Universiteit Leiden

Master Computer Science

Computer Vision-Based Prediction of Seed Germination Rates Across Species

Name:	Mojhde Hamidi
Student ID:	s4023803
Date:	[dd/mm/yyyy]
Specialisation:	Artificial intelligence
1st supervisor:	Lu Cao
2nd supervisor:	Joost Batenburg

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Contents

Acknowledgements

I would like to begin by expressing my deepest gratitude to my first supervisor, Lu Cao, whose invaluable guidance and thoughtful feedback have been a source of support throughout this journey. Her calm helped me stay focused, and I have learned immensely from her expertise and insights, which have profoundly shaped my understanding and growth.
I am also sincerely grateful to my second supervisor, Joost Betenburg, for his invaluable advice and support. My heartfelt thanks go as well to my colleagues at Growy, especially Mert Imre, who supervised me throughout the entire duration of my internship and provided unwavering help and guidance.
Finally, to my family and friends—thank you for your patience, understanding, and constant encouragement. Your belief in me has been my greatest motivation, and I could not have reached this point without your support.

Abstract

Rapid urbanization and limited arable land have made *vertical farming* a vital solution for sustainable food production. However, despite its promise, vertical farming faces significant operational challenges—particularly in the *automated monitoring of germination*—a crucial phase that significantly impacts overall yield and resource efficiency. Manual inspection in dense, multi-layer cultivation systems is labor-intensive, inconsistent, and unsuitable for scaling. This thesis, undertaken in collaboration with *Growy*—a Dutch vertical farming company specializing in robotics and automation, addresses these challenges by developing an *AI-driven germination monitoring system*. The study reformulates traditional seed-level object detection as a *patch-level image classification approach*, which better accommodates the high-density and overlapping seedlings characteristic of vertical farming environments.

We gathered a custom dataset, consisting of around *8,000 image patches*, using a robotic imaging system integrated into Growy's hydroponic setup. Two advanced deep learning architectures—*Vision Transformer (ViT)* and *ConvNeXt*—were trained and evaluated using both *cross-entropy* and *focal loss* functions to predict germination categories ranging from 0% to 100%. The findings indicated that *ConvNeXt with cross-entropy loss* achieved the most consistent and accurate results, with validation accuracies of approximately *95%*, while ViT demonstrated greater sensitivity to data imbalance and training conditions. Moreover, the comparative analysis of focal loss and cross-entropy loss in the context of imbalanced data revealed that cross-entropy produced smoother learning curves than focal loss.

The proposed pipeline offers a *scalable and efficient solution* for automated germination assessment, reducing manual labor and enhancing monitoring precision in vertical farming systems. Beyond germination, the framework is adaptable to other growth stages and crop species, for a fully integrated and *AI-based plant phenotyping* in controlled-environment agriculture.

Keywords: Germination rate, vertical farming, image classification, Transformer, CNN

1 Introduction

Vertical farming is an agricultural paradigm in which crops are cultivated in vertically stacked layers—often within buildings—rather than across large horizontal expanses (Despommier 2010; Beacham, Vickers, and Monaghan 2019; Benke and Tomkins 2017). This method enables the efficient use of limited land resources, particularly in urban settings, by transforming vertical space into productive growing areas (Padhiary et al. 2025; Fuentes-Peñailillo et al. 2023). Vertical farming typically employs soilless cultivation methods, such as hydroponics or aeroponics, combined with tightly controlled environments that regulate light, temperature, humidity, and nutrients (Despommier 2010; L. He et al. 2025). By separating plant development from external climate variability, vertical farming offers the potential to stabilize crop yields, reduce both land and water usage, and lower risks associated with pests and unpredictable weather events (Beacham, Vickers, and Monaghan 2019; Panotra et al. 2024).

Despite its promise, vertical farming also presents significant obstacles. Research has underscored the substantial energy requirements for artificial lighting and environmental regulation, which can constrain both scalability and long-term sustainability (Al-Chalabi 2015; Kozai 2018). On a practical level, crop management activities such as monitoring germination, tracking growth stages, and estimating yields remain labor-intensive and time-consuming. Manual inspection proves challenging, given the limited access between tightly spaced cultivation layers, hindering efficient data collection (L. He et al. 2025; Panotra et al. 2024). These issues create a bottleneck for optimizing crop productivity in vertical farms, especially during germination, a stage that directly impacts farm efficiency and yield.

Recent development in automation and computer vision present promising solutions to these challenges. Automated imaging systems, combined with machine learning, enable precise and scalable monitoring of plant development, thereby reducing reliance on manual observation (Andreas Kamilaris 2018; Liakos et al. 2018). High-resolution images processed with robust algorithms can track germination dynamics in real time, providing accurate insights while saving time and resources. These technologies align with the broader trend of integrating AI-driven precision agriculture into vertical farming systems (Padhiary et al. 2025). In particular, automated robotics and AI-driven imaging networks can continuously capture seed data, while computer vision models such as Convolutional Neural Networks (CNNs) objectively detect germination and growth stages (Ashok and Adesoba 2023; Nico Heider 2025; Dias et al. 2011). Through automation, vertical farms can expand germination monitoring across various crops, optimize resource allocation, and overcome the inherent limitations of manual inspection (Fuentes-Peñailillo et al. 2023).

However, determining the optimal germination period for different plant species remains an unresolved challenge. While prior research has explored automated plant phenotyping, relatively few studies focus on early-stage germination analysis in vertical farming environments, where space constraints and experimental scale complicate manual validation. Addressing this gap is critical for streamlining workflows, minimizing resource wastage, and maximizing production efficiency in controlled-environment agriculture.

This thesis was developed in partnership with *Growy*, a vertical farming company based in Amsterdam that focuses on sustainable, technology-based agriculture through robotics and automation. Growy's expertise includes multi-layer hydroponic cultivation, advanced climate management, and AI-integrated monitoring systems, placing the company at the forefront of innovation in European vertical farming. Building on Growy's biological research, this study aims to optimize plant profiles by determining the shortest germination periods for selected

species. Specifically, the thesis introduces and validates an automated image classification system designed to predict germination rates at various developmental stages. By harnessing computer vision and machine learning, the proposed approach addresses the current limitations of manual monitoring, improves operational efficiency, and enhances the scalability of germination assessment in vertical farming.
The thesis is structured as follows: Section 2 reviews related literature on vertical farming, germination analysis, and computer vision in agriculture. Section 3 describes the research methodology, including experimental design, imaging setup, and classification algorithms. Section 4 presents the results of automated germination analysis. Section 5 discusses the implications of these findings for vertical farming. Section 6 addresses the limitations of germination analysis in vertical farming. Section 7 outlines possible directions for future research. Section 8 concludes the study by summarizing the key findings.

