Flower Detection Using Object Analysis: New Ways to Quantify Plant Phenology in a Warming Tundra Biome

Karol Stanski, Isla H. Myers-Smith, and Christopher G. Lucas

***Abstract*—Rising temperatures caused by global warming are affecting the distributions of many plant and animal species across the world. This can lead to structural changes in entire ecosystems, and serious, persistent environmental consequences. However, many of these changes occur in vast and poorly accessible biomes and involve myriad species. As a consequence, conventional methods of measurement and data analysis are resource-intensive, restricted in scope, and in some cases, intractable for measuring species changes in remote areas. In this article, we introduce a method for detecting flowers of tundra plant species in large data sets obtained by aerial drones, making it possible to understand ecological change at scale, in remote areas. We focus on the sedge species *E. vaginatum* that is dominant at the investigated tundra field site in the Canadian Arctic. Our system is a modified version of the Faster R-CNN architecture capable of real-world plant phenology analysis. Our model outperforms experienced human annotators in both detection and counting, recording much higher recall and comparable level of precision, regardless of the image quality caused by varying weather conditions during the data collection. (K. Stanski, GitHub - karoleks4/flower-detection: Flower detection using object analysis: New ways to quantify plant phenology in a warming tundra biome. GitHub. Accessed: Sep. 17, 2021. [Online]. Available: https://github.com/karoleks4/flower-detection.)**

***Index Terms*—Object recognition, remote sensing.**

I. Introduction

The Arctic is warming more rapidly than any other biome on the planet, experiencing an average temperature increase of more than 2°C since 1950 [1]. Its average temperature is predicted to rise by a further 6°C–10°C within the next 100 years [1]. This warming leads to a longer growing season [2] which has been estimated by recent studies to increase in the future by approximately 4.7 days per decade [3]. However, vegetation change research is constrained by the logistics of *in situ* field observations [4]. These standard techniques are extremely costly and cannot be scaled to cover large areas [5]. Therefore, the exact influence of warming on tundra plant communities remains uncertain.

Phenology is a study of plant and animal life cycle events including flower and leaf emergence and decay [6]. The timing of plant phenology can be influenced by changes in a variety of factors, including temperature [5], [7]. Thus, phenological records are valuable for studying the influence of climate change across the world's biomes [6]. The typical way that ecologists have gathered phenological data is to observe phenological changes on-site in localized plots (e.g., 1 m^2 patches or along short transects) [8]. Unfortunately, on-site observations are extremely time-consuming and highly difficult in less accessible areas, including areas of particular importance for understanding ecological and climate change [9]. However, rapidly developing technology allows for new data collection approaches, including proximal remote sensing using drones [10]. The use of unmanned aerial vehicles (UAVs) is a cost-effective method of conducting detailed analysis with high spatial and temporal resolution while avoiding destructive sampling of sensitive ecosystems [11].

Plant phenology captured using high spatial resolution drone imagery comes with methodological challenges such as variable light conditions and complex background [12]. Initial attempts of robust and accurate data analysis included template matching [13], geographic object-based image analysis (GOBIA) [14], regression analysis [15], and Markov point processes [16], none of which yielded accurate enough results to draw meaningful ecological conclusions [17]. A more advanced method, utilizing maximally stable extremal regions (MSER) from drone imagery [18], has been used for turtle and seabird counting with limited success [18], [19].

Recent advances in deep-learning-based models, including convolutional neural networks (CNNs), present new opportunities for efficient and accurate image analysis. Models based on these architectures learn features that tend to be more informative than handcrafted features [20] such as scale-invariant feature transform (SIFT) or histogram of oriented gradients (HOG) [21], and achieve higher image classification accuracies than their predecessors. Moreover, accompanied by hardware advances (i.e., GPUs), some models are now capable of real-time detection [22].

In this article, we propose a fully automated and efficient method for plant flower detection and counting from

Manuscript received April 5, 2021; revised June 23, 2021 and August 6, 2021; accepted August 20, 2021. Date of publication September 8, 2021; date of current version September 24, 2021. This work was supported by NERC ShrubTundra project under Grant NE/M016323/1. *(Corresponding author: Karol Stanski.)*

Karol Stanski and Christopher G. Lucas are with the School of Informatics, University of Edinburgh, EH8 9AB Edinburgh, U.K. (e-mail: karol.stanski@gmail.com; clucas2@inf.ed.ac.uk).

Isla H. Myers-Smith is with the School of GeoSciences, University of Edinburgh, EH9 3FF Edinburgh, U.K. (e-mail: Isla.Myers-Smith@ed.ac.uk).

Digital Object Identifier 10.1109/JSTARS.2021.3110365

high-resolution drone imagery utilizing recently developed deep-learning techniques. Such a tool allows us to quantify the effects of climate change on a tundra biome and other flowering ecosystems, complementing and potentially eliminating the need for on-site measurements. We focus on the sedge species *Eriophorum vaginatum* (*E. vaginatum*), which was the most abundant flowering plant within the investigated area (Qikiqtaruk - Herschel Island; 69°N, 139°W) [23].

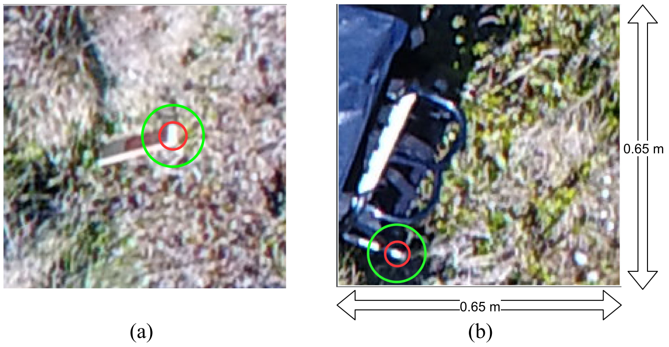
The main contributions of our article are as follows.

1) *Detection model*
   Our model yields better than human-level performance in detecting *E. vaginatum* flowers in Arctic tundra, and can easily be extended to detect other objects. The following modifications make it possible to detect smaller objects than the original faster R-CNN [24]:
   a) parametric ReLU (PReLU) activation unit to alleviate the issue of vanishing gradients [25];
   b) shallow feature extractor to boost small object detection performance;
   c) context path to eliminate false positive detections by considering the information enclosing an object.
2) *Dataset*
   We have created a dataset of 2592 manually annotated images containing nearly 50 000 *E. vaginatum* flower objects. As a result, our database is a valuable resource for future studies regarding the phenology of *E. vaginatum* species as well as tundra biomes in general.
3) *Novel evaluation process*
   We introduce a comprehensive evaluation process for our method assessing its performance against human annotators in both object detection and counting.

