

CLASSIFICATION OF MAIZE GROWTH STAGES USING DEEP NEURAL NETWORKS WITH VOTING CLASSIFIER

Justyna S. Stypułkowska^{1,2,*} , Przemysław Rokita² 

¹Lukasiewicz Research Network – Institute of Aviation, Warsaw, Poland

²Warsaw University of Technology, Warsaw, Poland

*Corresponding author: Justyna S. Stypułkowska (justyna.stypulkowska@ilot.lukasiewicz.gov.pl)

Abstract Deep learning significantly supports key tasks in science, engineering, and precision agriculture. In this study, we propose a method for automatically determining maize developmental stages on the BBCH scale (phases 10-19) using RGB and multispectral images, deep neural networks, and a voting classifier. The method was evaluated using RGB images and multispectral data from the MicaSense RedEdge MX-Dual camera, with training conducted on HTC_r50, HTC_r101, HTC_x101, and Mask2Former architectures. The models were trained on RGB images and separately on individual spectral channels from the multispectral camera, and their effectiveness was evaluated based on classification performance. For multispectral images, a voting classifier was employed because the varying perspectives of individual spectral channels made it impossible to align and merge them into a single coherent image. Results indicate that HTC_r50, HTC_r101, and HTC_x101 trained on spectral channels with a voting classifier outperformed their RGB-trained counterparts in precision, recall, and F1-score, while Mask2Former demonstrated higher precision with a voting classifier but achieved better accuracy, recall, and F1-score when trained on RGB images. Mask2Former trained on RGB images yielded the highest accuracy, whereas HTC_r50 trained on spectral channels with a voting classifier achieved superior precision, recall, and F1-score. This approach facilitates automated monitoring of maize growth stages and supports result aggregation for precision agriculture applications. It offers a scalable framework that can be adapted for other crops with appropriate labeled datasets, highlighting the potential of deep learning for crop condition assessment in precision agriculture and beyond.

Keywords: AI, deep learning, image recognition, RGB imaging, multispectral imaging, voting classifier, precision farming, determining growth stages of maize, BBCH scale.

1. Introduction

Knowledge of the developmental phases of plants and their precise determination for individual locations are crucial for calculating plant condition parameters within larger areas of semi-cultivated fields, in line with the concept of precision agriculture [12,21,22]. A measure commonly used by scientists to quantify the developmental phase of a plant is in is the international plant development scale BBCH [17,21]. The BBCH scale (Biologische Bundesanstalt, Bundessortenamt und Chemische Industrie) is a standardized system for identifying the phenological development stages of plants. It uses a two-digit coding system where the first digit represents the principal growth stage (e.g., germination, leaf development, flowering), and the second digit provides a more detailed subdivision of each stage (e.g., the number of leaves developed) [17]. This scale allows for consistent documentation and comparison of growth stages across different plant species and has

been widely adopted by researchers, agronomists, and farmers for crop monitoring and management [17, 21].

The BBCH scale is widely utilized by agronomists, researchers, and farmers to monitor and document the growth stages of crops. It aids in standardizing the timing for agricultural practices such as fertilizing, and pesticide application, ensuring optimal crop management and productivity [16, 21, 22]. Until now, the determination of developmental phases of specific plant species has only been done manually by visually analyzing the plants [17]. Automating this process allows faster analysis, which will have a directly impacts on timely human intervention and help provide plants with the right conditions for development. These facts underline the need to develop a robust and rapid method of assessing developmental phases, which can be carried out in an automated manner.

Artificial intelligence comes to the aid of this process [16, 21, 24]. The use of deep learning techniques and the appropriate preparation of new training datasets make it possible to develop trained models capable of detecting the indicated plants and determining their developmental phases based on image analysis [12]. The automatic detection and classification of the BBCH phases of plant development using artificial intelligence is still something of a novelty at present, but it is certainly the direction of the future in precision agriculture [22]. The automation of this process with a defined accuracy and speed, using deep learning algorithms, is therefore a very valuable and desirable advancement compared to the current method of manually determining these parameters. The development of this issue is heavily dependent on the creation of a dedicated dataset [24]. We were the first to create datasets composed of images representing maize plants growing in a real crop field. One dataset consists of RGB images, and the other consists of images acquired in 10 spectral channels. In these datasets, we assigned each plant a corresponding BBCH scale value (from stage 10 to stage 19). This enabled us to develop a method to train AI algorithms to create a model capable of analyzing the images and automatically determining the developmental phases of the plants under study, in this case maize, from the images. Without the use of a similar solution, assessing the quality of plant development parameters on a large scale in a controlled environment is unattainable, given the enormous time and effort required from participants. Our solution supports crop management from its initial stage and can support yield early enough so that the amount of food produced can be easily and efficiently increased. Our method is the start of research into accurately determining the early stages of plant development (in this case maize) from close range and is far superior to existing manual methods in terms of efficiency.

In this paper, we focus on the replication of results using the following deep learning architectures: HTC_r101, HTC_r50, HTC_x101, and Mask2Former [20, 30, 32, 33]. These architectures demonstrate optimal performance in terms of the training process and do not require excessive computational resources for training. We investigate their efficiency and effectiveness when trained on a set of RGB and multispectral images.

In addition, we conducted tests by dividing the multispectral dataset into individual single spectral channels, which provided an answer as to which of the tested algorithms performs best on the indicated datasets and which dataset to use for the detection and classification of developmental phases to achieve the best results. We have also introduced an additional method using a voting classifier, in which models trained on individual spectral channels vote on the final result of selecting a class denoting a specific developmental phase on the BBCH scale. This novel research expands our scope of data analysis to other spectra beyond the previously popular RGB imaging. Our work opens up a new avenue to explore new questions and inspires us to continue our research with a new dataset combining spectral channels and to use another multispectral camera for this purpose as well.

2. Related works

Assessing plant growth stages is crucial for determining their condition parameters [21]. In precision agriculture, an additional requirement is the automation of this process and its accuracy, even at the level of individual plants or small areas within large fields [21]. This ensures proper control over plant development conditions and helps maintain high food quality.

Automatic determination of plant growth stages and conditions has significantly advanced thanks to deep learning and image analysis [22]. Traditional methods rely on manually inspecting plants to document their growth stages, while modern approaches use automated representation learning from images to predict outcomes and assign growth stage values to plants, typically using the BBCH scale [12].

Several studies have explored the use of deep learning models to determine plant condition parameters, including the classification of maize growth stages. This addresses the increasing demand for such solutions in agriculture, particularly in precision farming. These solutions are being developed to meet the need for effective crop management and the monitoring of condition parameters.

For example, Xu et al. [30] proposed a deep learning approach for determining maize growth stages by counting leaves. They developed a two-step method combining instance segmentation and object detection, employing Mask R-CNN and YOLOv5 architectures. This method effectively detected and counted leaves, overcoming challenges related to background and weeds. Using RGB images captured by UAVs, their approach represents a significant advancement in precision agriculture.

Liu et al. [20] developed a system to measure maize seedling emergence by evaluating count, size, uniformity, and distribution. Using deep learning with UAV-captured RGB images, they overcame challenges like shadows and planting density. The system, based on the YOLO architecture and TOPSIS method, accurately assessed seedling quality and identified areas with poor emergence in experimental fields.

Yu et al. [32] used deep convolutional neural networks (DCNNs) to estimate maize aboveground biomass (AGB) from multisource UAV imagery. They showed that AGB estimation, essential for crop growth assessment, can be effectively modeled using regression between AGB and agronomic traits from UAV data. DCNNs provided superior results, especially during the vegetative phase.

Zhang et al. [33] developed a method for detecting maize tassels in UAV-captured RGB images. They highlighted that tassel developmental stage and branch number are key phenotypic traits for assessing growth, pollen quantity, and planning tassel pruning in seed fields. Using a Random Forest classifier and the VGG16 network, their algorithm effectively detected tassels in complex field conditions, improving crop management and yield quality assessment.

Yao et al. [31] proposed a method for classifying maize growth stages using phenotypic traits and UAV-captured data. They combined vegetation indices (VI), textural features (TF), and phenotypic parameters like leaf chlorophyll content (LCC), leaf area index (LAI), fractional vegetation cover (FVC), and canopy height (CH). The highest accuracy (95.1%) was achieved with a Random Forest classifier using LCC, LAI, FVC, and CH. The study showed phenotypic features outperform vegetation indices, and integrating UAV data with machine learning enables accurate maize growth stage monitoring.

