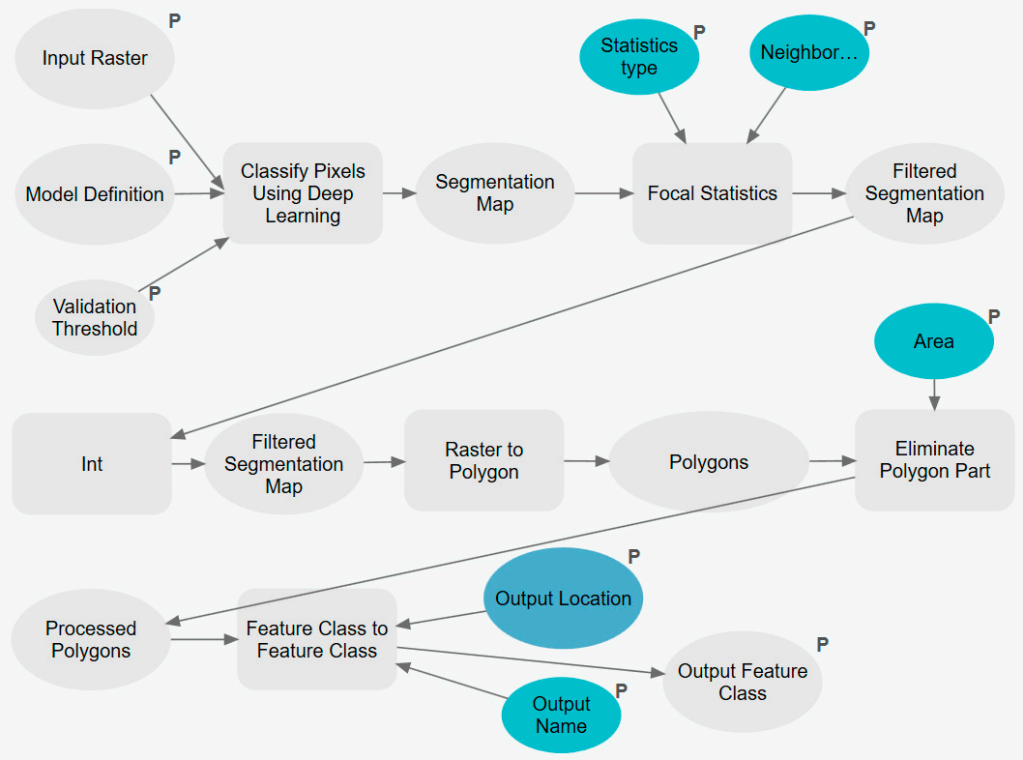

Figure 7. Schematic representation of the processing steps implemented in a toolbox in ArcGIS Pro.

Figure 9 shows the prediction results on the same spatial extent for an aerial image. While the prediction was close to the labels in the western and northern forest border, it was less accurate on the eastern and southern border. This was likely due to shadows that created similarities to the damaged forest (Figure 10). This could be compensated for by using more training data that includes shadows or by adding additional penalties to incorrectly labeled shadow areas during training. Shadows are a common problem in remote sensing, and several approaches exist to deal with this problem. In their paper, [25], for example, show an approach on the classification of cloud, shadow, and land cover scenes using PlanetScope Data that would allow masking these areas.

In addition, we see artifacts outside the forests in settlements. However, this could easily be resolved by using a forest mask (a delineation polygon of forests is available for many forest agencies in Germany). The total predicted damaged area for the model trained on satellite data and the labels generated based on the aerial image data is 34% larger than the labeled area (23.35 km^2 vs. 17.35 km^2). The model trained on the labels generated from the aerial image data was 18% smaller with 14.15 km^2 instead of 17.35 km^2 when maximizing the IoU parameter. Depending on the threshold (0.80 used for satellite and 0.73 for aerial images), the total predicted area can be adjusted without the need of training the model again, depending on the needs of the user. The visual inspection of the maps was very useful in the decision-making process.