II. Related Work

Object detection is a fundamental problem in analyzing remote sensing imagery. Recent advances in detection methods, based on CNNs, have led to dramatic improvements in detection accuracy relative to earlier methods that rely on handcrafted features. State-of-the-art architectures include two-stage region proposal based CNNs (R-CNNs), such as faster R-CNN [24] or feature pyramid network (FPN) [26], which achieve very high accuracy at the cost of real-time performance, and more direct single-step approaches like you-only-look-once (YOLO) [22] which are often capable of real-time detection but have slightly lower accuracy [27].

Most previous work has focused on improving detection accuracy for objects occupying a sizeable area of an image based on standard datasets such as Pascal VOC with instances taking up 14% of the image on average. However, some increasingly popular applications, including analysis of remote sensing imagery, have led to demand for detectors that can identify distant and small objects, requiring architectural improvements. In addition to frequently involving small or low-resolution objects, remote-sensing imagery often includes noisy backgrounds and variable lighting and weather conditions, compounding the challenges in creating high-performance detection systems [17].



(a) (b)

Fig. 1. Examples of false positives generated by the model before the addition of the contextual path. Green and red circles denote effective receptive fields with and without the addition of the contextual path.

One approach to improving the accuracy of small object detection involves using multiple-scale features of increasing complexity. For instance, single shot multibox detector (SSD) [28] predicts objects at each feature level whereas some fully convolutional networks (FCN) [29] combine multiple predictions by averaging segmentation probabilities. Furthermore, higher level fine-grained features provide vital contextual information surrounding an object by increasing the network's effective receptive field to disambiguate an instance from a noisy background (see Fig. 1). Recent studies provide many examples of incorporating context through feature fusion using simple concatenation [12], [30], or element-wise addition operations on the extracted feature maps which greatly improve network performance [31].

Despite introducing various techniques to improve small object detection, most of the studies concerning object detection from remote sensing imagery have been focused on analyzing urban scenes and vehicles in particular. Examples include ships [32] or aircraft [12], [33] as well as buildings such as airports [34]. Few efforts have been made to apply recent advancements to quantify ecological events such as phenological stages. Some attempts involving flower objects and UAV imagery focused mainly on segmentation rather than counting [35]. Other examples include automatic counting of rice seeding [36] or oil palm trees detection and counting [37]. However, these methods are unable to detect overlapping objects and are neither efficient nor robust due to the fixed-size sliding window approach they employ. Another example of applying object analysis in ecology is camera trap detection of wildlife where studies often utilise state-of-the-art model architectures and achieve high levels of accuracy [38]–[40]. Most recently, deep learning models had been used to detect and count insects although from much lower height [41]. Therefore, to analyse the extensive ecological imagery collected throughout the years, including observations of various plant and animal species at long ranges, it is vital to have efficient, reliable, and robust systems [17].

III. Dataset

The original remote sensing imagery was collected from four different sites across Qikiqtaruk - Herschel Island in the Canadian Arctic. The images were gathered between June 2017 and

TABLE I
SPECIFICATIONS OF THE ORIGINAL DRONE IMAGERY AND OUR FINAL DATASET; PS1-PS4 DENOTE PHENOLOGY SITES AT DIFFERENT PARTS OF HERSCHEL ISLAND

Feature	**Original data**	**Dataset**
Collection sites	PS1, PS2, PS3, PS4	PS1, PS2, PS3, PS4
Collection period	June - August	12^{th} - 14^{th} July
Altitude	12m, 24m, 50m, 100m	12m
Weather conditions	Various	Various
Number of images	2625	2592
Image dimensions	5320 × 4200	440 × 440

August 2017 using Phantom 4 Advanced Pro drone platforms. The flights were conducted in variable weather (e.g., wind, cloud cover, mist, etc.) and at different times of the day under variable lighting conditions, giving a wide range of image qualities and appearances of the *E. vaginatum* flowers. Each site was surveyed from four different altitudes, ranging from 12 to 100 m, yielding high-resolution imagery of size 5320 × 4200 pixels. Table I summarizes the details original dataset.

We divided the images into 440 × 440 pixel tiles, to simplify the annotation process for human experts and reduce the memory consumption when training our network. At this stage, we considered only data collected from the 12 m altitude due to its high resolution and visibility of the objects for the human annotators. From this subset, we extracted a uniform random set of 2592 tiles which include a 20 pixel overlap with adjacent tiles to avoid any object truncation and allow lossless reconstruction of the original images of greater size. The overlap was crucial to provide necessary context information regarding the surrounding of the instances which otherwise could be missed or incorrectly classified as a flower.

The ground-truth had been generated through an annotation procedure including nine human experts, mainly Ph.D. and Masters students, who were present on the sites or who were carefully instructed on specific *E. vaginatum* flower characteristics. Data annotation had been split into two parts each lasted between September to December 2017 and 2018, respectively. To achieve the best possible quality of the ground-truth, each tile was annotated by multiple experts with differences resolved by the majority vote and average bounding box generation. Fig. 2 demonstrates an example of an annotated tile. The total number of annotated objects in the dataset reached 50 521 flowers, indicating the scale of the task. Furthermore, the dataset itself represents a valuable resource for the studies of the Arctic tundra phenology by providing accurate locations and population estimates of the *E. vaginatum* species.

Finally, we split the data into three randomly selected, nonoverlapping datasets. The training set was by far the biggest, accounting for 66.6% (1728 tiles) of the annotated images. The remaining tiles were evenly divided into the validation and test sets, both representing 16.6% (432 tiles) of the original dataset. These sets were used to determine the best performing hyperparameter setup and network evaluation, respectively. We also evaluated our model by comparing its performance directly with the detection and counting accuracy of the human experts (see Sections V-D and V-E).

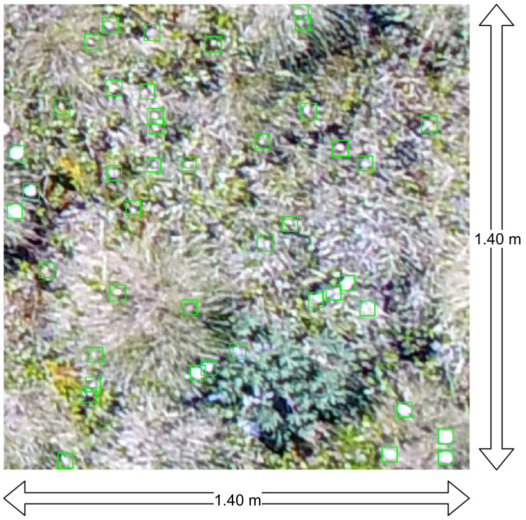


Fig. 2. Example tile from our dataset. The green bounding boxes denote the ground-truth annotated by human experts.