Bera et al. [3] proposed PND-Net, a system combining graph convolutional networks (GCN) with traditional CNNs to classify plant nutrient deficiencies and diseases. The model integrates local leaf image features (Xception, ResNet-50, Inception-V3, MobileNet-V2) with spatial relationships captured by GCN, using spatial pyramid pooling (SPP) for multi-scale feature aggregation. Tests on datasets (banana, coffee, potato, PlantDoc) showed high performance: 90.00%, 90.54%, 96.18%, and 84.30%, respectively. PND-Net also achieved state-of-the-art results in medical image classification (BreakHis, SIPaKMeD), making it valuable for precision agriculture and medicine.

Bera et al. [4] proposed RAFA-Net, a method combining CNNs with a regional attention mechanism for food classification and plant stress recognition. The model captures contextual information and long-range dependencies using spatial pyramid pooling (SPP) and average pooling. Tested on food datasets (UECFood-100, UECFood-256, MAFood-121) and plant stress datasets (IP-102, PlantDoc-27), RAFA-Net achieved top accuracies of 91.69%, 91.56%, 96.97%, 92.36%, and 85.54%. The results highlight RAFA-Net's effectiveness in precision agriculture and food processing.

Wu et al. [28] proposed an innovative approach for identifying strawberry diseases using a deep learning model based on the Squeeze-and-Excitation (SE) mechanism. The system integrates sensor data acquisition and plant imaging, transmitting images to the cloud via a dedicated gateway for analysis. This solution enables efficient monitoring of strawberry health, improving crop management and yields.

Bompani et al. [5] explore the implementation of computer vision algorithms on a

heterogeneous multicore microcontroller to accelerate pest detection, specifically targeting the codling moth in apple orchards. Sensor nodes with cameras capture and process images locally, thereby reducing transmission delays. This approach improves real-time pest monitoring, which is crucial for protecting crops and minimizing losses.

Bansal et al. [1] proposed PA-RDFKNet, a deep learning model integrating RGB and hyperspectral imaging for plant age estimation. By combining features from both modalities, PA-RDFKNet significantly improves accuracy over single-modality methods. This approach supports precision agriculture by enhancing plant growth monitoring and optimizing agronomic practices.

Bera et al. [2] introduced APDC (Attention-based Plant Disease Classification), a method using CNNs with an attention mechanism to identify plant diseases from leaf images. The model extracts features, highlights key regions, and classifies them with a softmax layer. Tested on PlantPathology, PaddyCrop, PaddyDoctor, and PlantVillage datasets, APDC achieved accuracies of 97.74%, 99.16%, 99.62%, and 99.97%. This end-to-end trainable model, using lightweight CNNs like MobileNet-v2 and DenseNet-169, is efficient for precision agriculture.

Based on the reviewed articles, most researchers successfully use deep learning models to determine growth stages and other plant condition parameters. They primarily employ RGB imaging techniques captured by UAVs for this purpose. As we have observed, some of the most powerful deep learning models, such as HTC and Mask2Former, have not yet been explored for this specific task. The literature review also revealed that RGB imaging is typically used for determining plant growth stages, including maize, rather than multispectral imaging. This has led to gaps in knowledge regarding whether HTC and Mask2Former can be effectively applied and implemented for maize growth stage classification. Another research gap is the limited use of multispectral imaging for determining plant growth stages, particularly for maize. This article aims to address these research gaps. We intend to investigate which type of imaging (RGB or multispectral) is more suitable for this task and which of the examined algorithms proves to be the most effective. The goal of the research is to select the most effective configuration in the form of: imaging type plus the chosen algorithm from the following options: HTC_r50, HTC_r101, HTC_x101, and Mask2Former.

3. Data and methods

3.1. Datasets

In order to carry out the research, the results of which we present in this article, we used specially prepared datasets of tagged images of maize at different stages of development, divided into RGB and multispectral sets. Data were collected from the same test plots, during the 2021–2022 growing seasons. We collected the data using an RGB camera

Tab. 1. Parameters of RGB and Multispectral Datasets

Dataset Type	Camera	Image Count	Resolution	Total Labeled Objects	Objects per Class (BBCH 10–19)
RGB	RGB Camera	396	12 MP	884	83, 83, 137, 83, 83 83, 83, 83, 83, 83
Multispectral	Micasense RedEdge-MX DUAL (10 spectral channels)	556 per channel	5 MP	9070 (907 per channel)	82, 86, 162, 82, 82 83, 84, 83, 83, 80 (per channel)

and a MicaSense RedEdge-MX DUAL multispectral camera, which captures 10 spectral channels covering the 400 to 900 nm range.

The RGB camera used automatic white balance and exposure settings, eliminating the need for additional manual calibration. For the multispectral camera, we used standard calibration procedures involving reference panels with known reflectance values. This ensured consistency and accuracy of the spectral data captured across different channels.

The datasets include images captured under varying weather conditions, different levels of sunlight, and at various times of the day to maximize diversity. The images were labeled according to the international BBCH plant development scale, adapted for maize, as shown in Figure 1. This scale reflects the number of leaves developed by a plant, with values ranging from 10 to 19, where the tens digit represents the leaf stage, and the ones digit indicates the number of leaves.

Table 1 summarizes the detailed parameters of both datasets, including image count, resolution, total labeled objects, and distribution of objects across BBCH phases.

After labeling the images using the Label Studio environment with the polygon method, the datasets were divided into training and validation sets. The training sets comprised 70% of the data, while the validation sets comprised 30%, with stratification ensuring an even distribution of BBCH phases across the sets.

Examples from each dataset are shown in Figure 2, which displays RGB images of maize at different stages of development, captured under various weather conditions and at different times of the day. Figure 3 presents images taken by the individual lenses of the MicaSense RedEdge-MX DUAL multispectral camera, alongside a comparative image captured by the RGB camera.

3.2. Methods

In this section, we provide a detailed description of the methods used in our study. Figure 4 illustrates the general scheme of the proposed maize growth stage classification system.

Our system employs three primary classification approaches: classification on RGB

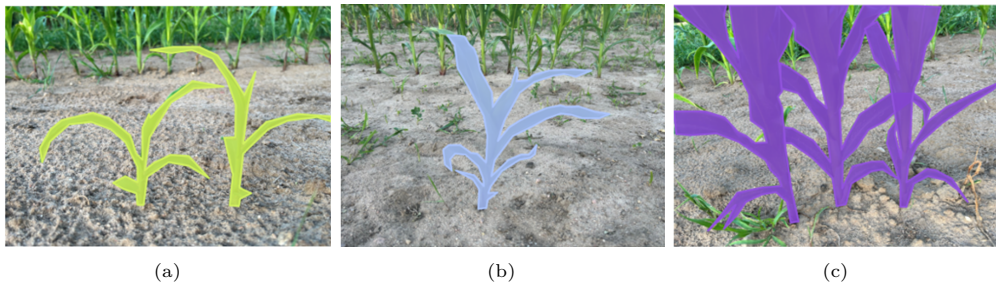


Fig. 1. Examples of determining BBCH scale values for maize and labelling them on images using 'the polygon method', (a) maize at stage 14 of the BBCH scale, (b) maize at stage 18 of the BBCH scale, (c) maize at stage 19 of the BBCH scale.

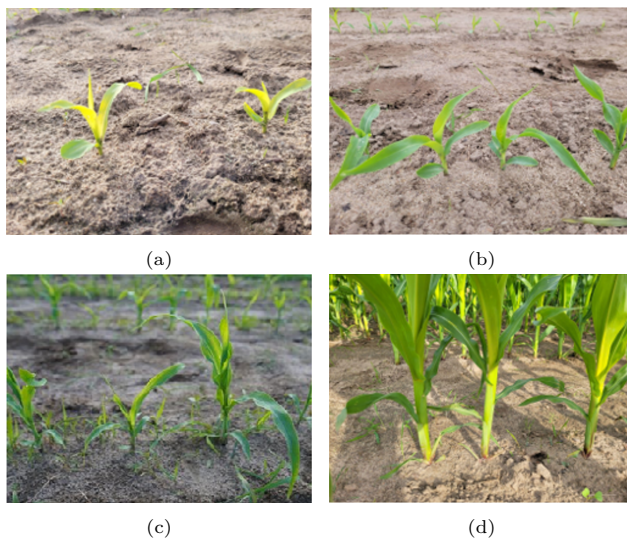


Fig. 2. Examples of RGB images showing maize at various stages of development, captured under different weather conditions and times of day.

images, classification on individual spectral channels, and classification using a voting classifier. Each approach uses deep learning models, including HTC_x101, HTC_r101, HTC_r50, and Mask2Former, trained on corresponding datasets consisting of RGB or individual spectral channels. The best-performing algorithm was selected based on the highest accuracy achieved during the training process.