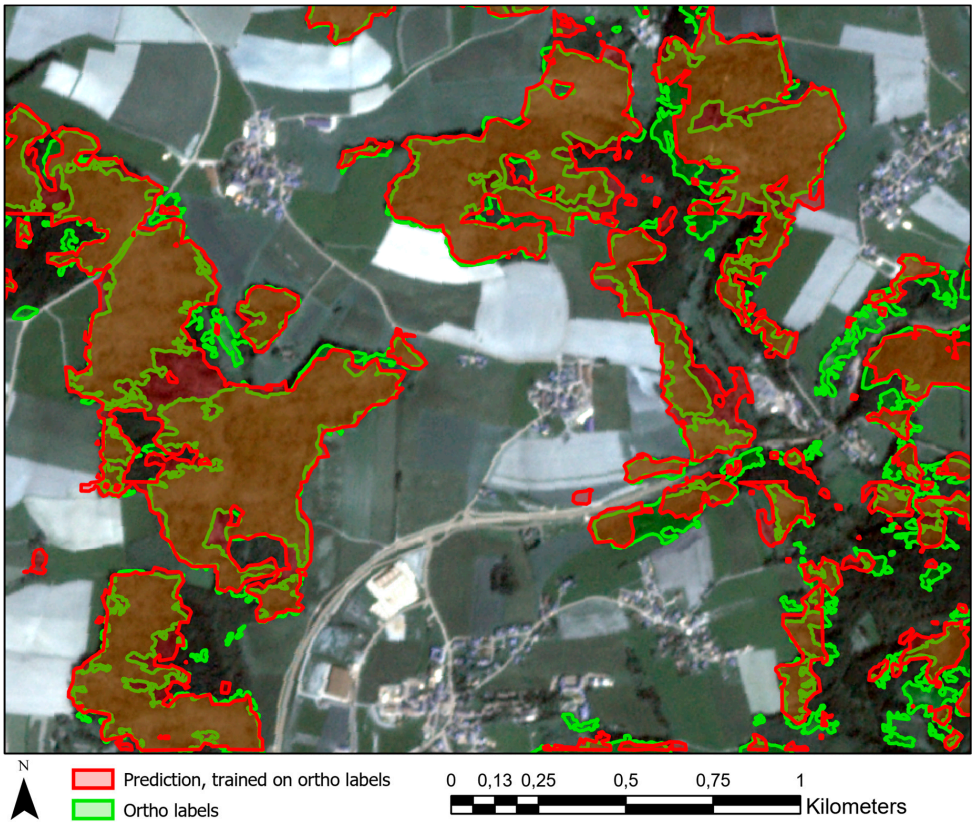

Figure 8. Prediction (red) on PlanetScope satellite image compared to labels (green).