2 Related work

Object detection is a fundamental task in computer vision that combines image classification with spatial localization, answering the question of "what objects are present and where." Unlike image classification, which assigns a single label to an entire image, object detection predicts both bounding boxes and class labels for multiple objects simultaneously. Modern approaches are typically divided into two categories: two-stage detectors, such as Faster R-CNN (Ren et al. 2015), which first generate region proposals before classification, and one-stage detectors, such as SSD (W. Liu et al. 2016) and YOLO (Redmon et al. 2015), which directly predict object categories and bounding boxes in a single forward pass. Despite these advances, significant challenges persist in detecting small, overlapping, or densely clustered objects, as bounding-box based representations can become ambiguous in such settings (H. Wang and Gao 2024). These limitations are especially relevant for seed germination monitoring, where seeds and their emerging shoots overlap, motivating exploration of alternative strategies beyond conventional object detectors.
Image classification is another foundational task in computer vision, in which the goal is to assign a category label to an entire image. Earlier approaches were based on shallow models, but the field has been improved by deep learning, particularly through convolutional neural networks (CNNs) such as AlexNet (Krizhevsky, Sutskever, and Hinton 2012), VGG (Simonyan and Zisserman 2015), and ResNet (K. He et al. 2015), which achieved high accuracy by learning hierarchical feature representations directly from data.
On large-scale datasets such as ImageNet, image classification models have demonstrated impressive performance. However, challenges remain when these models are applied to fine-grained tasks or domains where class differences are subtle. In such cases, distinguishing between visually similar categories can be difficult. In agricultural contexts, image classification has been particularly effective, especially when morphological differences between seed types are minimal or visual cues are obscured. For tasks such as plant disease recognition (Mohanty, Hughes, and Salathé 2016), growth stage analysis (Qin et al. 2023), and quality assessment (Song et al. 2023), image classification models often outperform object detection methods, particularly when precise localization is not essential.

2.1 Deep Learning for Automated Germination Detection

Deep learning has increasingly been used to detect seed germination. For instance, Zhao et al. (2023) introduced YOLO-r, which combines image partitioning, a Transformer encoder, layers tailored for small objects. Their model reached a mean average precision (mAP) of 95.39% for rice under complex conditions. Yao et al. (2024) proposed SGR-YOLO, an enhanced YOLOv7 variant with Efficient Channel Attention (ECA), a BiFPN neck, and GIoU loss, reporting accuracies of 94% in hydroponic boxes and 98.2% in Petri dishes for wild rice. More recently, Sun et al. (2025) presented CSGD-YOLO for corn, built on a lightweight YOLOv8n with Ghost_Detection and RFAConv modules, achieving an mAP of 92.99% with a very compact model.

However, in dense cultivation settings such as vertical farms, bounding-box detectors often fail. As shoots and roots from neighboring seeds overlap, their shapes intertwine, making it difficult for object detectors to separate individuals reliably. Thus, while object detection works well in controlled or sparse layouts, scaling to realistic, high-density agriculture will require alternative strategies.

2.2 Reformulations Beyond Object Detection

When fine-grained localization of individual seeds is impractical, reformulating germination monitoring as a patch-level classification problem offers a scalable solution. Rather than tracking each seed, local image patches are classified according to germination density, capturing aggregate biological indicators such as color changes, stem elongation, or leaf emergence. This approach mirrors the methodology used by Growy biologists, who estimate germination rates by assessing representative regions rather than counting every seed.

This reframing enhances robustness, reduces annotation complexity, and preserves biological interpretability. Importantly, it enables models to scale to hundreds of seeds per image, which is essential in vertical farming systems.

2.3 Advances in Classification Architectures

The rise of Vision Transformers (ViT) has reshaped computer vision by modeling global relationships across image patches rather than relying only on local convolutions (Dosovitskiy, Beyer, Kolesnikov, et al. 2020). ViT treats an image as a sequence of patches and uses a Transformer encoder to capture long-range dependencies, achieving strong results on large-scale classification and inspiring a wave of transformer-based vision models.

By contrast, ConvNeXt is a modernized CNN with a design inspired by transformers — such as inverted bottlenecks and layer normalization — while preserving CNN efficiency (Z. Liu et al. 2022). It follows a four-stage, ResNet-like hierarchy that progressively reduces spatial resolution and increases channels, enabling both fine-grained and high-level feature learning. Unlike ResNet, ConvNeXt replaces bottleneck residual blocks with streamlined ConvNeXt blocks, simplifying the architecture while improving representational power.

Together, these trends highlight that models must be not only accurate but also efficient and adaptable to domain-specific constraints, such as high-density plant layouts. Agricultural imaging further benefits from approaches that remain robust with limited datasets and constrained computational resources.

2.4 Handling Class Imbalance and Rare Events

Image datasets may have imbalanced classes, especially in agriculture, where early and late growth stages are captured less often than intermediate ones. Standard cross-entropy can bias training toward majority classes and lowering the robustness. To address this, Lin et al. (2017) proposed focal loss, which down-weights easy examples and emphasizes harder cases.
For germination monitoring, where classes may be underrepresented, focal loss can rebalance learning. This thesis evaluates both cross-entropy and focal loss to assess their effects on classification robustness.

This thesis compares ViT and ConvNeXt on patch-level germination monitoring in vertical farming. Unlike prior work limited to specific crops or sparse imagery, it addresses high-density real-world conditions, integrates loss function design for imbalanced data. By doing so, it bridges the gap between experimental detection research and the practical deployment of automated germination monitoring systems.