IV. METHOD

The original faster R-CNN architecture is capable of accurate detection of objects that occupy a sizeable area of an image, such as animals or vehicles in the foreground of a photograph. However, remote sensing imagery poses the additional challenges of much smaller object sizes and their resolution. In particular, a single *E. vaginatum* flower occupies only about 0.1% of the whole image area compared with the average of 14% for instances in Pascal VOC dataset [42].

Previous attempts to adjust faster R-CNN to various small object detection tasks included anchor box size adjustments [42], multiscale feature fusion using concatenation [12] or elementwise addition [31]. Here, we modify the faster R-CNN architecture for small object detection used specifically for plant phenology analysis from remote sensing imagery. The detection pipeline, shown in Fig. 3, consists of a backbone feature extractor, region proposal network (RPN) and the final fast R-CNN detector.

The first stage of the pipeline involves feature extraction from the entire input image performed by the backbone network which we describe in more detail in Sections IV-A and IV-C. This fully convolutional network produces a set of features that is shared by the remaining two components making the model a unified framework.

The second stage denotes region proposal generation by the RPN. This small class-agnostic network was the most prominent improvement as the predecessors of the faster R-CNN heavily relied on less efficient methods including selective search to generate a predefined number of proposals that were most likely to contain objects [43]. This nearly cost-free solution significantly improved the model's efficiency by the RPN sharing feature extractor with the rest of the detection network.

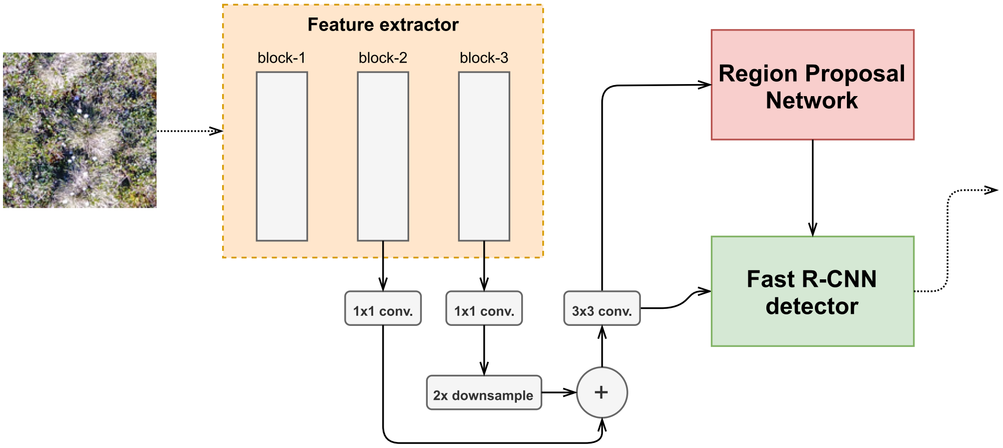


Fig. 3. Architecture diagram of our modified faster R-CNN pipeline for *E. vaginatum* flower detection from remote sensing imagery.

The original implementation of this component involved three anchor sizes $(128^2, 256^2, 512^2)$ of three different ratios (1:2, 1:1, 2:1). However, as suggested by [42], such large sizes are unsuitable for detecting smaller objects which can be enclosed by a smaller box. Thus, we decreased the anchor sizes to $(12^2, 14^2, 16^2, 21^2)$ and reduced them to just a single 1:1 ratio, reflecting the fact that *E. vaginatum* imagines can be reliably enclosed by a square bounding box. The number of proposals generated for each training and testing image was set to 2000 and 300, respectively, following previous work [24].

The final step in our pipeline is the fast R-CNN detector which utilises the image features extracted by the base network along with the proposals generated by the RPN. The proposals are processed by the detector's region of interest (RoI) pooling layer to produce a fixed-size feature vector followed by a set of fully connected layers. The primary purpose of the detector is further classification and bounding box refinement to produce the final detections. For this step, our methods follow those described in [43]. We used the same loss function as the original faster R-CNN architecture consisting of classification and regression components (i.e., multitask loss) with the latter utilizing smooth-L1 loss [24].

A. Shallow Feature Extractor

We reduced the overall depth of the feature extractor compared with the original VGG-16 backbone network, making it much more appropriate for small object detection [44]. Using fewer blocks results in a more suitable receptive field and reduced risk of object characteristics being lost during pooling operations [12]. Shallower convolutional layers extract coarser low-level features which are more appropriate for detecting simpler shapes of *E. vaginatum* flowers. Moreover, feature extractors based on the VGG architecture and its variations yield promising results over other alternatives in tasks involving small object detection [45]. Therefore, our baseline network contains three blocks, each consisting of two to three convolutional layers complemented by activation units and followed by a max-pooling layer (see Table II).

TABLE II
ARCHITECTURE OF THE FEATURE EXTRACTOR BLOCKS

block name	architecture
block-1	$[3 \times 3, 64] \times 2$
block-2	$[3 \times 3, 128] \times 2$
block-3	$[3 \times 3, 256] \times 3$

Due to the specificity of our task and dataset characteristics compared with any standard datasets, we did not utilise pretrained VGG-16 layers. Instead, we trained our network from scratch. That is because the objects of the desired domain ought to be of comparable shape and size as the objects on which the network was pretrained [46]. Hence, the network designed for a new domain is unlikely to benefit from the set of parameters after being trained on a completely unrelated dataset [47].