The first approach involves the classification of RGB images using the best algorithm identified through model evaluation. In the second approach, classification is performed



Fig. 3. Examples of images captured from individual spectral channels and an RGB camera, demonstrating various bands and perspectives.

using 10 different models, each trained on a separate spectral channel, with the best-performing algorithm applied to each channel. The third approach utilizes a voting classifier, which combines the results from the 10 models trained on individual spectral channels. The final maize growth stage is determined based on the consensus of these models. A more detailed explanation of each method is provided in the subsequent sections of the article.

3.2.1. Models architectures

During our research, we used the following deep artificial neural network architectures for image analysis: HTC_x101, HTC_r101, HTC_r50, Mask2Former [8,10,15,29]. The highlights of these architectures we describe below.

HTC_x101, HTC_r101 and HTC_r50

HTC, or Hybrid Task Cascade, is an advanced model architecture used for simultaneous object detection and instance segmentation tasks. Proposed by SenseTime Research, HTC is renowned for its high accuracy in benchmarks like COCO [7]. The HTC architecture includes configurations like HTC_x101, HTC_r101, and HTC_r50, varying by the backbone used. HTC_x101 employs the ResNeXt-101 backbone, featuring 101 layers and grouped convolutions to enhance efficiency and accuracy [29]. HTC_r101 utilizes the ResNet-101 backbone, which includes a residual mechanism to improve gradient

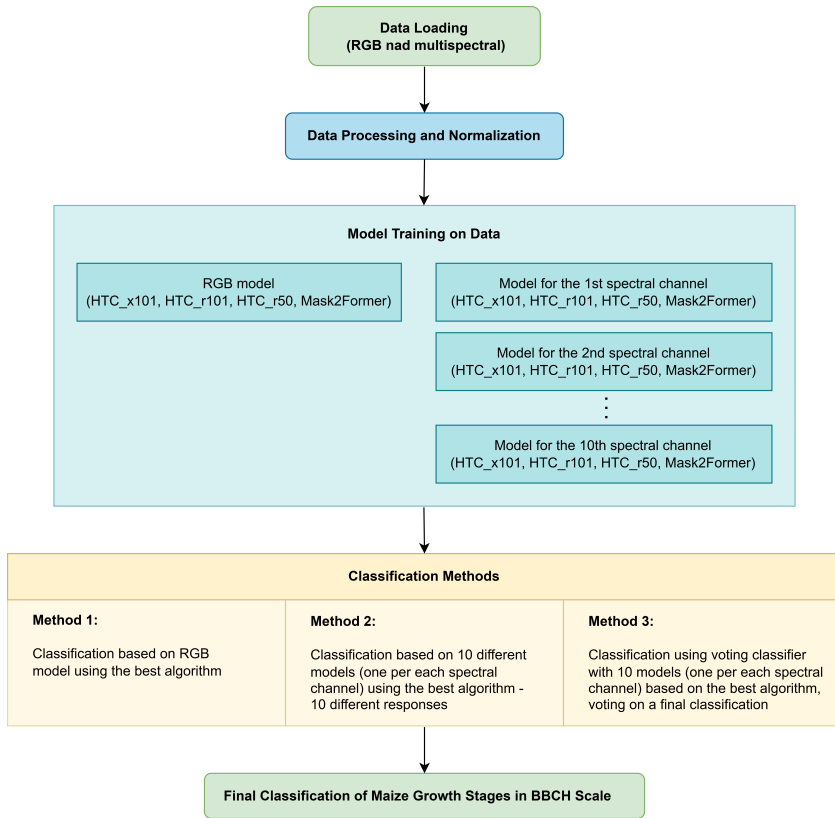


Fig. 4. Scheme of the proposed maize development stage classification system.

propagation and training [15]. HTC_r50, on the other hand, uses the ResNet-50 backbone, incorporating 50 layers with a residual mechanism that strikes a balance between performance and computational efficiency [18]. All three configurations use the Feature Pyramid Network (FPN) as the ‘neck’, which generates feature maps at different scales and enables efficient detection of objects across various sizes [18].

Regarding the ‘heads’, each model features similar components. The RPN (Region Proposal Network) head generates Region of Interest (ROI) proposals for further processing [23]. The ROI head conducts multi-stage bounding box regression and object classification to enhance detection accuracy [13]. The Mask head is responsible for instance segmentation, accurately delineating object contours within an image [14]. Finally, the Semantic head incorporates contextual information to improve performance in semantic segmentation tasks [35].

The HTC_x101, HTC_r101, and HTC_r50 models are trained using Stochastic Gradient Descent (SGD) optimization, with key hyperparameters including learning rate, momentum, and weight decay [6]. They incorporate advanced techniques such as RoIAlign (Region of Interest Align), which enhances the accuracy of object detection and segmentation. The use of multi-stage bounding box regression and semantic segmentation contributes to their high performance in benchmarks like COCO [14].

HTC's various configurations offer the flexibility to choose the right architecture depending on computational and precision requirements, making them versatile tools for advanced computer vision applications.

Mask2Former

Mask2Former is an advanced model architecture designed for various image segmentation tasks, including instance, semantic, and panoptic segmentation. It incorporates several innovations that enhance its performance over previous models.

The architecture employs a transformer-based backbone, utilizing a self-attention mechanism to efficiently process visual data and capture global dependencies within images, which is crucial for accurate segmentation [27]. Unlike traditional methods that generate masks for predefined regions, Mask2Former features dynamic mask prediction. This approach generates masks based on the context of each image, significantly increasing the model's flexibility and precision in mask generation [37].

Additionally, Mask2Former integrates instance, semantic, and panoptic segmentation tasks into a unified framework, allowing it to perform multiple types of segmentation without requiring structural modifications. The model also utilizes a query-based learning mechanism, where queries are dynamically updated during training to adapt to various scenarios, enhancing the quality of the generated masks [37].

Furthermore, Mask2Former employs advanced loss functions, such as focal loss, to effectively address issues with class imbalance in the training data, thereby improving overall performance and training efficiency [19].

Mask2Former is trained with finely tuned hyperparameters such as learning rate, weight decay, and the use of regularization techniques such as dropout and data augmentation [25]. The model also uses self-attention and query mechanisms for dynamic mask learning, which enhances its ability to accurately segment [37].

The architecture achieves high performance in benchmarks such as COCO, ADE20K, and Cityscapes, demonstrating superiority over previous segmentation methods [11, 36]. Its versatility and innovative approach to mask prediction make it a powerful tool in the field of image segmentation, for both research and practical applications [9, 37]. Mask2Former represents a significant step forward in segmentation model architectures, combining advanced visual data processing techniques with an efficient approach to dynamic mask prediction [34, 37].

Tab. 2. Hyperparameters and Configuration for HTC_r50, HTC_r101, HTC_x101, and Mask2Former Models

Model	Backbone	LR	Momentum	Weight Decay	Batch Size	Loss Functions
HTC_r50	ResNet-50	0.0003	0.9	0.0001	1	CrossEntropyLoss, SmoothL1Loss
HTC_r101	ResNet-101	0.0003	0.9	0.0001	1	CrossEntropyLoss, SmoothL1Loss
HTC_x101	ResNeXt-101	0.0003	0.9	0.0001	1	CrossEntropyLoss, SmoothL1Loss
Mask2Former	ResNet-50	0.0003	0.9	0.0001	1	CrossEntropyLoss, DiceLoss

3.2.2. Implementation, training and evaluation procedures

In our research, we used the PyTorch library to implement various models. We defined model architectures, specifying backbones, the Feature Pyramid Network (FPN) as the neck, and heads such as RPN, ROI, Mask, and Semantic Heads. The detailed configuration, including hyperparameters, backbones, and loss functions, is presented in Table 2.

To initialize these models, we employed weights pre-trained on the ImageNet dataset, leveraging knowledge embedded in large-scale datasets to enhance performance on our smaller labeled datasets.