3 Methodology

3.1 Problem Definition

In traditional agricultural research, seed germination detection is often treated as a standard object detection task, where individual seeds are annotated and analyzed as separate instances. This approach is successful for well-separated seeds which allow clear visual boundaries and straightforward classification. However, the Growy vertical farming system presents a different scenario. Seeds are densely sown within gutters which leads to overlapping seedlings and rapid morphological changes during growth. These factors complicate the individual seed detection process and requires an adapted methodological framework tailored to Growy's high-density cultivation environment. The following sections outline the reasoning behind reformulating the germination detection task from an object detection problem to an image classification approach. This transition was made from practical challenges encountered in applying object detection methods to the dataset during annotating.

3.1.1 Initial Approach - Object Detection

Seed germination detection has traditionally been conceptualized as an object detection problem, wherein each seed is regarded as an independent object to be classified as either germinated or non-germinated. This method performs effectively when seeds are spatially separated, imaged under controlled conditions, and retain distinct morphological features throughout the germination process. For example, figure 1 illustrates a sample from Yao et al. (2024), in which seeds were successfully annotated with bounding boxes, unimpeded by overlapping structures or the risk of excluding neighboring plants during labeling.
In the present case study, however, these assumptions do not apply due to the unique characteristics of the dataset. Figure 2a illustrates our early attempt to annotate the dataset for an object detection task. As shown, even at densities lower than typical production levels, constructing bounding boxes is challenging. As plants grow, the complexity increases markedly, ultimately making precise annotation infeasible, as demonstrated in figure 2b.

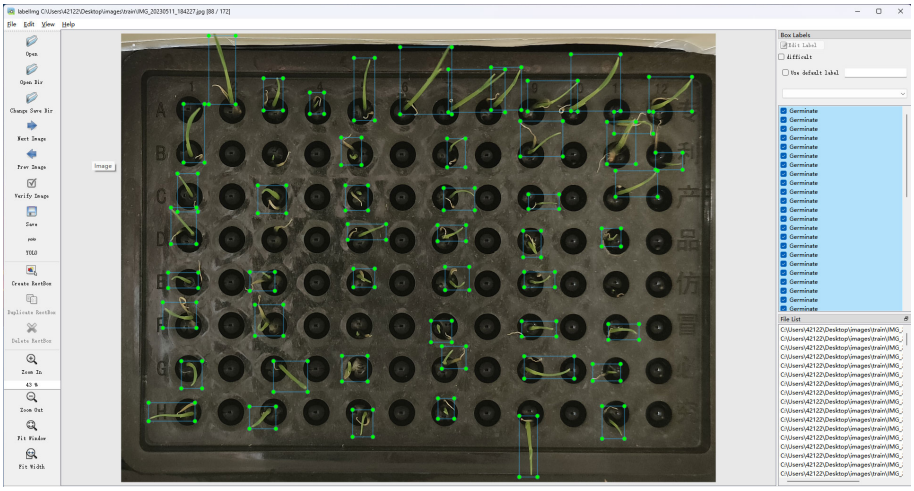
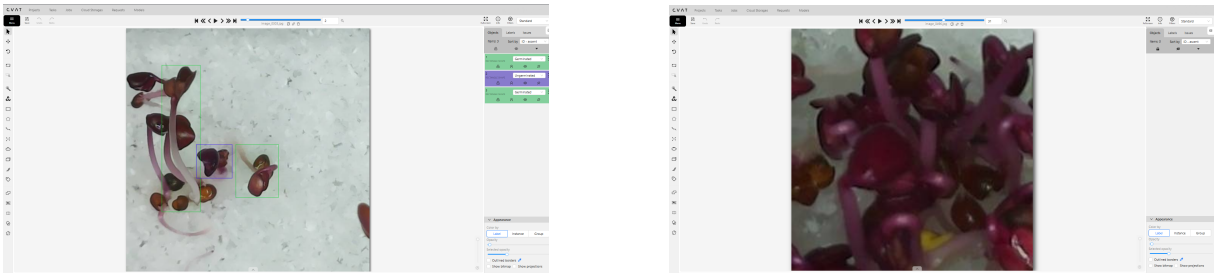


Figure 1: Annotating wild rice seed germination detection (Yao et al. 2024)



(a) An annotated sparse seeded gutter

(b) A tile of denser seeded gutter

Figure 2: An example of annotating a tile of Red Daikon seed

Figure 3 further clarifies this limitation. Each image contains hundreds of seeds captured at very high resolution, permitting detailed inspection but resulting in densely packed arrangements. Unlike datasets with only a few seeds per frame, the high instance count significantly increases annotation complexity and computational demands. Moreover, bounding boxes produced by object detection models cannot reliably isolate individual seedlings, as overlapping leaves and roots obscure distinct boundaries. Seeds that are initially separable in early stages become indistinguishable as development progresses. Accurately estimating germination rates would require counting seeds in each gutter, identifying germinated plants, and calculating average proportions—an approach rendered impractical by these challenges. Consequently, the object detection framework is unsuitable for this context and does not provide reliable results.

3.1.2 Reformulated Approach - Image Classification

Object detection proved unsuitable for this application due to frequent occlusions, overlapping seedlings, and high seed density within gutters. Consequently, the task was reformulated as *patch-level image classification* rather than individual seed detection. Each high-resolution image was partitioned into square patches of 512×512 pixels, with each patch representing a small region of the gutter containing a few seeds. This design preserves local spatial context relevant to germination status while avoiding the complexity of precise instance localization and tracking.

This reformulation also aligns with the established workflow of Growy biologists, who estimate

Figure 3: An example of a Red-Daikon-cultivated-gutter picture

germination by visually assessing representative regions of each gutter rather than counting every seed. In accordance with this practice, each patch was assigned a germination category (i.e., a discrete stage label), enabling the model to directly predict patch-level germination.
This approach provides several advantages: first, it increases robustness to overlapping seedlings, as the classifier evaluates aggregate germination within a region rather than depending on precise separation of roots and shoots. Second, it enhances scalability, as large images containing hundreds of seeds are decomposed into manageable units for efficient, parallel processing. Third, it simplifies annotation, since assigning a categorical label per patch is much faster and less error-prone than drawing bounding boxes or masks for each seed, thereby reducing annotation time and variability.
In principle, patch-level germination could be formulated as a regression problem. In practice, however, the dataset was collected at discrete time points, and true continuous labels were not available. The inherent intervals between imaging schedules mean that per-patch targets are naturally discrete and ordinal rather than precise continuous measurements. Treating these labels as continuous would introduce label noise and temporal bias, such as assigning intermediate numeric values that do not correspond to actual biological states. Therefore, a classification approach better aligns with the available supervision and minimizes target mismatch.
By shifting the focus from individual object detection to regional classification, the model can exploit biologically meaningful patterns—such as local density, color variation, and early leaf emergence—to deliver reliable monitoring of germination progress across the gutter while avoiding the limitations inherent in instance-level detectors.