B. PReLU

Despite a wide variety of available activation functions, ReLU has been the most widely used among the state-of-the-art architectures, including the original VGG-16 feature extractor within faster R-CNN. ReLU activation functions are computationally efficient due to simple thresholding at zero (1) which greatly accelerates the network's convergence; six times faster than sigmoid or tanh functions in some cases [48]. However, since the negative inputs and gradients are all set to zero, those units will eventually stop responding to variations in error/input and *die*, making that segment of the network *passive* [49]. This phenomenon could significantly limit the ability of the network to properly learn from the data [25]

$$relu(y_i) = \begin{cases} y_i, & \text{if } y_i > 0 \\ 0, & \text{if } y_i \leq 0. \end{cases} \tag{1}$$

To alleviate this issue, many alternatives including leaky ReLU introduce a *leakage* parameter (α_i) to the horizontal part of the ReLU graph. However, a constant parameter value for leaky ReLU has a marginal impact on improving network performance when compared with the equivalent architecture using

ReLU [50]. Thus, we followed the idea of *leakage* parameter and incorporated PReLU which progressively learns such parameter for each input channel [α_i, (2)] yielding higher accuracy with a marginal extra computational cost [25]

$$prelu(y_i) = \begin{cases} y_i, & \text{if } y_i > 0 \\ \alpha_i y_i, & \text{if } y_i \leq 0. \end{cases} \tag{2}$$

PReLU introduces a small number of learnable parameters, equal to the total number of channels, which does not slow the training process significantly [25]. Due to the parameter adaptation for each channel, PReLU eliminates the dying ReLU problem as well as reduces the risk of overfitting due to its randomness especially in deeper architectures [51]. Given these advantages of PReLU over standard ReLU activation functions, we experimented with PReLU and found that it improved average precision and F_1 score by 4% and 1% respectively despite our network being relatively shallow. To our knowledge, this is the first work demonstrating PReLU's performance advantages when used within the faster R-CNN framework.

C. Feature Fusion (Context)

Reducing the depth of the feature extractor by eliminating the number of convolutional blocks and pooling layers can bring significant performance improvements while detecting smaller objects [12]. Our first version of the model consisted of only two such blocks with an effective receptive field of 14 pixels delivering a satisfactory F_1 score of 0.74. However, this architecture was highly susceptible to the noisy background including light reflections and field markers, yielding a high number of false-positive detections.

To tackle this issue, we incorporated context information from the enclosing pixels for each object instance by adding an extra convolutional block with a bottom-up path for feature fusion. This way, we extended the effective receptive field to as much as 40 pixels with no information loss due to the coarser features from the shallower block being included in our final set of features. The effective receptive field of 40 pixels is a result of operations applied in each convolutional block with the output fields being 6, 14, and 40 pixels for block-1 to block-3, respectively, when applied in sequence. Similar solutions had been utilized by other faster R-CNN implementations regardless of the feature extraction network type (i.e., VGG-16, Res-Net), which boosted the detection accuracy of smaller objects [12], [30], [31].

The purpose of the additional feature block is to allow the model to extract more complex features. Those features needed to be rescaled in order to perform the fusion with the other set from the higher block. We used a 1×1 convolutional and a 2×2 transposed convolutional operations to compress channel and adjust height/width dimensions, respectively. We performed feature fusion using an element-wise addition layer which simply adds the inputs channel by channel.

After merging, fused feature maps were passed through another convolutional layer with 3×3 kernel followed by PReLU activation to degrade the spatial aliasing effect of downsampling [31], producing the final output of the feature extractor. Hence, the feature extractor component produced 128 feature maps of size 220×220 which were then passed to the remaining two components of our faster R-CNN pipeline.

TABLE III
PARAMETER SETUP FOR EACH TRAINING STAGE

Stage	Iterations	Initial learning rate
1	120k/120k/70k	0.001
2	130k/130k/85k	0.001
3	70k/70k/35k	0.0001
4	70k/70k/70k	0.0001

V. RESULTS AND EVALUATION

Throughout this research, we have tested the original faster R-CNN with VGG-16, VGG-19, and ResNet feature extractors, along with the previous iteration of the faster R-CNN meta-architecture, namely R-CNN [52], and fast R-CNN [43]. However, none of these models proved to be suitable for the task of *E. vaginatum* flower detection and model-to-model comparison, as each achieved final F_1 score of below 0.5. Nevertheless, our evaluation procedure includes other points of reference such as comparison against human experts or counts collected on the ground to deliver a thorough assessment of the model performance. All the experiments and network training were conducted on a machine with an NVIDIA Titan V GPU and 32 GB of memory.

A. Parameter Setting

Our setup included the four-stage alternate training procedure described in [24]. Specific parameters used for each training stage are presented in Table III. The learning rate decay parameter was set to 0.1. The optimiser chosen was mini-batch gradient descent with the momentum parameter of 0.9 and weight decay set to 0.0005. Each mini-batch included one image and 256 proposals per image in detector training. The weights were randomly initialized using zero-mean Gaussian distribution with a standard deviation of 0.01. Furthermore, we applied normalization of each input by subtracting the mean channel values from each colour channel of the image which were determined from the training set.

B. Evaluation Metrics

Our evaluation procedure was based on widely-used metrics within the object detection community. These included precision, recall, F-measure, and average precision (AP) [20]. Furthermore, we used the percentage ratio between the number of the model to ground-truth detections. This metric was vital to establish our network's capabilities of tracking patterns within a species population.

The correctness of each detection is determined by the intersection over union (IoU) with the ground-truth bounding box being at least 0.5. The number of correct detections is denoted as true positives (TP) whereas the incorrect ones as false positives

TABLE IV
FINAL MODEL PERFORMANCE ON THE TEST SET

	Precision	Recall	F_1	AP	Counts
Test set	0.81	0.97	0.87	0.91	1.20

(FP). The instances which are not detected by the network are defined as false negatives (FN). Thus, precision and recall are defined as follows:

$$\text{precision} = \frac{\text{true positives}}{\text{true positives + false positives}} \tag{3}$$

$$\text{recall} = \frac{\text{true positives}}{\text{true positives + false negatives.}} \tag{4}$$

Average precision is a single metric capable of expressing the complex relationship between precision and recall. Mean average precision (mAP) denotes the mean of APs among all considered classes. Since our task considers only one class (*E. vaginatum*), AP is equivalent to mAP. AP denotes area under the precision-recall curve where the integral representing the average precision is approximated by the finite sum over every position in the ranked sequence of the detected objects

$$\text{AP} = \int_0^1 p(r)dr \approx \sum_{k=1}^{n} P(k)\Delta r(k). \tag{5}$$

The F-measure is another metric capable of expressing the relationship between precision and recall scored by a model. We used F_1 score, weigh precision, and recall equally. F_1 score is defined as

$$F_1 = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision + recall}}. \tag{6}$$

C. Standard Metrics Evaluation

To ensure that our results were representative of the actual model performance, we followed the standard validation and testing procedures. Validation set was used to determine the most optimal hyper-parameter setup whereas the model's final performance was established based on the testing set. The results were presented in Table IV with respect to different evaluation metrics.

The original version of faster R-CNN struggled with detecting small objects, recording values below 0.3 across all the considered metrics. In comparison, our version of the network recorded values of precision and recall of 0.81 and 0.97, respectively. Significantly higher recall value can be attributed to the model's tendency to detect more flowers than indicated by the ground-truth (i.e., over-counting by 20% on average) suggesting a noticeable number of false positives. Nevertheless, recorded values of precision and recall demonstrated a balanced performance of the network, summarized by the F_1 score of 0.87. Our network yielded an average precision of 0.91, which we regard as high when compared to other results for similar remote sensing tasks [31], [53].