To further improve robustness and generalization, we applied data augmentation techniques during training. These included resizing images to 1333x1000 pixels while maintaining their aspect ratio, random horizontal flipping with a probability of 50%, normalization using mean and standard deviation values for RGB channels, and padding to ensure image dimensions were divisible by 32. For segmentation tasks, masks were downsampled by a factor of 0.125. During testing, multi-scale augmentation was applied with resizing to 1333x1000 pixels, and flipping was disabled to maintain consistency in evaluation.

Models were trained using Stochastic Gradient Descent (SGD) to minimize the loss function. We used CrossEntropyLoss and SmoothL1Loss for most models, while for Mask2Former, we employed CrossEntropyLoss and DiceLoss.

Model performance was evaluated using the following measures: mAP (Mean Average Precision), accuracy, precision, sensitivity (recall), and IoU (Intersection over Union).

3.2.3. Description of the algorithm voting process

For the multispectral images, we noted that objects appeared at varying distances from the image edges due to different lens angles. This issue was particularly pronounced for maize at higher developmental stages (BBCH 14–19), where variations in angles altered object shapes, making it difficult to create composite images from different spectral

channels. While one lens might capture a leaf as relatively straight, another could record it as curved. These discrepancies complicated the superimposition of images from different spectral channels into a single composite image.

- **image_1:** *class_1* – [polygon_1: area, polygon_2: area], *class_2* – [polygon_1: area]
- **image_2:** *class_3* – [polygon_1: area]

The final results obtained take the following form:

- image_1: class_6
- image_2: class_5

To address this issue and fully utilize all spectral channels of the RedEdge-MX camera, we employed a voting classifier composed of 10 individual models. Each model was trained on images from a specific spectral channel using the HTC algorithm with a ResNeXt101 backbone, as this configuration consistently provided the best performance. After training, the predictions from these models were aggregated by plant identifier (file_id). For each plant, the final class was determined by majority voting. In cases where there was a tie, the class with the highest average polygon score was selected.

The aggregated predictions yield results in the following format:

- **image_1:** *class_1* – [polygon_1: score, polygon_2: score], *class_2* – [polygon_1: score]
- **image_2:** *class_3* – [polygon_1: score]

The voting classifier aggregates these results and assigns the class with the highest score to an image. The final results are presented as:

- image_1: class_6
- image_2: class_5

During validation, the markings are aggregated into classes using defined polygons for each spectral channel. An image may contain multiple polygons for various classes. To determine a single class per image, we selected the polygon with the largest area. The initial data structure for each spectral channel is as follows:

The voting classifier predicts the class for each plant based on images taken from different angles by separate cameras. By aggregating predictions from all spectral channels, we can achieve more robust and accurate classifications, even when individual models produce inconsistent results due to variations in object appearance.

Figure 5 illustrates the workflow of the voting classifier, from training individual models on spectral channels to aggregating predictions and selecting the final class.

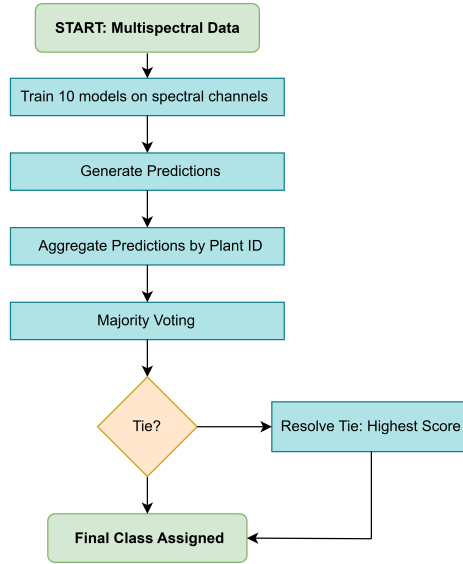


Fig. 5. Workflow of the voting classifier: training models on spectral channels, making predictions, and aggregating results through majority voting.

4. Experimental results

In this chapter, we present the experimental results for BBCH scale classification of developmental stages. We compare results obtained from RGB data, multispectral data, and the voting method developed using multispectral data.

4.1. Algorithms results on multispectral data

During the experimental study, we noticed differences in classification performance between the different algorithms used for the training process of deep neural networks and between the different spectral channels on which the algorithms were trained.

In Table 3 we present the classification results for the spectral channels recorded for each of the algorithms tested. The model trained on the data from channel 10 clearly differs in classification efficiency from the models trained on the other spectral channels. The table presented illustrates the effectiveness of each classification method on multispectral data.

From the analysis, we conclude that for the HTC algorithm with ResNet101, spectral channel 07 (red edge 705 (10)) yielded the best results (in the notation of channels in the optical specifications of cameras and multispectral sensors, in the description,

Tab. 3. Classification results for each of the spectral channels analysed using individual deep learning algorithms.

channel	accuracy				precision			
	HTC_r101	HTC_r50	HTC_x101	Mask2Former	HTC_r101	HTC_r50	HTC_x101	Mask2Former
01	0.486957	0.552174	0.547826	0.539130	0.486508	0.592528	0.582530	0.550819
02	0.568282	0.559471	0.559471	0.555066	0.594116	0.563848	0.586652	0.611719
03	0.548246	0.550661	0.530702	0.508772	0.587350	0.573304	0.570517	0.567900
04	0.567100	0.549784	0.541126	0.510823	0.603094	0.587433	0.563036	0.502541
05	0.575893	0.580357	0.558036	0.531250	0.596516	0.605068	0.578664	0.545954
06	0.547085	0.551570	0.581081	0.520179	0.570063	0.577274	0.591761	0.545100
07	0.582609	0.565217	0.569565	0.495652	0.606866	0.581716	0.593181	0.566532
08	0.538117	0.522321	0.540179	0.486607	0.575040	0.556559	0.572584	0.504101
09	0.551570	0.569507	0.549550	0.524664	0.586433	0.584708	0.561682	0.540334
10	0.219298	0.223684	0.214912	0.232456	0.161993	0.175143	0.162701	0.174677

channel	recall				F1-score			
	HTC_r101	HTC_r50	HTC_x101	Mask2Former	HTC_r101	HTC_r50	HTC_x101	Mask2Former
01	0.546268	0.608676	0.582942	0.547557	0.482994	0.558260	0.543616	0.506817
02	0.611719	0.619491	0.589024	0.599922	0.563584	0.554428	0.564948	0.543282
03	0.593651	0.591402	0.566799	0.549762	0.552418	0.557723	0.521853	0.509900
04	0.626295	0.600151	0.566362	0.557919	0.566393	0.555305	0.539768	0.494669
05	0.621161	0.630494	0.587758	0.571906	0.585859	0.589184	0.552485	0.523784
06	0.571536	0.585762	0.610729	0.558193	0.546741	0.556209	0.591596	0.521969
07	0.627316	0.587195	0.602076	0.541441	0.587952	0.560104	0.570215	0.474963
08	0.561341	0.553343	0.555533	0.498973	0.542855	0.529349	0.552871	0.476158
09	0.611539	0.596285	0.565288	0.541253	0.557820	0.570818	0.544688	0.527224
10	0.215904	0.228801	0.217906	0.257787	0.175971	0.187777	0.176843	0.185934

Tab. 4. Best classification results for each spectral channel.

Spectral channel	Wavelength [nm]	Best algorithm	Metrics
01 (coastal blue)	444 (28)	HTC (ResNet50)	accuracy, precision, recall, F1-score
02 (blue)	475 (32)	HTC (ResNet101)	accuracy, recall
		Mask2Former	precision
		HTC (ResNeXt101)	F1-score
03 (green)	531 (14)	HTC (ResNet50)	accuracy, F1-score
		HTC (ResNet101)	precision, recall
04 (green)	560 (27)	HTC (ResNet101)	accuracy, precision, recall, F1-score
05 (red)	650 (16)	HTC (ResNet50)	accuracy, precision, recall, F1-score
06 (red)	668 (14)	HTC (ResNeXt101)	accuracy, precision, recall, F1-score
07 (red edge)	705 (10)	HTC (ResNet101)	accuracy, precision, recall, F1-score
08 (red edge)	717 (12)	HTC (ResNet101)	precision, recall
		HTC (ResNeXt101)	accuracy, F1-score
		HTC (ResNet50)	accuracy, F1-score
09 (red edge)	740 (18)	HTC (ResNet101)	precision, recall
		Mask2Former	accuracy, recall
10 (NIR)	842 (57)	Mask2Former	accuracy, recall
		HTC (ResNet50)	precision, F1-score

e.g. 444 (28), the first value (444 nm) refers to the central wavelength, and the value in parentheses (28 nm) represents the bandwidth; so, in this example, the spectral range is $430 \pm (28/2)$ nm). With ResNet50, channel 05 (red 650 (16)) performed best, and with ResNeXt101, channel 06 (red 668 (14)) was optimal. For Mask2Former, channel 02 (blue 475 (32)) provided the best results. Different algorithms thus achieve optimal performance with different spectral channels. Table 4 summarizes the best-performing algorithm for each spectral channel.