3.2 Dataset Collection

In the Growy vertical farming system, robots in each layer are responsible to reduce manual labor. Diagram 4 shows a top view of a layer and its robot operating in the farm.
The cultivated gutters are placed in available slots and the robot moves along the y-axis between them. To reach different parts of the gutter, a head is installed on each robot which is able to move along the x-axis. Figure 5 shows a side view of the robot system. The red rectangle that moves along the gutters, is the working head of the robot which is responsible for watering and taking pictures.

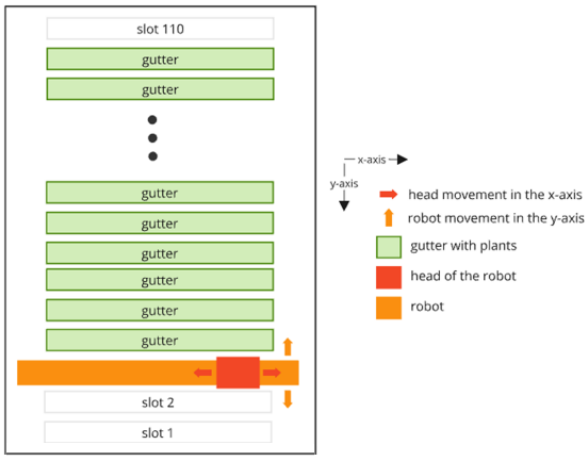


Figure 4: Top view of a layer and its robot

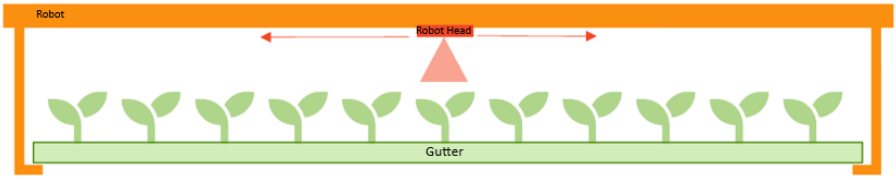


Figure 5: Side view of robot movement along a gutter

As part of this project, a camera module was introduced to the robot head to capture image for germination monitoring automatically. Figure 6 shows an overview of the camera board installed on a robot head. The imaging hardware was built on the Raspberry Pi Zero 2 W platform, a compact quad-core single-board computer featuring a 64-bit Arm Cortex-A53 CPU at 1 GHz, 512 MB of LPDDR2 SDRAM, wireless LAN, Bluetooth 4.2 with BLE support, and a CSI-2 camera connector for high-resolution imaging.

Figure 6: Robot head and camera board

Image acquisition was organized through predefined instructions referred to as photoruns. The photorun protocol schedules picture-taking for all gutters on a given layer. During routine

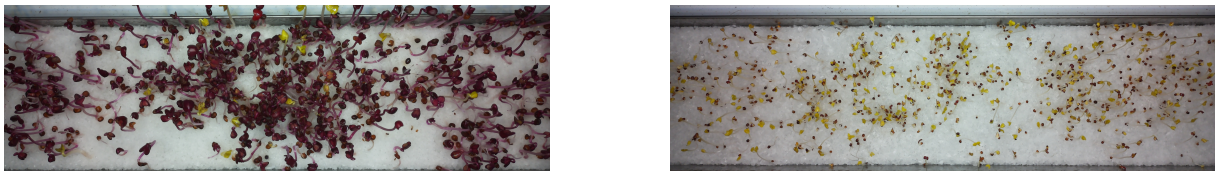
operation, the robot adheres to this schedule for consistent image coverage. However, when the robot is assigned higher-priority tasks, such as irrigation, image capture is skipped. Consequently, some photoruns may lack images for specific gutters or, at times, for entire layers. This constraint introduces variability into the dataset, as the timing of captured images depends on the robot's operational workload.

3.2.1 Class Overlap

During constructing a dataset for germination analysis, it is important to recognize the substantial diversity present within the images. Within a single frame, different patches may correspond to varying germination rates, and the appearances of seeds and seedlings can be repetitive across patches or captured from similar perspectives. Additionally, as germination is a continuous process, the visual features of one germination class often overlap with those of adjacent classes, creating challenges in establishing definitive class boundaries during annotation.

3.2.2 Crops and Growth Phase

The dataset was collected from a vertical farming system using two products: "Red Daikon" and "Bright & Spicy mixed". The Red Daikon gutters contain only Red Daikon seeds, whereas the Bright & Spicy mix comprises a blend of Mustard, Paksoi, Tatsoi, and Mizuna. Cultivated gutters progressed through four stages—germination, pre-growth, growth, and pre-harvest—but images were acquired exclusively during the germination stage. Each image was captured at a resolution of 4608×2592 pixels, providing high levels of visual detail. Figure 7 shows sample cultivated gutters of "Red Daikon" and "Bright & Spicy mix" on the final day of germination.



(a) Red Daikon (b) Bright & Spicy mix

Figure 7: Samples of the last day of germination of 2 different cultivars

3.2.3 Planting Method and Seed Distribution

Seeds are sown in gutters filled with Growfoam, which served as a soil substitute and plant substrate. Seeds are manually sprinkled across the foam surface, but because the material contains cavities, some seeds fall into gaps, producing uneven spatial distributions. Certain regions become densely populated, while others remained sparse. To capture this variability, gutters are planted at several density levels (10%, 30%, 50%, 70%, and 100% of production capacity), though most emphasis is placed on the 100% condition. Each Red Daikon gutter contains approximately 40g of seeds, while each Bright & Spicy gutter contained 5.5g when cultivated under production conditions. This variability ensured that the dataset reflected a range of densities likely to be encountered in later input patches.