Furthermore, in the inference phase, our faster R-CNN can process each 440 × 440 tile in 0.05 s on average while running on a single GPU. This corresponds to the processing speed of just over 7 s per single remote sensing image (5320 × 4200 pixels) and introduces a negligible total cost to the overall data collection and processing pipeline.

TABLE V
COMPARISON OF HUMAN AND NETWORK PERFORMANCE BEST SCORES INDICATED IN BOLD

Detector	Precision	Recall	F_1	AP	Counts
Expert-A	0.93	0.58	0.71	0.55	0.62
Expert-B	**0.98**	0.52	0.68	0.51	0.53
Expert-C	0.82	0.44	0.57	0.36	0.54
Expert-D	0.96	0.45	0.61	0.43	0.47
Expert-E	0.82	0.60	0.69	0.48	0.74
Expert-F	0.89	0.49	0.63	0.42	0.54
Our method	0.87	**0.81**	**0.84**	**0.82**	**0.93**

The best scores indicated in bold.

D. Network-Human Evaluation

Due to the uniqueness of the task and the goal to achieve a human-like performance of our method, we compared our faster R-CNN with the performance of human experts in *E. vaginatum* detection and counting. We formed a testing set using 15 randomly selected tiles from the validation set and asked six independent human experts (A-F) who were involved in the dataset annotation to repeat the process. The same set was processed by our model. The results are presented in Table V.

Despite the small size of our corpus relative to the very large corpora in standard image detection tasks, our results present consistent patterns. Humans recorded significantly smaller numbers of detected objects compared with the model, which was indicated by humans' low recall values. This suggests that human annotators struggled to notice certain objects due to their very small area and background noise (see Fig. 4). An alternative explanation could be that humans are generally more reluctant to annotate objects, but we did not further investigate the specific causes of human recall failures. Humans also tended to record fewer false positives (10% of their outputted detections on average compared with nearly 15% for the model), explaining higher precision scores. Human annotators were also inconsistent in the number of flowers they detected. This finding underscores the complexity of the task and the need for a reliable automatic method, setting aside the costs of obtaining human annotations.

Despite the network's precision of 0.87 being far from the best human annotator's score of 0.98, the network's score was midrange, with two annotators recording much lower values of 0.82. Nevertheless, the human annotators' extremely low recall scores prevented them from achieving accurate counts, with nearly half of the ground-truth flowers being missed on average. Thus, our model was decidedly closer to the expected number of flower objects present within each tile, recording an impressive 0.93 of the expected number of objects with the highest human's score being only 0.74, with most human annotators not exceeding 0.60.

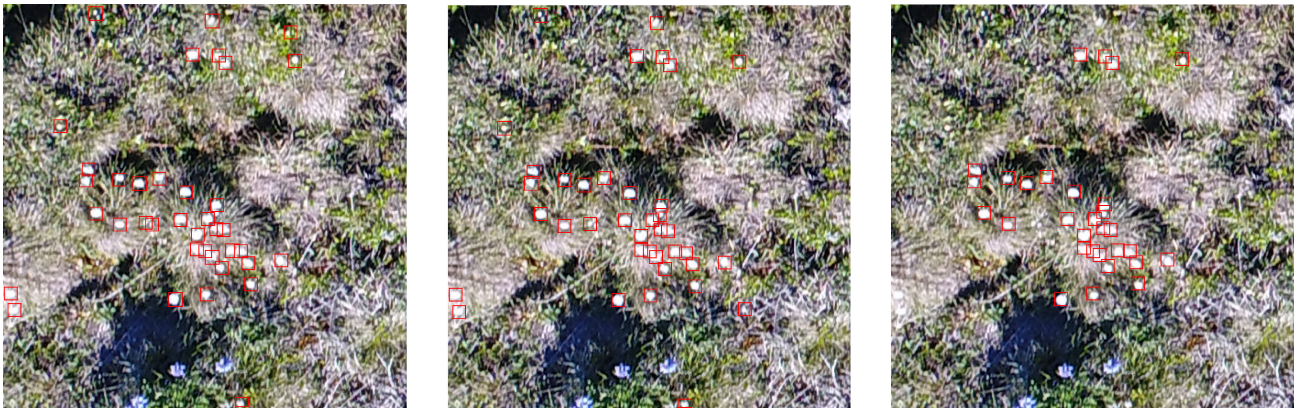

(a) (b) (c)

Fig. 4. Example of flower detections made by different subjects (b), (c) when compared to the ground-truth annotations (a). Red bounding boxes denote the detections.

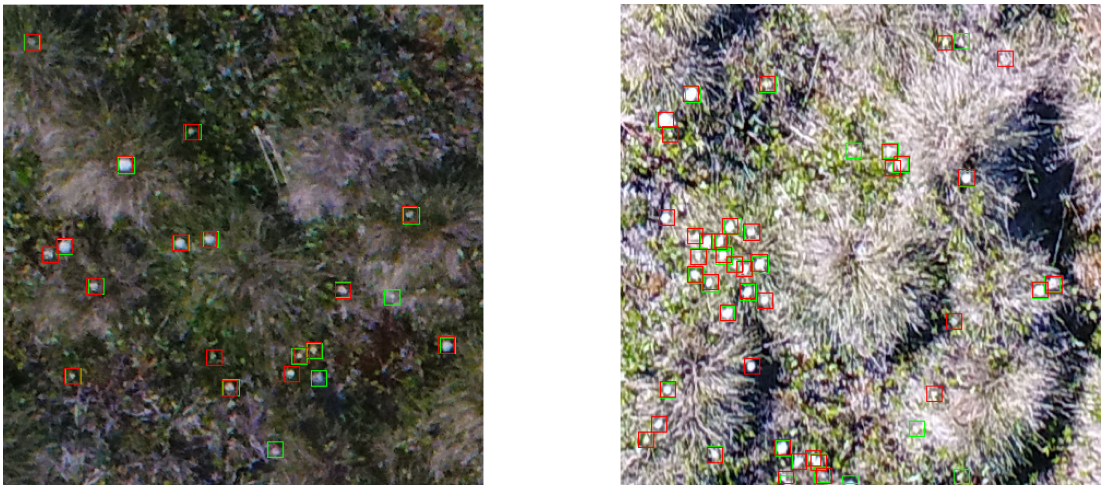

(a) (b)

Fig. 5. Example tiles of varying brightness due to changing weather conditions. The red and green boxes denote model detections and ground-truth, respectively.