The classification results for spectral channel 10 (NIR 842 (57)) show a significant

Tab. 5. Comparison of the classification results across different algorithms on RGB images with those obtained using multispectral images and voting classifier approach. The analysis uses accuracy, precision, recall, and F1-score measures.

Model	Approach	accuracy	precision	recall	F1-score
HTC_r101	voting classifier	0.651466	0.684678	0.686132	0.643477
	RGB	0.706667	0.460286	0.496550	0.465737
HTC_r50	voting classifier	0.661238	0.690290	0.699036	0.659793
	RGB	0.680000	0.424074	0.404996	0.406152
HTC_x101	voting classifier	0.657980	0.663664	0.676181	0.652508
	RGB	0.760000	0.525599	0.580694	0.524339
Mask2Former	voting classifier	0.625407	0.615539	0.630647	0.592380
	RGB	0.800000	0.596212	0.660516	0.600132

deviation in accuracy compared to other channels. This indicates that classification using only this spectral channel has the lowest object classification efficiency among the analyzed bands.

4.2. Algorithms results on RGB data

In addition to the multispectral studies, we conducted experiments with RGB images. We observed variations in classification performance among different algorithms when trained on RGB images.

The Mask2Former algorithm achieved the best RGB image classification results in terms of accuracy, precision, recall, and F1-score, making it the most effective model for RGB among those studied. However, the HTC algorithm with a ResNeXt101 backbone ranked second, followed by HTC with a ResNet101 backbone in third place and HTC with a ResNet50 backbone in fourth place.

4.3. Results obtained in the voting process

Table 5 summarizes the classification accuracy results for all tested algorithms, comparing RGB models with those using our voting method based on individual spectral channels. The evaluation measures used are accuracy, precision, recall, and F1-score.

The results show that the HTC algorithm with a ResNet101 backbone, trained on RGB data, achieves a higher accuracy measure than the voting classifier based on all spectral channels. However, for the precision, recall, and F1-score measures, the voting classifier achieves better results.

For the HTC algorithm with a ResNet50 backbone, the model trained on RGB data outperforms our voting method based on single spectral channel models in terms of accuracy. However, the voting method performs better in precision, recall and F1-score measures.

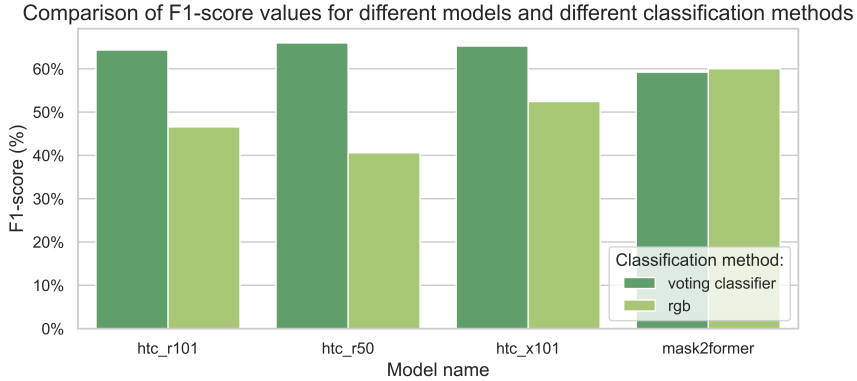


Fig. 6. Visualisation comparing the classification results between algorithms trained on RGB images and those using multispectral images with a voting classifier approach.

The HTC algorithm with a ResNeXt101 backbone, trained on RGB data, achieves a higher accuracy measure than the voting classifier based on all spectral channels. However, for the precision, recall, and F1-score measures, the voting classifier achieves better results.

For the Mask2Former algorithm, the model trained on RGB data outperforms our voting method based on single spectral channel models in terms of accuracy, recall, and F1-score. However, the voting method performs better in the precision measure.

In turn, in Figure 6, we present a graphical summary of the comparative data for the model trained on RGB images and the method using a voting classifier. We used the F1-score measure for comparison.

For HTC with ResNet101, ResNet50 and ResNeXt101, the voting classifier outperforms the RGB-trained model. In contrast, for the Mask2Former algorithm, the RGB-trained model outperforms the voting classifier.

Key conclusions include that the voting classifier based on single spectral channels performed better than the RGB classifier. Single-channel models generally show lower quality compared to the RGB classifier trained on three-channel images; however, the potential of voting techniques to enhance predictions improved the overall performance.

4.4. Learning curves

In this chapter, we present the learning curves recorded during the training of each model (the complete set of curves is available in the repository [26]). Below are the curves for each algorithm, illustrating training performance across different spectral channels.

Figure 7 compares the performance of the HTC_r101 model trained on individual

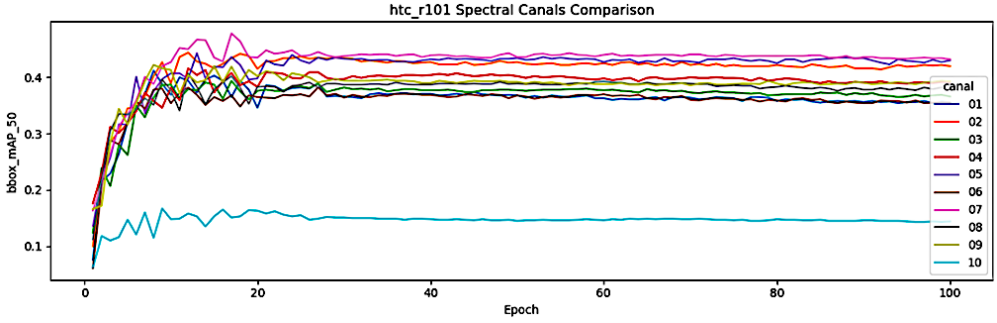


Fig. 7. Comparison of the training curves of the HTC_r101 model trained on individual spectral channels.

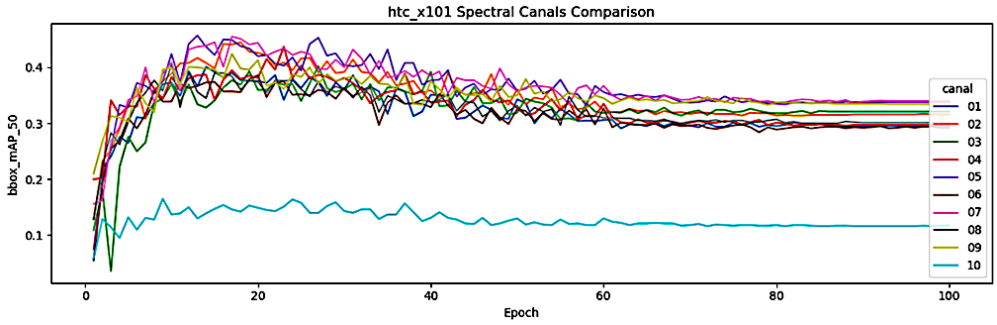


Fig. 8. Comparison of the training curves of the HTC_x101 model trained on individual spectral channels.

spectral channels over 100 epochs. The horizontal axis shows epochs, and the vertical axis displays object detection performance expressed by the measure `bbox_mAP_50`. This measure denotes the Bounding Box Mean Average Precision at IoU 50%. It considers detections as correct if the Intersection over Union (IoU) between the predicted and ground truth bounding boxes is at least 50%. Channel 10 exhibits the lowest performance, with `bbox_mAP_50` values around 0.1 after 20 epochs. In contrast, other channels show better results, with `bbox_mAP_50` values around 0.4 and minor fluctuations. Channel 7 achieves the highest efficiency, with a maximum `bbox_mAP_50` of about 0.47, maintaining the best performance among all individual spectral channels.