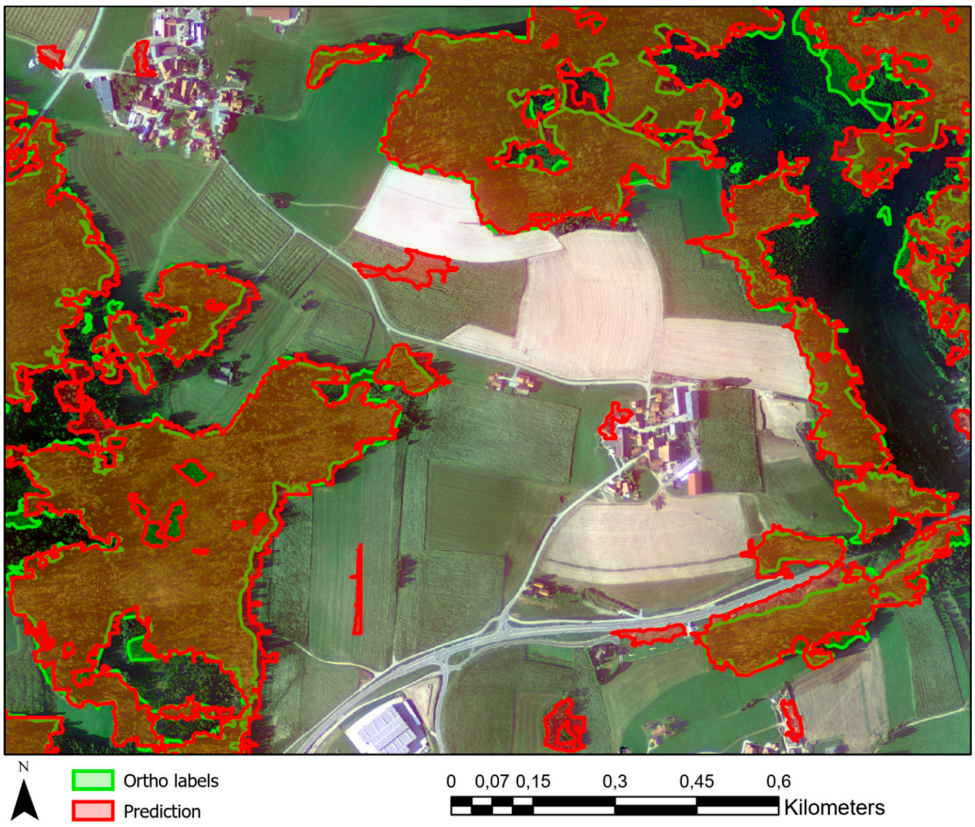

Figure 9. Prediction on orthophoto from an aerial flight campaign (red) compared to the labels (green).

Results for the much deeper VGG19 were slightly better than the custom U-Net with an IoU score of 0.755 at a threshold of 0.51. In a qualitative observation, however, the predictions look more discontinuous and coarser at the edges.

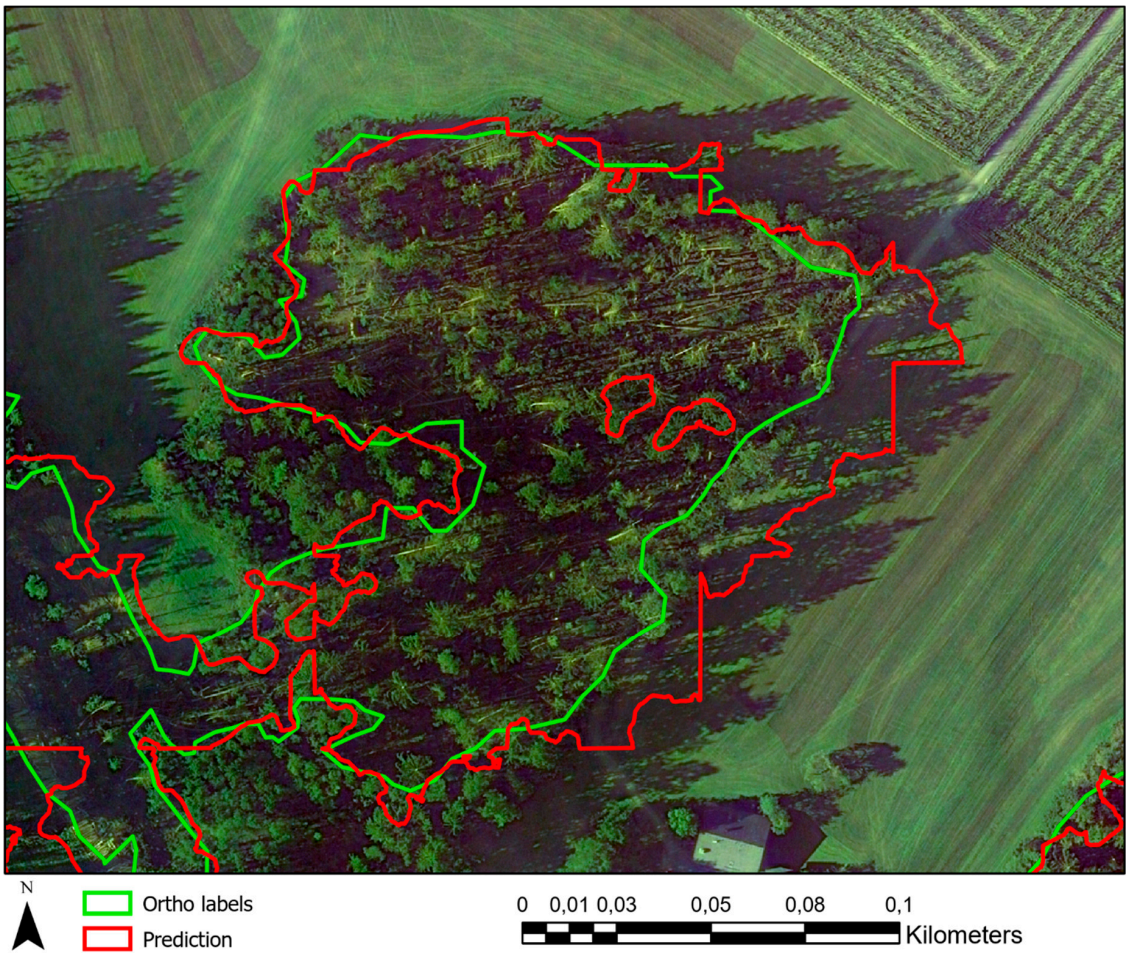

Figure 10. Prediction on orthophoto (red) overlain with the manually digitized reference polygons (green). Shadows lead to overestimating the damaged area by the model.