Based on the results, our method outperformed all human annotators, with much higher F_1 and AP scores. The poorer performance of human annotators may be attributed to their inability to notice very small flower objects or possibly frustration caused by the tedious nature of the task. These factors did limit our modified faster R-CNN making it more suitable and reliable for analysis of phenology data. Furthermore, faster R-CNN is highly scalable and capable of covering vast areas, which would otherwise require hundreds or thousands of trained annotators.

TABLE VI
COMPARISON OF HUMAN AND NETWORK PERFORMANCE AGAINST THE GROUND COUNTS BEST SCORES INDICATED IN BOLD

Detector	Mean count	Standard deviation
Expert-G	0.80	0.15
Expert-H	0.80	0.16
Expert-I	0.83	0.16
Our method	**1.06**	**0.11**

The best scores indicated in bold.

E. Ground Counts

During the data acquisition, researchers collected flower counts on the ground within $2\tilde{}m \times 2\tilde{}m$ areas around each investigated site. These counts were gathered to determine the number of flowers present at each area unaffected by any image qualities, noise or distortions, unlike the annotation process. Since these data were obtained in person on the ground, those counts are the closest to the true ground-truth making it an appropriate means of assessing our method's performance. We only considered the number of detections (counts) as specific flower locations were not recorded.

The sample consisted of 12 images of the investigated sections from all sites. This set was presented to three human experts (H-I) who performed manual counting. The same imagery was processed by our system. The results are presented in Table VI.

TABLE VII
E. Vaginatum Flower Population Estimation

Site	12/07/2017	14/07/2017	Change
PS-1	29 324	29 337	+0.04 %
PS-2	27 045	27 169	+0.46%
PS-3	31 812	31 100	-2.24%
PS-4	23 805	24 821	+4.27%

The results follow our previous observations regarding inconsistent human performance compared with the model, despite the humans having prior experience with the task. Their counts varied greatly between each image which is summarized by the higher value of standard deviation compared to 0.11 for our method. These results are highly encouraging due to the fact that individuals who took part in the evaluation process were experts within the biodiversity field. The annotators had extensive experience in studying, counting and analyzing habitats of various plant species. They have been analyzing similar data before and knew exactly how *E. vaginatum* flowers look like. They were also familiar with their habitat as well as growing patterns (i.e., often in clusters). Thus, we expected human annotators to be an appropriate benchmark of the model performance. Humans' underperformance can be attributed to severe image distortions such a light reflections or blur as well as very small object size relative the tile dimensions.

Furthermore, our network detected a marginally higher number of objects on average (1.06) than was counted on the ground. This surplus of detections matched our previous testing results although the overcount was not as profound (1.20 on the test set), possibly due to a much smaller image sample size. Despite yielding too many objects, the model's result was consistently closer to the true value than the best human score of only 0.83. Once again, humans detected less than 0.85 of the total number of flower objects, most likely due to the small visible flower area and partial obstruction by grass and other obstacles. On the other hand, small object sizes and background noise did not prevent the network from delivering more accurate counts due to its ability to extract relevant multilevel features and consult contextual information around each instance. This conclusion had been drawn on the fact that our early iteration of the network did not include the contextual path which had an adverse effect, particularly in the case of the most ambiguously looking flowers. That version reported a high number of false-positive detections, especially in presence of light reflections or white field markers as shown in Fig. 1.

F. Population Tracking

With our faster R-CNN capable of reliable object detection and counting, we tested it in a potential real-world environment. To do this, we selected imagery of the four sites from two different days including variable weather conditions to evaluate the robustness of our method regarding changing lighting and image quality. The results are presented in Table VII.

Our faster R-CNN detected a very similar number of flowers on both days, regardless of the site. Real-world flower counts were expected to be similar due to the 48-hour difference in data collection. Such consistency indicated the network's potential reliability even on a much larger scale than previously considered under different lighting and wind conditions (see Fig. 5). Furthermore, we did not observe any extreme variation between the estimates among different sites. According to the results in Table VII, the variation was estimated as ±5%, partially due to noise (i.e., lighting) and occlusion (i.e., branches or grass covering flowers). Nevertheless, the true population size of *E. vaginatum* flowers is likely to be marginally lower than presented in Table VII. That is due to our method's tendency to detect a higher number of objects as shown by our previous results (1.06).

VI. Conclusion

In this article, we introduced our modified version of the faster R-CNN architecture capable of *E. vaginatum* flower species detection and counting. Our major modifications of the feature extractor component included reduced depth and utilization of context information through feature fusion along with the incorporation of a PReLU activation unit. These adaptations yielded promising results in the testing phase, which were further confirmed by the network consistently outperforming human experts at both detection and counting tasks despite varying image quality due to differing weather conditions during data collection. Furthermore, our method did not suffer from random light reflections or noisy background as much as human subjects as indicated by its significantly higher recall scores.

Although we were unable to assess the accuracy of the flower count estimates for *E. vaginatum* species within the investigated area, our results which consistently exceed 20 000 objects per site demonstrate how time-consuming the task of manual counting would be. Other conventional methods involving tracking the numbers within a much smaller region and assuming a similar distribution for the rest of the area might lead to only rough and most likely inaccurate estimations. That is because they do not consider abnormalities caused by varying terrain characteristics or distribution patterns. Our faster R-CNN can cover a much broader area with a high density of flowers in just over 5 minutes per site consisting of 44 images on average. Such scale could only be matched by dedicating a vast number of human annotators to the task, at great expense, and would lead to less accurate counts.

As a future improvement, we found the idea of utilizing multispectral datasets in addition to the conventional RGB optical bands particularly interesting. Adding bands of different wavelengths such as near-infrared (NIR) proved to be beneficial in low visibility conditions in other object detection tasks [54]. Additional bands would help the model to avoid or significantly reduce the number of nonvegetation false-positives by applying NDVI index [55]. Ultimately, all spectral bands could be utilized to make the model decide which bands are the most significant to detect the specific plant species [56].

Global change impacts necessitate new tools to capture ecological responses across the world's biomes [9]. Our work indicates the great potential of faster R-CNN models for image analysis as reliable tools in plant phenology research. Our method is likely to generalise well to different flower species as well as other kinds of plant phenology or ecological data given a thoroughly annotated and sufficiently large image set. Thus, future phenology research can extend localized on-site measurements to landscape scales by combining drone-based data collection with automated flower detection systems.