Figure 8 presents the learning curves for the HTC_x101 model across spectral channels. Channel 10 shows the lowest performance, with `bbox_mAP_50` around 0.4 initially, dropping to 0.35 between 40-60 epochs, and stabilizing above 0.3 afterward. Channels 05 and 07 perform best, with very similar results.

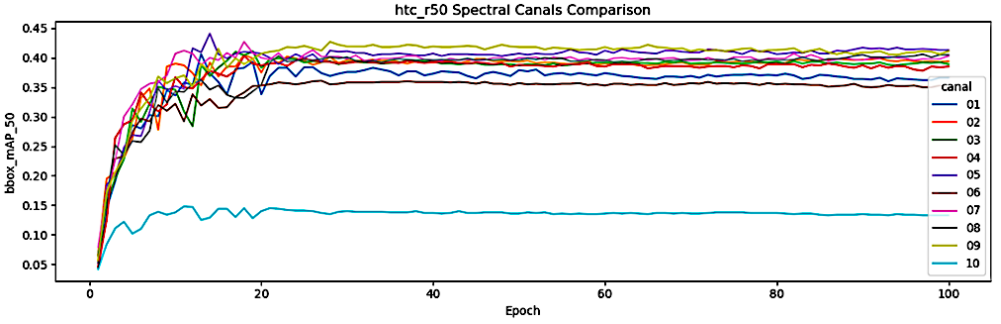


Fig. 9. Comparison of the training curves of the HTC_r50 model trained on individual spectral channels.

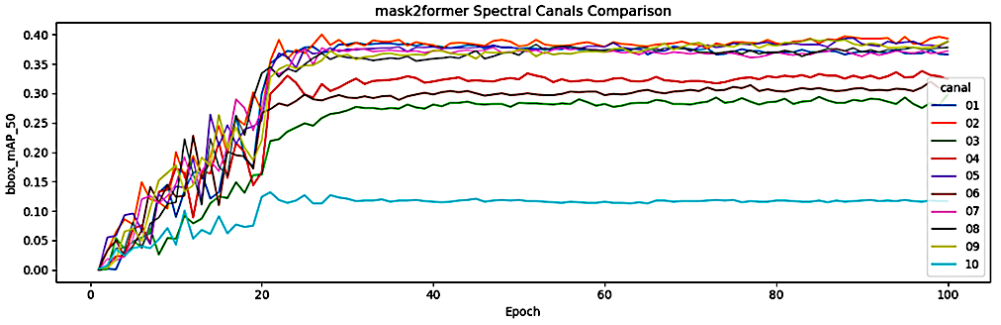


Fig. 10. Comparison of the training curves of the Mask2Former model trained on individual spectral channels.

For the HTC_r50 algorithm (Figure 9), the learning curves are similar to those of HTC_r101. Channel 10 shows the lowest performance, with `bbox_mAP_50` values around 0.125 after 10 epochs, remaining stable with slight fluctuations up to 100 epochs. The values for channels 01-09 are more stable after reaching a maximum, indicating their better performance in detecting objects with the HTC_r50 model.

For the Mask2Former algorithm (Figure 10), the lowest results can also be observed for channel 10, where `bbox_mAP_50` oscillates around 0.125 after the first 20 epochs and remains at this level until the end of the observation, i.e. the end of 100 epochs. The remaining channels reach higher `bbox_mAP_50` values, but are no longer such a compact group as in the previous algorithms. Of all the channels, the highest `bbox_mAP_50` values are reached by channel 02.

The results show that different algorithms reach peak `bbox_mAP_50` values for various spectral channels, with most channels stabilizing after 20 epochs (except HTC_x10,

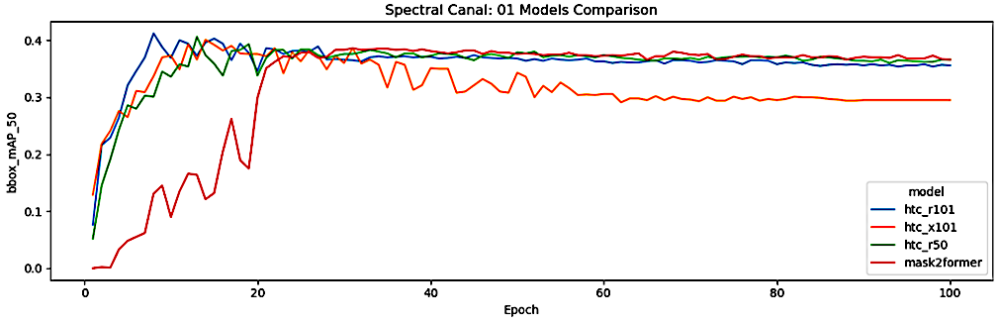


Fig. 11. Comparison of the training curves of different models on the dataset from the first spectral channel.

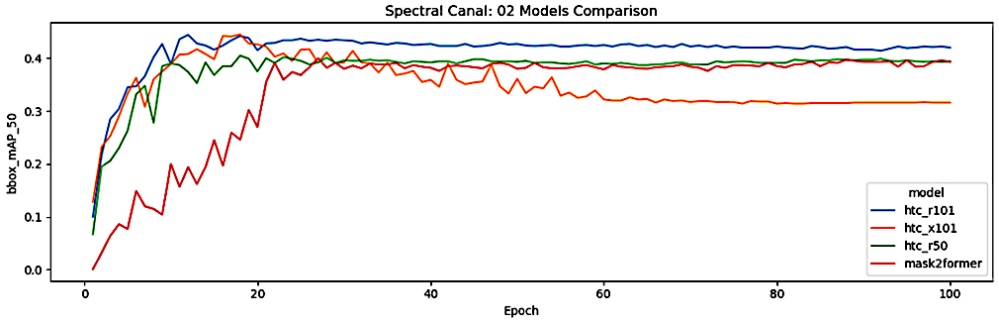


Fig. 12. Comparison of the training curves of different models on the dataset from the second spectral channel.

stabilizing after 60 epochs). Channel no. 10 consistently yields the poorest learning results.

We now examine models trained on individual spectral channels. Figures 11 and 12 compare the performance of four models on datasets from two spectral channels over 100 epochs. Additional graphs for other channels are available in the repository [26]. The horizontal axis represents epochs, and the vertical axis shows `bbox_mAP_50`, indicating detection accuracy.

Figure 11 shows results for spectral channel 01. The Mask2Former model achieves the highest and most stable `bbox_mAP_50` of around 0.4 after 20 epochs. HTC_r101 and HTC_r50 models also stabilize around 0.4 but perform slightly worse. HTC_x101 starts strong but declines after 20 epochs, stabilizing around 0.3, indicating lower performance.

Analyzing spectral channel 02 (Figure 12), HTC_r101 and HTC_r50 rapidly increase

bbox_mAP_50 to around 0.4 and stabilize. Mask2Former also rises to about 0.4 within 20 epochs. HTC_x101 initially increases to 0.4 but then drops to around 0.3. HTC_r101 achieves the highest performance, while HTC_x101 shows the lowest one.

For spectral channel 03 (see the repository [26]), HTC_r101 and HTC_r50 rapidly increase in bbox_mAP_50 during the first 10 epochs, stabilizing around 0.4. Mask2Former rises swiftly in the first 20 epochs, then stabilizes at about 0.3. HTC_x101 initially increases to 0.4, but then declines and stabilizes around 0.3. The highest performance is achieved by the HTC_r50 model, while the Mask2Former model performs the worst.

For spectral channel 04 (see the repository [26]), HTC_r101 and HTC_r50 rapidly increase bbox_mAP_50 to around 0.4 and stabilize. Mask2Former rises to 0.3 within 20 epochs. HTC_x101 also reaches 0.4 initially but declines to around 0.3. HTC_r50 shows the highest performance, while Mask2Former performs the worst.

By analysing the results for spectral channel 05 (see the repository [26]) it can be discovered that HTC_r101 and HTC_r50 rapidly increase bbox_mAP_50 to around 0.4 and stabilize. Mask2Former also stabilizes at about 0.4 after 20 epochs. HTC_x101 rises to 0.4 initially but declines to around 0.3. HTC_r101 achieves the highest results, followed by HTC_r50, while HTC_x101 shows the lowest performance.

For spectral channel 06 (see the repository [26]), the HTC_r101 and HTC_r50 algorithms quickly increase bbox_mAP_50 to around 0.4 and then stabilize. Mask2Former rises swiftly in the first 20 epochs and stabilizes at about 0.3. HTC_x101 reaches 0.38 initially but declines to around 0.3. HTC_r101 achieves the highest results, followed by HTC_r50, with HTC_x101 performing the worst.