After observing promising results with the trained models, additional crops—"Korean Mint" and "Paksoi"—were introduced into the dataset. However, these were later excluded. In the case of Paksoi, the dataset did not contain a sufficient number of high-quality tiles in few of the germination categories, which prevented reliable class representation. For Korean Mint, the extremely small size of seeds and seedlings made it difficult to distinguish germination stages with confidence. As a result, both crops were removed from the final dataset to maintain data quality and class balance.

3.2.4 Imaging Setup and Calibration

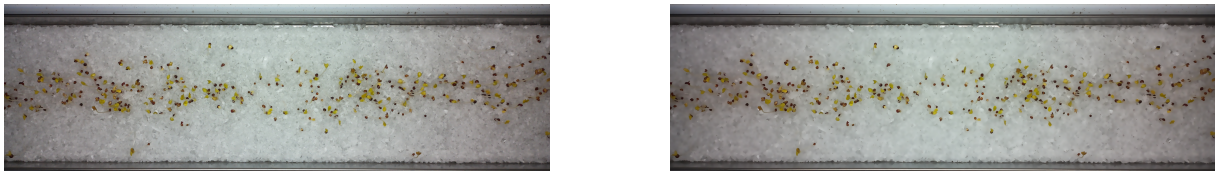
Initial image calibration was necessary because the combination of a white background and moist Growfoam produced strong reflections, which made many seeds appear white and indistinguishable from the background. Early tests revealed that at 50% brightness, reflections caused many seeds to appear white, blending into the background. The issue was resolved by lowering the brightness to 30%, fixing the blue light setting at 15%, and ensuring that seed and seedling colors were not distorted. Despite having a diffusing sheet on the camera board, images initially had sharper focus in the center compared to the edges. To address this, the number of photographs per gutter was increased from 5 to 8 per run, ensuring full coverage. Each gutter was photographed six times a day, with a 4-hour gap.

3.2.5 Image Preprocessing

The increased number of photographs introduced overlaps between adjacent images. Manual calculations revealed that approximately 460 pixels along the x-axis of the left side overlapped. These overlaps were cropped during preprocessing, which added extra annotation time. Initially, standard Gaussian filters and deblurring techniques (such as Wiener deconvolution (Wiener 1949)) were applied to the green channel to enhance image clarity. However, these methods proved ineffective, particularly for this case, as the background is quite noisy and the foreground contains tiny seeds.

To address these challenges, we adopted the Non-Local Means Denoising algorithm, as described by Coll and Morel (2011). This method replaces the color of a pixel with an average of the colors of similar pixels, even if they are far apart, by scanning a large portion of the image to find these pixels. This approach effectively smooths the background while preserving fine details in the foreground, such as the delicate structures of small seedlings. To optimize the Non-Local Means (NLM) Denoising parameters effectively, σ value had to be adjusted based on the image characteristics. Theoretically, a lower sigma (5-15) preserves sharp details while still smoothing out the noise. However, to effectively remove noise, a higher sigma (15–30) should be considered. In this case, higher sigma effects the small foreground objects and it needed to apply a value from the lower range. Practically, $\sigma = 8$ showed results that suited all of the crops in the dataset. Figure 8b shows an improved picture of "Bright & Spicy mix" in "class 70%" by applying NLM method with the mentioned setting which is smoother than the original picture (figure 8a). This enhancement was done by applying ImageJ's Plugin - Non-Local Means Denoising. ImageJ (Schroeder et al. 2021)is an open-source image processing software for scientific image analysis.

Applying Non-Local Means Denoising significantly improved image quality, allowing for more accurate annotation and analysis of germination stages, especially for crops with tiny or overlapping seedlings.



(a) Raw Picture (b) Improved by NLM Denoising

Figure 8: An example of Bright&Spicy of "class 70%"

3.2.6 Gutter Detection and Image Tiling

Accurate gutter detection was required to ensure that each frame contained the full cultivation area. Common edge detection techniques, such as Canny or Sobel, were tested, but they failed to provide useful results for this task. These methods tend to detect all edges within an image, making it difficult to isolate the horizontal, parallel lines that correspond to the gutter edges, which are consistent along the x-axis. Template matching in OpenCV (2025)was initially performed using a "full-gutter" for the template. This approach proved unreliable, with a high error rate and inconsistent detections. The method was refined by using only the upper edge of the gutter (figure 9) as the template, which provided greater stability and accuracy. By detecting the upper edge, the location of the lower edge is determined by a fixed vertical distance (1400 pixels). If the lower edge cannot be detected within this expected range, it is assumed that the gutter is not fully present in the input image. This refinement was critical because the robotic imaging system occasionally captured frames outside the intended region.

Figure 9: The gutter edge used as the reference for "Template Matching" in OpenCV

Following gutter validation, each image was divided into 512 $\times$ 512 patches along both axes by taking 512 steps 8 times along x-axis and 2 time along y-axis. Tiling began slightly inside the gutter boundaries (200 pixels from both edges toward the center) to exclude irrelevant background content, as these parts do not contain usable and enough seeds frequently due to uneven Growfoam placement. To preserve the uniqueness of the dataset, overlap between adjacent tiles was not permitted, even though this occasionally resulted in partial seedlings being split across tiles. This decision became particularly relevant during the process of splitting the dataset for validation. Since the splitting was done randomly, there was a risk that the model could inadvertently "cheat" during validation by seeing parts of the same seedlings in both the training and validation sets. We made sure that possible overlap tiles were removed from the dataset as the way explained in section 3.2.5.

3.2.7 Germination Rate Categorization

Unlike conventional germination studies, which typically rely on standardized crop-specific vigor benchmarks, this vertical farming dataset required customized germination categories. After consultation with company biologists, germination rate was determined by evaluating images of each gutter in different germination stages and assigning scores based on visual cues such as seed color changes, stem elongation, leaf size, and leaf unfolding. Each image patch was then assigned to one of six germination categories: 0%, 10%, 20%, 50%, 70%, and 100%.