Acknowledgment

The authors would like to thank NVIDIA Corporation for the donation of the Titan V GPU used for this research through the NVIDIA GPU Programme. Data collection was funded by the NERC ShrubTundra project (NE/M016323/1). Special thanks are also due to all the Team Shrub members including J. Kerby, A. Cunliffe, G. Daskalova, H. Thomas, S. Angers-Blonin, M. Garca Criado, J. Boyle, A. Bjorkman, E. Walker, S. Kellerhals, and I. Rich for dedicating their own time and efforts to collect the data and annotate the imagery for this project. The authors thank Y. Parks for supporting the field data collection and the Inuvialuit People for the opportunity to conduct research on their land.

References

[1] Stocker, T. Ed. *Climate Change, Climate Change 2013–The Physical Science Basis: Working Group I Contribution to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge, U.K.: Cambridge Univ. Press, 2014.

[2] R. Khorsand *et al.*, “Plant phenological responses to a long-term experimental extension of growing season and soil warming in the tussock tundra of Alaska,” *Glob. Change Biol.*, vol. 21, no. 12, pp. 4520–4532, 2015.

[3] T. Park *et al.*, “Changes in growing season duration and productivity of northern vegetation inferred from long-term remote sensing data,” *Environ. Res. Lett.*, vol. 11, no. 8, 2016, Art. no. 084001.

[4] I. Myers-Smith *et al.*, “Complexity revealed in the greening of the Arctic,” *Nature Climate Change*, vol. 10, no. 2, pp. 106–117, 2020.

[5] J. Prevey *et al.*, “Greater temperature sensitivity of plant phenology at colder sites: Implications for convergence across northern latitudes,” *Glob. Change Biol.*, vol. 23, no. 7, pp. 2660–2671, 2017.

[6] E. E. Cleland, I. Chuine, A. Menzel, H. A. Mooney, and M. D. Schwartz, “Shifting plant phenology in response to global change,” *Trends Ecol. Evol.*, vol. 22, no. 7, pp. 357–365, 2007.

[7] J. L. Schedlbauer, N. Fetcher, K. Hood, M. L. Moody, and J. Tang, “Effect of growth temperature on photosynthetic capacity and respiration in three ecotypes of *Eriophorum vaginatum*,” *Ecol. Evol.*, vol. 8, no. 7, pp. 3711–3725, 2018.

[8] S. Oberbauer *et al.*, “Phenological response of tundra plants to background climate variation tested using the International Tundra Experiment,” *Philos. Trans. Roy. Soc. London. Ser. B, Biol. Sci.*, vol. 368, no. 1624, 2013, Art. no. 20120481.

[9] T. Brown *et al.*, “Using phenocams to monitor our changing earth: Toward a global phenocam network,” *Front. Ecol. Environ.*, vol. 14, pp. 84–93, 2016.

[10] M. B. Cruzan *et al.*, “Small unmanned aerial vehicles (micro-UAVs, drones) in plant ecology,” *Appl. Plant Sci.*, vol. 4, no. 9, 2016, Art. no. 1600041.

[11] C. Carl, D. Landgraf, M. van der Maaten-Theunissen, P. Biber, and H. Pretzsch, “Robinia pseudoacacia L. flower analyzed by using an unmanned aerial vehicle (UAV),” *Remote Sens.*, vol. 9, no. 11, 2017, Art. no. 1091.

[12] G. X. Hu, Z. Yang, L. Hu, L. Huang, and J. M. Han, “Small object detection with multiscale features,” *Int. J. Digit. Multimedia Broadcast.*, vol. 2018, pp. 4546896:1–4546896:10, 2018.

[13] P. E. Allen and C. Thorpe, “Some approaches to finding birds in video imagery,” Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-RI-TR-91-34, Dec. 1991.

[14] G. Groom, M. Stjernholm, R. D. Nielsen, A. Fleetwood, and I. K. Petersen, “Remote sensing image data and automated analysis to describe marine bird distributions and abundances,” *Ecological Inform.*, vol. 14, pp. 2–8, 2013.

[15] S. M. Barber-Meyer, G. L. Kooyman, and P. J. Ponganis, “Estimating the relative abundance of emperor penguins at inaccessible colonies using satellite imagery,” *Polar Biol.*, vol. 30, no. 12, pp. 1565–1570, 2007.

[16] S. Descamps, A. Béchet, X. Descombes, A. Arnaud, and J. Zerubia, “An automatic counter for aerial images of aggregations of large birds,” *Bird Study*, vol. 58, no. 3, pp. 302–308, 2011.

[17] G. Cheng and J. Han, “A survey on object detection in optical remote sensing images,” *CoRR*, 2016. [Online]. Available: http://arxiv.org/abs/1603.06201

[18] R. Kimmel, C. Zhang, A. M. Bronstein, and M. M. Bronstein, “Are MSER features really interesting?,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 11, pp. 2316–2320, Nov. 2011.

[19] S. Mader and G. J. Grenzdörffer, “Automatic sea bird detection from high resolution aerial imagery,” *Int. Arch. Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. XLI-B7, pp. 299–303, 2016.

[20] Z. Zhao, P. Zheng, S. Xu, and X. Wu, “Object detection with deep learning: A review,” *CoRR*, 2018. [Online]. Available: http://arxiv.org/abs/1807.05511

[21] G. Antipov, S. Berrani, N. Ruchaud, and J. Dugelay, “Learned vs. hand-crafted features for pedestrian gender recognition,” in *Proc. 23rd Annu. ACM Conf. Multimedia Conf.*, Brisbane, Australia, 2015, pp. 1263–1266.

[22] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” *CoRR*, vol. abs/1804.02767, 2018, *arXiv:1804.02767*.

[23] I. Myers-Smith *et al.*, “Eighteen years of ecological monitoring reveals multiple lines of evidence for tundra vegetation change,” *Ecological Monographs*, vol. 89, no. 2, 2019, Art. no. e01351.

[24] S. Ren, K. He, R. B. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” *CoRR*, vol. abs/1506.01497, 2015, *arXiv:1506.01497*.

[25] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification,” *CoRR*, vol. abs/1502.01852, 2015, *arXiv:1502.01852*.

[26] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, “Feature pyramid networks for object detection,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 936–944.

[27] J. Huang *et al.*, “Speed/accuracy trade-offs for modern convolutional object detectors,” *CoRR*, vol. abs/1611.10012, 2016, *arXiv:1611.10012*.

[28] W. Liu *et al.*, “SSD: Single Shot MultiBox Detector,” in *Proc. Comput. Vis. - ECCV 2016 - 14th Eur. Conf.*, Amsterdam, The Netherlands, Oct. 11–14, 2016, vol. 9905, pp. 21–37.