3.3. Transfer Results (Hesse Data)

We tested the transferability of U-Net for airborne and satellite data as well as for VGG19 on airborne data. As described in the data section, we do not have complete ground-truth labels to report the usual metrics and thus only report results from a qualitative visual inspection within the GIS environment. The prediction of U-Net on satellite data was good given that the satellite data had a slightly different spatial resolution (3 m vs. 3.5 m) and also had considerable variability in spectral characteristics (Figure 11). The extensive data augmentation is a possible explanation. Figure 11 shows an example of the prediction. A lot of the fields were 'false positives', however, these features could be removed using a forest mask that was available from HessenForst. Nevertheless, damaged areas inside the forested areas were recognized quite well. The prediction of our custom U-Net on the airborne data, however, failed. This was due to the difference in recognizable features in this high-resolution data (different forest structures, scattered trees vs. partly removed trees). VGG19 trained on a huge database (14,197,122 images of ImageNet) can deal with a lot more variability and performs considerably better in this scenario. More data, especially more diverse data from different storms and representing various forest types would be required to further investigate the performance of the U-Net architecture and will hopefully be available in the future.

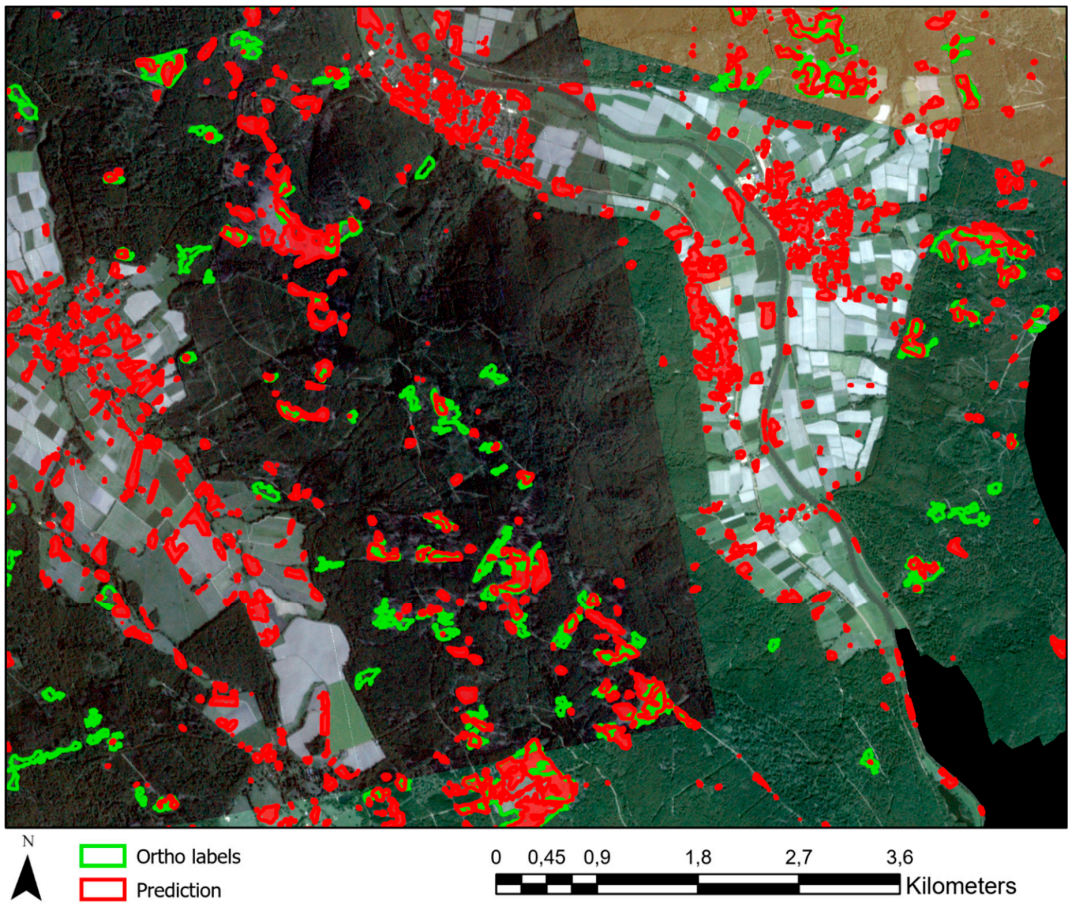

Figure 11. Prediction on a satellite image (red) from Hesse compared to the labels (green). Note that the labels are not complete and of bad quality (see methods section). Spectral differences in the Planet data are due to different acquisition dates.

4. Discussion

In our hierarchical approach for storm damage assessment, we showed how innovative algorithms from the field of computer vision could be adapted to remote sensing data and have the potential of rapidly providing valuable information for disaster management. We developed an AI-based workflow for jointly using satellite images with 3 m resolution for rapid assessment and airborne data for accurate mapping of damaged areas. All models were tested on a test dataset, and a transfer study was conducted to a different area of Germany with slightly different input data to assess the power of generalization. With accuracies of 92% for satellite data and 86% for orthophotos and an intersection over union score of 0.55 and 0.73, respectively, results are very promising. At first glance, it is confusing that the accuracy for satellite data is higher than for orthophotos, while for the IoU parameter, we find the opposite pattern. The latter parameter is the better measure in our scenario as the accuracy parameter also includes true negatives, which is misleading in scenarios with class imbalance. In addition, the parameters are also dependent on the spatial resolution of the data and reflect the accuracy with respect to 20 cm and 3 m, respectively. This leads to the wrong impression that the classification based on Planet data itself is better if only looking at the accuracy parameter.

4.1. Comparison to Other Remote-Sensing Approaches