Because growth is non-uniform within and across gutters, patches from the same image often represented different growth stages, necessitating manual review and cleaning of the dataset. Moreover, some patches contained no plants due to uneven seed distribution. On average, approximately 80% of the cropped tiles were retained as usable samples.
The final dataset contained a total of 7,979 image patches including Red Daikon and Bright&Spicy Mix, distributed across the six germination categories as follows: 1,652 in class 0%, 1,263 in class 10%, 1,349 in class 20%, 1,101 in class 50%, 1,172 in class 70%, and 1,442 in class 100%.

Class	**Daikon patches**	**B&S patches**	**Total patches**	**Percentage of dataset**
0%	794	858	1,652	20.7%
10%	820	443	1,263	16.0%
20%	674	675	1,349	17.0%
50%	524	577	1,101	13.7%
70%	578	594	1,172	14.6%
100%	841	601	1,442	18.0%
Total	**4,231**	**3,748**	**7,979**	**100%**

Table 1: Distribution of image patches across germination categories

3.3 Pipeline

The proposed pipeline operates by first receiving a sequence of input images captured from a gutter. Within each image, the gutter region is identified, and the program extracts square tiles of size 512×512 pixels. Each tile is then processed by an image classification model, which predicts its corresponding germination rate. This procedure is repeated iteratively until all patches of the image have been analyzed.
Once predictions for all tiles are obtained, their values are averaged to estimate the germination rate of the entire gutter. Finally, the program aggregates the average germination rates across all images of a given gutter, producing the overall germination estimate as the final output. Figure 10 visualizes how the pipeline works.

3.4 Models and Architecture

Selecting an appropriate architecture for germination classification is critical to achieving both accuracy and generalizability across diverse cultivation conditions. Among the wide range of deep learning models available, this study focuses on two state-of-the-art architectures: the Vision Transformer (ViT) and ConvNeXt. Both models have demonstrated competitive performance across numerous image recognition benchmarks, combining representational strength with architectural flexibility that makes them suitable for agricultural imaging tasks characterized by high visual variability and limited annotated data.
The Vision Transformer (ViT), introduced by Dosovitskiy, Beyer, Kolesnikov, et al. (2020), represents a improvement in computer vision by treating images as sequences of patches and applying a Transformer encoder to capture global contextual relationships. Its patch-based formulation aligns naturally with the image tiling strategy adopted in this work. In agricultural and plant phenotyping domains, Transformer-based models (including ViT variants) have been

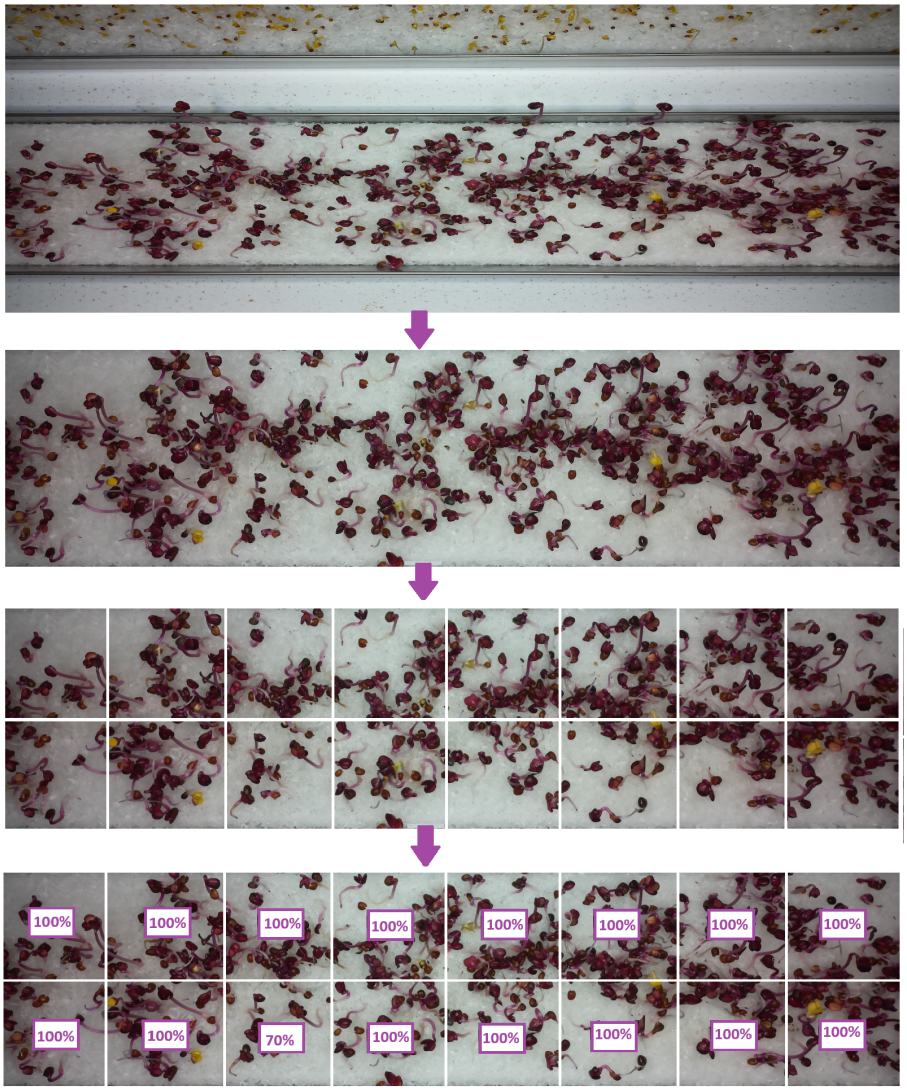


Figure 10: The proposed pipeline of predicting the germination rate of a gutter

increasingly explored. For example, Han et al. (2025) reviewed applications of deep learning, including Transformer architectures, for crop monitoring and phenotyping. These models demonstrate strong potential for automating visual analysis in controlled-environment agriculture.

The ConvNeXt architecture, proposed by Z. Liu et al. (2022), re-examines convolutional network design with modern enhancements inspired by Transformers, such as inverted bottlenecks, large kernel convolutions, and layer normalization. This design achieves Transformer-level performance while preserving the inductive biases and computational efficiency of traditional CNNs. Empirical evidence supports the applicability of ConvNeXt in agricultural imaging. X. Wang et al. (2023) introduced an ECA-ConvNeXt model for rice leaf disease identification, combining ConvNeXt with Efficient Channel Attention to achieve high accuracy under field conditions. Similarly, Irmawati et al. (2023) demonstrated the competitive performance of ConvNeXt in early detection of potato leaf diseases, highlighting its capacity to capture fine-grained texture patterns in complex backgrounds. Together, these studies confirm that ConvNeXt is effective for plant-imaging scenarios requiring high precision and adaptability.