[29] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” *CoRR*, vol. abs/1411.4038, 2014, *arXiv:1411.4038*.

[30] Y. Chen, Y. Li, and G. Wang, “An enhanced region proposal network for object detection using deep learning method,” *PLoS One*, vol. 13, 2018, Art. no. e0203897.

[31] Y. Ren, C. Zhu, and S. Xiao, “Small object detection in optical remote sensing images via modified faster R-CNN,” *Appl. Sci.*, vol. 8, no. 5, 2018, Art. no. 813.

[32] Z. Zou and Z. Shi, “Ship detection in spaceborne optical image with SVD networks,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 5832–5845, Oct. 2016.

[33] F. Zhang, B. Du, L. Zhang, and M. Xu, “Weakly supervised learning based on coupled convolutional neural networks for aircraft detection,” *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5553–5563, Sep. 2016.

[34] ü. Budak, A. Sengür, and U. Halici, “Deep convolutional neural networks for airport detection in remote sensing images,” in *Proc. 26th Signal Process. Commun. Appl. Conf.*, Izmir, Turkey, 2018, pp. 1–4.

[35] S. Fang *et al.*, “Remote estimation of vegetation fraction and flower fraction in oilseed rape with unmanned aerial vehicle data,” *Remote Sens.*, vol. 8, no. 5, 2016, Art. no. 416.

[36] J. Wu, G. Yang, X. Yang, B. Xu, L. Han, and Y. Zhu, “Automatic counting of in situ rice seedlings from UAV images based on a deep fully convolutional neural network,” *Remote Sens.*, vol. 11, no. 6, 2019, Art. no. 691.

[37] W. Li, H. Fu, L. Yu, and A. P. Cracknell, “Deep learning based oil palm tree detection and counting for high-resolution remote sensing images,” *Remote Sens.*, vol. 9, no. 1, 2017, Art. no. 22.

[38] M. S. Norouzzadeh, D. Morris, S. Beery, N. Joshi, N. Jojic, and J. Clune, “A deep active learning system for species identification and counting in camera trap images,” *CoRR*, vol. abs/1910.09716, 2019, *arXiv:1910.09716*.

[39] S. Schneider, S. Greenberg, G. W. Taylor, and S. C. Kremer, “Three critical factors affecting automated image species recognition performance for camera traps,” *Ecol. Evol.*, vol. 10, no. 7, pp. 3503–3517, 2020.

[40] I. A. Zualkernan *et al.*, “Towards an IoT-based deep learning architecture for camera trap image classification,” in *Proc. IEEE Glob. Conf. Artif. Intell. Internet Things*, 2020, pp. 1–6.

[41] T. T. Høye *et al.*, "Deep learning and computer vision will transform entomology," *Proc. Nat. Acad. Sci.*, vol. 118, no. 2, 2021.[Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7812775/

[42] H. Krishna and C. V. Jawahar, "Improving small object detection," in *Proc. 4th IAPR Asian Conf. Pattern Recognit.*, Nanjing, China, 2017, pp. 340–345.

[43] R. B. Girshick, "Fast R-CNN," *CoRR*, vol. abs/1504.08083, 2015, *arXiv:1504.08083*.

[44] C. Eggert, S. Brehm, A. Winschel, D. Zecha, and R. Lienhart, "A closer look: Small object detection in faster R-CNN," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2017, pp. 421–426.

[45] L. W. Sommer, T. Schuchert, and J. Beyerer, "Fast deep vehicle detection in aerial images," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Santa Rosa, CA, USA, 2017, pp. 311–319.

[46] B. Singh and L. S. Davis, "An analysis of scale invariance in object detection SNIP," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 3578–3587.

[47] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. 27th Int. Conf. Artif. Neural Netw., Artif. Neural Netw. Mach. Learn. Proc., Part III*, Rhodes, Greece, 2018, pp. 270–279.

[48] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. 25: 26th Annu. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1106–1114.

[49] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for Activation Functions," *CoRR*, vol. abs/1710.05941, 2017, *arXiv:1710.05941*.

[50] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. Int. Conf. Mach. Learn. Workshop Deep Learn. Audio, Speech Lang. Process.*, 2013. [Online]. Available: www.semanticscholar.org/paper/Rectifier-Nonlinearities-Improve-Neural-Network-Maas/367f2c63a6f6a10b3b64b8729d601e69337ee3cc

[51] B. Xu, N. Wang, T. Chen, and M. Li, "Empirical evaluation of rectified activations in convolutional network," *CoRR*, vol. abs/1505.00853, 2015, *arXiv:1505.00853*.

[52] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, 2014, pp. 580–587.

[53] J. Pang, C. Li, J. Shi, Z. Xu, and H. Feng, "R^2-CNN: Fast tiny object detection in large-scale remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5512–5524, Aug. 2019.

[54] M. O. Gani, S. Kuiry, A. Das, M. Nasipuri, and N. Das, "Multispectral object detection with deep learning," in *Communications in Computer and Information Science*, vol. 1406. Berlin, Germany: Springer, 2021, pp. 105–117.

[55] B. Ayhan *et al.*, "Vegetation detection using deep learning and conventional methods," *Remote. Sens.*, vol. 12, no. 15, 2020, Art. no. 2502.

[56] J. You, X. Li, M. Low, D. B. Lobell, and S. Ermon, "Deep Gaussian process for crop yield prediction based on remote sensing data," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4559–4566.

Karol Stanski received the master's degree in informatics from the University of Edinburgh, Edinburgh, U.K., in 2019.

He completed summer internship programmes with MathWorks, Cambridge, U.K., and Arm in 2017 and 2018, respectively.

Isla H. Myers-Smith received the Ph.D. degree from the University of Alberta, Edmonton, Canada.

She is a Chancellor's Fellow and a Senior Lecturer in the School of GeoSciences, University of Edinburgh, Edinburgh, U.K. Her research quantifies how tundra ecosystems are responding across temporal and spatial scales to rapid warming and environmental change. She conducts field research in the Arctic and sub-Arctic of Canada and conducts data synthesis at tundra biome and global scales.

Christopher G. Lucas received the Ph.D. degree from the University of California, Berkley, CA, USA.

He is a Chancellor's Fellow and a Lecturer WITH the School of Informatics, University of Edinburgh, Edinburgh, U.K. His research interests include the development and evaluation of sample-efficient and computationally efficient statistical and machine learning models, and comparing them to human performance. He has published this work in a variety of venues include NeurIPS, PNAS, Cognition, and Psychological Review.