Compared to traditional machine-learning algorithms such as Support Vector Machines (SVM) or Random Forest Algorithms, CNNs have the advantage of exploiting the correlation between neighboring pixels and, thus, finding spatial features effectively. This explains their success in different computer-vision tasks and has led to some popularity in remote sensing, as highlighted in our study.

In comparison to other approaches for mapping forest damage (e.g., [6,7]), the presented approach only needs one post-storm image and does not require additional datasets from active systems, such as ALS or SAR data, but achieves similar results. However, a comparison of different studies is only possible to a limited extent because different datasets (spatial resolution, spectral resolution, active vs. passive sensors) were used, and the investigations were carried out in different study areas.

The prediction accuracy of our model is comparable to the method described by [3] who used multispectral satellite images and classification on a pixel-based scale with a mean shift segmentation algorithm and it somewhat outperforms the pixel-based classification approach by [5]. Compared to [6], who presented an object-based detection method based on satellite data and parallelepiped regression, our results measured in terms of accuracy are slightly better (92% vs. 74%), however, our images have a better spatial resolution and thus cannot be compared directly (especially as class imbalance has a big impact on the accuracy). In comparison to change detection based on SAR data with a heuristic windthrow index (e.g., [7]), we reached comparable results. With respect to the prior study of Hamdi [9], the addition of batch normalization layers, and the introduction of a weighted cross-entropy loss function, we were able to increase the IoU score from 0.67 to 0.73 by but also saw a decrease in accuracy from 0.92 to 0.86. As the accuracy measure is not ideal for class imbalance settings, we give more importance to the IoU score. Furthermore, the labeling used in the present study was improved significantly by LWF in the meantime, which makes a direct comparison difficult. The extensive data augmentation of this study, which was not performed by Hamdi is very likely to improve results and also the capacity for generalization of the network.

4.2. Limitations of the Proposed Approach

Key challenges using meter-scale satellite data were the spatial resolution of only 3 m, the variability in spectral characteristics as well as the limited amount of training data. The implemented U-Net architecture is well suited to be trained from scratch on small datasets, and with an IoU score of 0.55 most of the relevant damaged areas were detected. The model does have difficulties, however, to detect small damaged areas that only consist of a few pixels. The total predicted damaged area is 34% larger than the ground truth (compared to high-resolution labels from LWF) when optimizing the IoU parameter. However, by adjusting the threshold value that transforms the pseudo probability prediction into a binary classification, this can be adjusted, depending on whether the total accuracy should be optimized, all damage included (but accepting more false positives), or no false positives are allowed. In general, the biggest limitation of the data for forest management is the spatial resolution that is inferior to the airborne data and the variability of the spectral characteristics.