For spectral channel 07 (see the repository [26]), the HTC_r101 and HTC_r50 algorithms quickly increase bbox_mAP_50 to 0.45 and 0.4, respectively, and stabilize. Mask2Former rises rapidly in the first 20 epochs and stabilizes at around 0.4. HTC_x101 reaches 0.45 initially but declines to 0.3. HTC_r101 performs the best, followed by HTC_r50, with HTC_x101 showing the lowest results.

For spectral channel 08 (see the repository [26]), the HTC_r101 and HTC_r50 algorithms quickly increase bbox_mAP_50 to around 0.4 and stabilize. Mask2Former also rises rapidly and stabilizes at about 0.4. HTC_x101 reaches 0.4 initially but declines to 0.3. HTC_r50 performs the best, while HTC_x101 shows the lowest results.

For spectral channel 09 (see the repository [26]), the HTC_r101 and HTC_r50 algorithms show a rapid rise in bbox_mAP_50 to around 0.4, stabilizing at this level. Mask2Former also reaches about 0.4 after 20 epochs. The HTC_x101 model initially increases to 0.4 but declines to 0.3. HTC_r50 achieves the highest performance, followed by HTC_r101 and Mask2Former, with HTC_x101 showing the lowest results.

For spectral channel 10 (see the repository [26]), the HTC_r101 and HTC_r50 algorithms quickly rise in bbox_mAP_50 to about 0.15, stabilizing there. The Mask2Former

also increases to around 0.125. The HTC_x101 model initially reaches 0.4 but declines to 0.125. HTC_r101 achieves the highest performance, followed by HTC_r50 and Mask2Former, with HTC_x101 showing the lowest results.

5. Conclusions

We have presented an innovative approach for automating maize growth monitoring using image analysis and artificial intelligence techniques. Our method employs deep neural networks to analyze RGB and multispectral images, along with an additional voting classifier. The goal was to efficiently detect and classify maize developmental stages based on the BBCH scale, enabling automatic monitoring of plant development phases and presenting the results on large scale, e.g., in the form of a map.

Our results demonstrate a highly automated method for detecting and classifying maize developmental stages with plant-level accuracy. Compared to manual methods, our solution significantly accelerates the classification process through real-time image analysis on a field robot, allowing for efficient maize growth stage monitoring. While UAV-based approaches cover larger field areas per image, our method offers greater precision at the individual plant level. By integrating advanced image analysis and deep learning algorithms, our solution achieves high automation and accuracy. A literature review confirms the novelty of our method.

The models in our study were trained on proprietary datasets of labeled maize images at various BBCH developmental stages (10–19), captured in both RGB and multispectral spectra. This allowed comprehensive training separately on RGB and each spectral channel. Furthermore, we evaluated various deep learning architectures to assess detection and classification performance across different training datasets and algorithms.

To improve the performance of the model we employed pre-trained backbones such as ResNeXt-101, ResNet-101, and ResNet-50, initialized with ImageNet weights. Fine-tuning these models on our labeled datasets leveraged the rich feature representations learned from large-scale datasets, improving accuracy and robustness. Additionally, we applied data augmentation techniques such as resizing, random horizontal flipping, normalization, padding, and mask downscaling, further enhancing model performance.

Single-channel models generally performed worse than RGB models due to their limited spectral information. However, the voting classifier improved prediction quality.

Comparing the results of different algorithms and training sets, we observed that HTC_r50, HTC_r101, and HTC_x101 achieved higher precision, recall, and F1-score when trained on single spectral channels with a voting classifier than on RGB data. For Mask2Former, precision was slightly higher with the voting classifier, while accuracy, recall, and F1-score were better for RGB data.

For RGB images, the best overall performance across all measures was achieved by Mask2Former, followed by HTC_x101, HTC_r101, and HTC_r50. For single spectral

channels with the voting classifier, HTC_r50 achieved the highest accuracy and F1-score, followed by HTC_x101, HTC_r101, and Mask2Former. Regarding precision and recall, HTC_r50 performed best, followed by HTC_r101, HTC_x101, and Mask2Former.

According to the accuracy measure, the best performance was achieved by the model Mask2Former trained on RGB data, while in terms of precision, recall, and F1-score, HTC_r50 trained on individual spectral channels with a voting classifier performed best.

Another key finding was the identification of optimal spectral channels for maize growth stage classification. For HTC_r101, channel 07 yielded the best results. For HTC_x101, channels 05 and 07 were optimal. HTC_r50 performed best on channel 05, while Mask2Former achieved the best results on channel 02.

Our solution enables precise plant condition tracking, supporting decision-making in precision agriculture. Moreover, our method can be adapted to other crops by developing appropriate datasets and retraining deep neural networks.

6. Discussion of limitations

A multispectral camera with multiple lenses captures spectral channels from different angles, causing variability in plant shapes across the channels. To mitigate this, a single-lens system should be used. We explored this approach in our other research.

The models were trained on proprietary datasets with labeled maize images at various growth stages. The limited diversity of the dataset may affect the models' ability to generalize to real-world field conditions. Expanding the dataset with images from different locations and conditions can improve model performance and robustness.

Our solution was developed specifically for monitoring the developmental stages of maize. Adapting the method to other crops requires creating new datasets and retraining the models. However, validating the effectiveness of the developed solution for maize offers promising prospects for successful application to other crops as well.

Acknowledgement

This research was supported by the FITOEXPORT project (Contract No. GOSPOSTRATEG 1/385957/5/NCBR/2018), funded by the National Centre for Research and Development (NCBR). The authors express their gratitude for the support provided.

References

- [1] S. Bansal, M. Singh, S. Barda, N. Goel, and M. Saini. PA-RDFKNet: Unifying Plant Age Estimation through RGB-Depth Fusion and Knowledge Distillation. *IEEE Transactions on AgriFood Electronics*, 2(2):226–235, 2024. doi:[10.1109/TAFE.2024.3418818](https://doi.org/10.1109/TAFE.2024.3418818).
- [2] A. Bera, D. Bhattacharjee, and O. Krejcar. An attention-based deep network for plant disease classification. *Machine Graphics and Vision*, 33(1):47–67, 2024. doi:[10.22630/MGV.2024.33.1.3](https://doi.org/10.22630/MGV.2024.33.1.3).