Both ViT and ConvNeXt present complementary advantages: ViT excels at modeling global

context and spatial relationships, while ConvNeXt effectively captures local textures and fine visual details. Evaluating both architectures within the same experimental framework enables a balanced assessment of their suitability for patch-level germination classification. Furthermore, both architectures support transfer learning from large-scale datasets such as ImageNet, accelerating convergence when fine-tuning on smaller agricultural datasets. Their scalability and proven generalization capabilities make them well aligned with the objectives of this study.

3.4.1 Training and Validation Sets

To preserve class proportions and ensure representative coverage, stratified sampling was applied. Eighty percent of the data were allocated for training, while the remaining twenty percent formed the validation split, used a fixed seed of 42 to ensure reproducibility. All images are RGB.

3.4.2 Vision Transformer for Image Classification (ViT)

Model Configuration

The experiments employed the **Vision Transformer Base (ViT-Base/16, 224)** pretrained on ImageNet-21k (Dosovitskiy, Beyer, Kolesnikov, et al. 2020), implemented via the Hugging Face Transformers library (`google/vit-base-patch16-224-in21k`). The classification head is replaced with a custom linear layer matching the number of germination classes. The model was fine-tuned end-to-end using GPU acceleration when available.

Data and Preprocessing

In order to train the model, we used`ViTImageProcessor` to adapt the dataset to the model's expected input format. Specifically, it enables the transformation of the original 512×512 pixel patches into the ViT input resolution of 224×224 pixels without cropping or losing any part of the image content. Instead of discarding spatial information through center cropping, the processor performs isotropic resizing while preserving the complete visual field of each patch. This ensures that germination-relevant features, including fine root structures and emerging shoots near the image borders, were retained in every sample.

Following resizing, channel-wise normalization was applied using the same statistical parameters (mean and standard deviation) as those used during the pretraining of the ViT on ImageNet-21k (Dosovitskiy, Beyer, Kolesnikov, et al. 2020). This normalization step is essential because pretrained Vision Transformers assume inputs normalized to the same distribution as their training data; deviations from this distribution can lead to activation mismatches in early network layers, thereby degrading convergence and accuracy. By standardizing input intensities across the RGB channels, we ensured consistency with the pretrained model's learned feature representations and improved fine-tuning stability.

Moreover, each image was tensorized into a pixel_values tensor which converts preprocessed images into the exact numerical structure expected by the Hugging Face `Trainer` and PyTorch backend. This step ensures that pixel intensity values are stored in a consistent range and data type and facilitates efficient batching, reproducible gradient computation, and seamless integration with mixed-precision training. Integer class labels were preserved separately to maintain clear correspondence between samples and targets throughout the data pipeline.

Handling Class Imbalance

Let n_c denote the number of training samples in class c. To find the effects of class imbalance, the weights are computed as:

$$\tilde{\alpha}_c = \frac{1}{n_c},$$

and normalized to have mean one for numerical stability:

$$\alpha_c = \tilde{\alpha}_c \cdot \frac{K}{\sum_{j=1}^{K} \tilde{\alpha}_j},$$

where K is the number of classes. The resulting normalized weights α_c are passed to the loss function to balance gradients during training.

Normalization of the class weights ensures that the overall scale of the loss remains consistent with the unweighted baseline, thereby stabilizing the optimization process. It alone can produce large coefficient disparities when class frequencies differ substantially, causing gradients from minority classes to dominate and destabilize convergence. By constraining the weights to have a mean of one, the relative importance between classes is preserved, but the overall gradient magnitude remains within a stable range. This prevents abrupt fluctuations in the effective learning rate.

Moreover, mean normalization allows the weighted loss to remain numerically comparable to the standard unweighted loss. This comparability simplifies hyperparameter tuning and ensures that training metrics, such as loss and accuracy, retain consistent interpretability across models and datasets. The weighting scheme balances fairness toward minority classes with the need for optimization stability.

Data Augmentation

To improve model generalization and robustness, initially CutMix augmentation (Yun et al. 2019) was applied to adjacent classes during training. CutMix combines two images (and their labels) by cutting a rectangular patch from one image and pasting it onto another. For a mini-batch (x, y), a mixing ratio λ was sampled, and a random rectangular region was cut and pasted between two images, forming:

$$x' = \mathrm{CutMix}(x_i, x_j), \quad y' = \lambda' y_i + (1 - \lambda') y_j,$$

where λ' is adjusted based on the ratio of the mixed area to the total image area. CutMix is applied exclusively only during training; no augmentation was used during evaluation.

In our specific classification case, this strategy was particularly appealing because individual image patches may contain seedlings at multiple germination stages. Consequently, CutMix implicitly regularizes the model against over-reliance on localized features by exposing it to mixed spatial contexts. However, it also presents a conceptual limitation: although the class labels are represented as numerical indices, they denote *discrete and ordinal* germination stages rather than continuous values. Thus, the label interpolation produced by CutMix (e.g., 40% class 1 and 60% class 2) does not correspond to any physically meaningful intermediate state. In the CutMix examples shown in figure 11, each image results from blending two patch samples with a mixing ratio that determines how much of the base image remains visible after inserting a patch from another class. When λ is high (e.g., 0.9–0.94 in the top-left images), only a small patch is added, so the mixed sample still looks visually coherent and retains the