The accuracy of the prediction on airborne data is high, with an IoU score of 0.73. The model is mostly capable of distinguishing between undamaged and damaged forests as well as damaged forests and fields. Only on forest edges where shadows occur, some false positive areas are noticeable. This could be either compensated for by more training data, a special penalty during training for false-positive predictions, or by acquiring images with a high position of the sun.

With respect to the transferability of the models to a different area, we found that U-Net is capable of detecting damage on the Planet data, however, errors occur in the background class and need to be corrected by post-processing algorithms within GIS. More and more diverse data would be needed to further improve the capacity of the models to generalize.

With airborne data, the custom model failed. This is due to the different structures and thus features in the second area. While the airborne data in Bavaria was collected right after the storm and fallen trees are visible, the data for Hesse was only acquired with some delay and damaged areas are not in the original damaged state anymore, but forest management activity has led to cleared areas and other features (such as vehicle tracks) in some areas. This, together with the different forest structures due to a different composition of tree species (more deciduous trees), leads to different features that the custom U-Net has not learned. To overcome this problem, U-Net would have to be trained on a much larger, more diverse dataset, maybe including a third class for already cleared

damaged areas. The issue of input data is confirmed by the transfer learning results based on the modified VGG-19 model that predicts major damaged areas on the Hesse dataset because it is a very deep, pre-trained feature extractor that includes more diverse features than the shallow custom U-Net. However, even for VGG-19, results could still be much better, and more research is needed to further investigate in that direction.

In summary, the most important limitation of the proposed deep-learning approach is the amount of reliable training data as well as sometimes varying conditions of the image collection (e.g., different season, different sensor, temporal delay after the storm.). In the future, if a comprehensive dataset is available, we believe that transferability could be significantly improved. However, another limitation to using huge amounts of data, besides the lack of labels, is the availability of computing facilities that (a) have the storage capacity required for large remote sensing datasets and (b) the GPU processing capability. Using commercial clouds such as Azure or Amazon is still expensive with these requirements and might not be feasible for forest agencies.

5. Conclusions and Outlook

We successfully tested storm damage assessment using Deep Learning on PlanetScope and high-resolution aerial images to support forest management with disaster response. This hierarchical approach allows for fast response and ensures accurate mapping in a second step. A major advantage compared to other state of the art forest damage detection methods is the requirement of only one after-storm image instead of additional data before the storm for a before-after comparison. However, the performance of the model is highly dependent on the quality of the labels used for training. The models were trained and tested on satellite and aerial images with 3 m and 0.2 m spatial resolution and the achieved IoUs are 0.73 and 0.55, respectively (corresponding accuracies of 86% and 93% are comparable to results reported for change-detection approaches) (e.g., [3,5,6] with accuracies of 74% to 92%) and to human perception, especially for satellite data. The custom U-Net proved to be more accurate than a very deep pre-trained network in our study area. Transferring the model to another area, however, was only possible for satellite data as features in high-resolution data differ too much. The transferability of classifiers is an open research question in remote sensing, and more and diverse data is necessary to investigate further the potential of Deep Learning in this direction. Our first tests with transfer learning based on VGG-19 show a lot of potential for high-resolution data; for satellite data appropriately, pre-trained networks are still missing. The integration into ArcGIS Pro results in a seamless workflow for forest agencies as the created toolbox can be used easily by any user, on any machine, as well as in the cloud to process huge datasets. Results can be accessed from mobile devices for field workers in the forest using Collector for ArcGIS.

Author Contributions: Conceptualization, M.B., C.S. and W.D.; writing—original draft, W.D. and M.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: We would like to thank the Bavarian State Institute of Forestry (LWF) for providing the data for this study and HessenForst for providing the data for our transfer study Furthermore, we thank Esri Germany and Switzerland for funding the thesis leading to the paper.