- [3] A. Bera, D. Bhattacharjee, and O. Krejcar. PND-Net: Plant Nutrition Deficiency and Disease Classification Using Graph Convolutional Network. *Scientific Reports*, 14(1):15537, 2024. doi:[10.1038/s41598-024-66543-7](https://doi.org/10.1038/s41598-024-66543-7).
- [4] A. Bera, O. Krejcar, and D. Bhattacharjee. RAFA-Net: Region Attention Network for Food Items and Agricultural Stress Recognition. *IEEE Transactions on AgriFood Electronics*, pp. 1–13, 2024. doi:[10.1109/TAFE.2024.3466561](https://doi.org/10.1109/TAFE.2024.3466561).
- [5] L. Bompani et al. Accelerating Image-based Pest Detection on a Heterogeneous Multicore Microcontroller. *IEEE Transactions on AgriFood Electronics*, 2(2):170–180, 2024. doi:[10.1109/TAFE.2024.3451888](https://doi.org/10.1109/TAFE.2024.3451888).
- [6] L. Bottou. Stochastic Gradient Descent Tricks. In: G. Montavon, G. B. Orr, and K.-R. Müller, eds., *Neural Networks: Tricks of the Trade*, vol. 7700 of *Lecture Notes in Computer Science*, pp. 421–436. Springer, second edition edn., 2012. doi:[10.1007/978-3-642-35289-8_25](https://doi.org/10.1007/978-3-642-35289-8_25).
- [7] K. Chen, J. Pang, J. Wang, Y. Xiong, X. Li, et al. Hybrid Task Cascade for Instance Segmentation. In: *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4974–4983. Long Beach, CA, USA, June 16-20 2019. doi:[10.1109/CVPR.2019.00511](https://doi.org/10.1109/CVPR.2019.00511).
- [8] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, et al. MMDetection: Open MMLab Detection Toolbox and Benchmark. *arXiv*, 2019. ArXiv:1906.07155. doi:[10.48550/arXiv.1906.07155](https://doi.org/10.48550/arXiv.1906.07155).
- [9] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. In: *Proc. European Conference on Computer Vision (ECCV)*, pp. 801–818. Springer, Munich, Germany, September 8-14 2018. doi:[10.1007/978-3-030-01234-2_49](https://doi.org/10.1007/978-3-030-01234-2_49).
- [10] B. Cheng, A. Schwing, and A. Kirillov. Masked-attention Mask Transformer for Universal Image Segmentation. In: *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1280–1289. New Orleans, LA, USA, 2022. doi:[10.1109/CVPR52688.2022.01280](https://doi.org/10.1109/CVPR52688.2022.01280).
- [11] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, et al. The Cityscapes Dataset for Semantic Urban Scene Understanding. In: *Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3213–3223. IEEE, Las Vegas, NV, USA, June 27-30 2016. doi:[10.1109/CVPR.2016.350](https://doi.org/10.1109/CVPR.2016.350).
- [12] S. Figiel. Development of Artificial Intelligence and Potential Impact of Its Applications in Agriculture on Labor Use and Productivity. *Zagadnienia Ekonomiki Rolnej / Problems of Agricultural Economics*, 373(4):5–21, 2022. doi:[10.30858/zer/153583](https://doi.org/10.30858/zer/153583).
- [13] R. Girshick. Fast R-CNN. In: *IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448. Santiago, Chile, December 7-13 2015. doi:[10.1109/ICCV.2015.169](https://doi.org/10.1109/ICCV.2015.169).
- [14] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In: *IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988. Venice, Italy, October 22-29 2017. doi:[10.1109/ICCV.2017.322](https://doi.org/10.1109/ICCV.2017.322).
- [15] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778. IEEE, Las Vegas, NV, USA, 2016. doi:[10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90).
- [16] A. Kamilaris and F. X. Prenafeta-Boldú. Deep Learning in Agriculture: A Survey. *Computers and Electronics in Agriculture*, 147:70–90, 2018. doi:[10.1016/j.compag.2018.02.016](https://doi.org/10.1016/j.compag.2018.02.016).
- [17] P. D. Lancashire, H. Bleiholder, T. van den Boom, P. Langelüddecke, R. Stauss, et al. A Uniform Decimal Code for Growth Stages of Crops and Weeds: BBCH Monograph. *Annals of Applied Biology*, 119(3):561–601, 1991. doi:[10.1111/j.1744-7348.1991.tb04895.x](https://doi.org/10.1111/j.1744-7348.1991.tb04895.x).

- [18] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, et al. Feature Pyramid Networks for Object Detection. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944. Honolulu, HI, USA, July 21–26 2017. doi:[10.1109/CVPR.2017.106](https://doi.org/10.1109/CVPR.2017.106).
- [19] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal Loss for Dense Object Detection. In: *Proc. 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988. IEEE, Venice, Italy, October 22–29 2017. doi:[10.1109/ICCV.2017.324](https://doi.org/10.1109/ICCV.2017.324).
- [20] M. Liu, W.-H. Su, and X.-Q. Wang. Quantitative Evaluation of Maize Emergence Using UAV Imagery and Deep Learning. *Remote Sensing*, 15(8):1979, 2023. doi:[10.3390/rs15081979](https://doi.org/10.3390/rs15081979).
- [21] U. Meier, H. Bleiholder, L. Buhr, C. Feller, H. Hack, et al. The BBCH system for coding the phenological growth stages of plants – history and publications. *Journal für Kulturpflanzen*, 61(2):41–52, 2009. doi:[10.5073/JfK.2009.02.01](https://doi.org/10.5073/JfK.2009.02.01).
- [22] E. F. I. Raj, M. Appadurai, and K. Athiappan. Precision farming in modern agriculture. In: *Smart Agriculture Automation Using Advanced Technologies: Data Analytics and Machine Learning, Cloud Architecture, Automation and IoT*, pp. 61–87. Springer Singapore, Singapore, 2022. doi:[10.1007/978-981-16-6124-2_4](https://doi.org/10.1007/978-981-16-6124-2_4).
- [23] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In: *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 91–99. Montreal, Canada, December 7–12 2015. doi:[10.1109/TPAMI.2016.2577031](https://doi.org/10.1109/TPAMI.2016.2577031).
- [24] T. A. Shaikh, T. Rasool, and F. R. Lone. Towards Leveraging the Role of Machine Learning and Artificial Intelligence in Precision Agriculture and Smart Farming. *Computers and Electronics in Agriculture*, 198:107119, 2022. doi:[10.1016/j.compag.2022.107119](https://doi.org/10.1016/j.compag.2022.107119).
- [25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, June 2014. doi:[10.5555/2627435.2670313](https://doi.org/10.5555/2627435.2670313).
- [26] J. Stypułkowska. bbch-maize-learning-curves. <https://github.com/JustynaStypulkowska/bbch-maize-learning-curves>, 2024. GitHub repository.
- [27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, et al. Attention Is All You Need. In: *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pp. 6000–6010. Curran Associates, Inc., Long Beach, CA, USA, December 4–9 2017. doi:[10.5555/3295222.3295349](https://doi.org/10.5555/3295222.3295349).
- [28] J. Wu, V. Abolghasemi, M. H. Anisi, U. Dar, A. Ivanov, et al. Strawberry Disease Detection Through an Advanced Squeeze-and-Excitation Deep Learning Model. *IEEE Transactions on Agri-Food Electronics*, 2(2):259–267, 2024. doi:[10.1109/TAFE.2024.3412285](https://doi.org/10.1109/TAFE.2024.3412285).
- [29] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated Residual Transformations for Deep Neural Networks. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5987–5995. IEEE, Honolulu, HI, USA, July 21–26 2017. doi:[10.1109/CVPR.2017.634](https://doi.org/10.1109/CVPR.2017.634).
- [30] X. Xu, L. Wang, M. Shu, X. Liang, A. Ghafoor, et al. Detection and Counting of Maize Leaves Based on Two-Stage Deep Learning with UAV-Based RGB Image. *Remote Sensing*, 14(21):5388, 2022. doi:[10.3390/rs14215388](https://doi.org/10.3390/rs14215388).
- [31] Y. Yao, J. Yue, Y. Liu, H. Yang, H. Feng, et al. Classification of Maize Growth Stages Based on Phenotypic Traits and UAV Remote Sensing. *Agriculture*, 14(7):1175, 2024. doi:[10.3390/agriculture14071175](https://doi.org/10.3390/agriculture14071175).
- [32] D. Yu, Y. Zha, Z. Sun, et al. Deep Convolutional Neural Networks for Estimating Maize Above-Ground Biomass Using Multi-Source UAV Images: A Comparison with Traditional Machine Learning Algorithms. *Precision Agriculture*, 24(1):92–113, 2023. doi:[10.1007/s11119-022-09932-0](https://doi.org/10.1007/s11119-022-09932-0).

- [33] X. Zan, X. Zhang, Z. Xing, W. Liu, X. Zhang, et al. Automatic Detection of Maize Tassels from UAV Images by Combining Random Forest Classifier and VGG16. *Remote Sensing*, 12(18):3049, 2020. doi:[10.3390/rs12183049](https://doi.org/10.3390/rs12183049).
- [34] H. Zhang, Y. Liu, K. Ma, H. Su, S. Li, et al. Transformers for Image Segmentation. In: *Proc. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1251–1260. IEEE, Long Beach, CA, USA, June 16-20 2019. doi:[10.1109/CVPR.2019.00130](https://doi.org/10.1109/CVPR.2019.00130).
- [35] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid Scene Parsing Network. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2881–2890. Honolulu, HI, USA, July 21-26 2017. doi:[10.1109/CVPR.2017.660](https://doi.org/10.1109/CVPR.2017.660).
- [36] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, et al. Scene Parsing through ADE20K Dataset. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5122–5130. IEEE, Honolulu, HI, USA, July 21-26 2017. doi:[10.1109/CVPR.2017.544](https://doi.org/10.1109/CVPR.2017.544).
- [37] X. Zhu, Z. Zhang, Z. Li, X. Wang, J. Sun, et al. Mask2Former: A Transformer Architecture for Universal Image Segmentation. In: *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7823–7833. IEEE, New Orleans, LA, USA, June 19-24 2022. doi:[10.1109/CVPR52688.2022.00767](https://doi.org/10.1109/CVPR52688.2022.00767).