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

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Network
DSM	Digital Surface Model
GPU	Graphics Processing Unit
IoU	Intersection over Union
LWF	Bavarian State Institute of Forestry
ReLu	Rectificed Linear Unit

References

1. Dorland, C.; Tol, R.S.J.; Palutkof, J.P. Vulnerability of The Netherlands and Northwest Europe to storm damage under climate change. *Clim. Chang.* **1999**, *43*, 513–535. [CrossRef]
2. Bavarian State Ministry of Food, Agriculture and Forest. *Annual Report 2017*; Bavarian State Ministry of Food, Agriculture and Forest: Munich, Germany, July 2018; p. 62.
3. Chehata, N.; Orny, C.; Boukir, S.; Guyon, D.; Wigneron, J.P. Object-based change detection in wind storm-damaged forest using high-resolution multispectral images. *Int. J. Remote Sens.* **2014**, *35*, 4758–4777. [CrossRef]
4. Duda, R.; Hart, P.; Stork, D. *Pattern Classification*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2001.
5. Ekstrand, S. Landsat TM-based forest damage assessment: Correction for topographic effects. *Photogramm. Eng.* Remote Sens. **1996**, *62*, 151–161.
6. Schwarz, M.; Steinmeier, C.; Waser, L. Detection of storm losses in alpine forest areas by different methodic approaches using high-resolution satellite data. In Proceedings of the 21st EARSeL Symposium: Observing Our Environment from Space: New Solutions for a New Millenium, Paris, France, 14–16 May 2001; pp. 251–257.
7. Rüetschi, M.; Small, D.; Wasser, L. Rapid detection of windthrows using sentinel-1 C-band SAR data. *Remote Sens.* **2009**, *2*, 115. [CrossRef]
8. Mokroš, M.; Výbošťok, J.; Merganič, J.; Hollaus, M.; Barton, I.; Koreň, M.; Tomaštík, J.; Čerňava, J. Early stage forest windthrow estimation based on unmanned aircraft system imagery. *Forests* **2017**, *9*, 306. [CrossRef]
9. Hamdi, Z.; Brandmeier, M.; Straub, C. Forest damage assessment using deep learning on high resolution remote sensing data. *Remote Sens.* **2019**, *17*, 1976. [CrossRef]
10. Mahdianpari, M.; Salehi, B.; Rezaee, M.; Mohammadimanesh, F.; Zhang, Y. Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery. *Remote Sens.* **2018**, *10*, 1119. [CrossRef]
11. Lin, H.; Shi, Z.; Zou, Z. Fully convolutional network with task partitioning for inshore ship detection in optical remote sensing images. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 1665–1669. [CrossRef]
12. PlanetLabs. Available online: https://www.planet.com/ (accessed on 20 December 2019).
13. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. *Med. Image Comput. Comput. Assist. Interv. (MICCAI)* **2015**, *9351*, 234–241.
14. International Symposium on Biomedical Imaging. Available online: https://biomedicalimaging.org/2015/ (accessed on 20 December 2019).
15. Simonyam, K.; Zisserman, A. Very deep convolutional networks for large scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
16. ImageNet. Available online: http://www.image-net.org/ (accessed on 12 December 2019).
17. Powers, D. Evaluation: From precision, recall and F-factor to ROC, informedness, markedness & correlation. *J. Mach. Learn. Technol.* **2011**, *2*, 37–53.
18. Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 16–20 June 2019; pp. 658–666.
19. Agarap, A. Deep learning using retified linear units (ReLU). *arXiv* **2018**, arXiv:1803.08375.
20. Kingma, D.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
21. Hu, X.; Chen, G. Efficient backpropagation learning using optimal learning rate and momentum. *Neural Netw.* **1997**, *10*, 517–527.
22. Japkowicz, N.; Stephen, S. The class imbalance problem: A systematic study. *Intell. Data Anal.* **2002**, *6*, 429–449. [CrossRef]
23. He, H.; Garcia, E.A. Learning from imbalanced data. *IEEE Trans. Knowl. Data Eng.* **2009**, *21*, 1263–1284.
24. Dong, Q.; Gong, S.; Zhu, X. Imbalanced deep learning by minority class incremental rectification. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 1367–1381. [CrossRef] [PubMed]
25. Shendryk, Y.; Rist, Y.; Ticehurst, C.; Thorburn, P. Deep learning for multi-modal classification of cloud, shadow and land cover scenes in PlanetScope and Sentinel-2 imagery. *ISPRS J. Photogramm. Remote Sens.* **2019**, *157*, 124–136. [CrossRef]