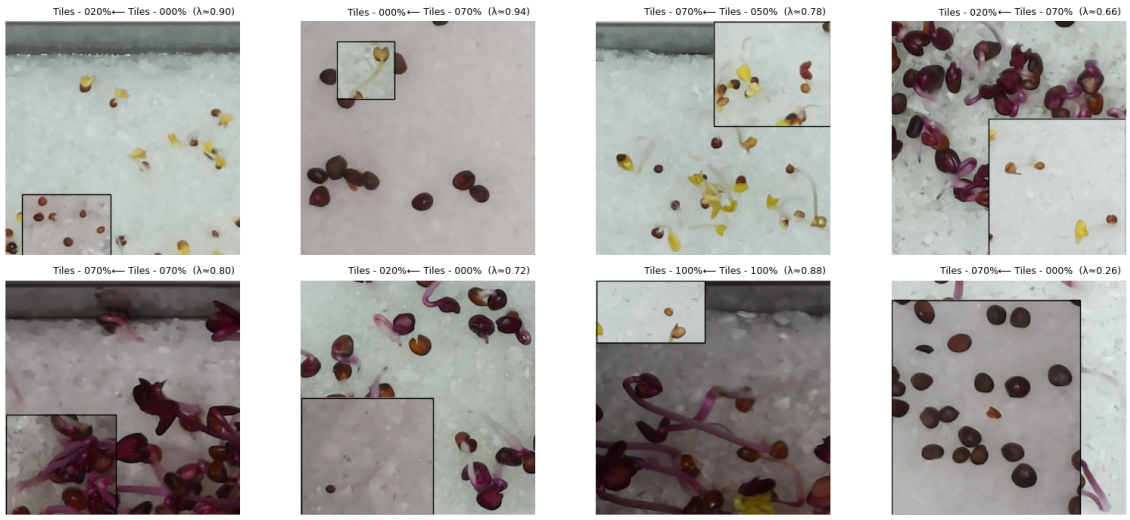


Figure 11: Examples of CutMix application on the mixed dataset

appearance of the base class. When λ is low (e.g., 0.26 in the bottom-right image), most of the image is replaced by the donor patch, producing an output that still appears realistic but primarily reflects the donor class. However, when λ is intermediate (0.5–0.7, such as in the middle columns), both classes occupy similar proportions, leading to images that seem visually inconsistent or confusing to humans. Moreover, patches from different crops in mixed dataset are blended, which neither in validation set (unseen data) nor in reality it can happen.
For this reason, while CutMix proved useful for improving feature generalization and robustness, the resulting label mixing must be interpreted as a regularization technique rather than a biologically valid representation of intermediate germination levels. As a solution, several label-preserving augmentations were applied during training, including random horizontal flips, small random rotations, and brightness jitter. These operations increase the model's invariance to minor geometric and photometric variations commonly observed in imaging setups without altering the labels.

Optimization and Hyper-parameter Tuning

Training was conducted using the Hugging Face `Trainer` framework with the following configuration: AdamW optimizer, an initial learning rate of 5×10^{-5}, batch size of four images per device, and 30 epochs. The learning rate is adjusted dynamically using the `ReduceLROnPlateau` scheduler, which reduced the learning rate when the validation accuracy plateaued. Evaluation is performed once per epoch, and the best model was selected based on validation accuracy.
To identify the optimal configuration for fine-tuning the Vision Transformer (ViT) on the germination classification task, the tuning process was implemented using the Weights & Biases (W&B) framework and employed a Bayesian optimization strategy to efficiently explore the hyperparameter space. Validation accuracy was defined as the target metric to maximize. The search space included key optimization, regularization, and scheduling parameters.
Learning rate: logarithmically sampled between 1×10^{-6} and 5×10^{-4} to capture both conservative and aggressive updates.
Batch size: [4, 8] — smaller batches were considered to stabilize gradients for limited dataset size and to fit within available GPU memory.
Label smoothing: [0.0, 0.05, 0.1] — evaluated to reduce overconfidence and improve gen-

eralization.
Weight decay: [0.0, 0.01, 0.05, 0.1] — tested as a regularization term to limit overfitting through L_2 penalization of model parameters.
Optimizer: `adamw` and `adam` — compared to assess the influence of decoupled weight decay versus standard Adam behavior.
Learning rate scheduler: `linear`, `cosine`, and `reduce_lr_on_plateau` — representing common strategies for transformer fine-tuning.
Warm-up ratio: [0.0, 0.05, 0.1, 0.2] — controlling the proportion of total steps used for linear learning rate ramp-up.
Number of epochs: [20, 30, 50] — to evaluate convergence behavior under different training durations.
Lambda: [0.0, 0.5, 0.85] - the fraction of the area that still belongs to destination image after CutMix.
In section 4.1.1, a number of sample tuning runs are explained to compare and analyze some settings.

3.4.3 ConvNeXt

Model Architecture and Transfer Learning

The model employed a pretrained **ConvNeXt-Tiny** backbone (Z. Liu et al. 2022) instantiated via the `timm` library, with a task-specific linear classification head replacing the original ImageNet classifier. Transfer learning follows a two-stage fine-tuning strategy. In the *warm-start* phase, the convolutional backbone is frozen for a few epochs, and only the classification head is optimized. After several epochs, all layers are unfrozen for full fine-tuning, allowing higher-level representations to adapt to the germination domain. This staged approach stabilizes early optimization and encourages smoother convergence when training on modestly sized, domain-specific datasets.

Data and Preprocessing

All images are resized and normalized to match the ImageNet statistics, ensuring compatibility with the ConvNeXt pretraining domain (Z. Liu et al. 2022). For training, light, label-preserving augmentations are applied, including random horizontal flips, small random rotations, and optional brightness jitter. These augmentations improve robustness to minor geometric and photometric variations typical in high-throughput agricultural imagery. Validation preprocessing employed deterministic resizing followed by the same normalization, ensuring that the validation distribution remained consistent and unbiased relative to the training domain.

Handling Class Imbalance

To address class imbalance and to better handle easy versus difficult samples, two complementary loss functions were evaluated.

Focal Loss. The Focal Loss (Lin et al. 2017) reweights each example based on its predicted confidence, focusing learning on hard, misclassified samples. For logits $\mathbf{z} \in \mathbf{R}^C$ and target y, with $p_t = \mathrm{softmax}(\mathbf{z})_y$, the loss is defined as:

$$\mathcal{L}_{\text{focal}} = -\alpha_y (1 - p_t)^\gamma \log p_t,$$

where $\gamma > 0$ controls the degree of down-weighting for easy examples, and α_y provides class-specific scaling.

Cross-Entropy with Label Smoothing. An alternative objective incorporated label smoothing to mitigate overconfidence and improve generalization:

$$\mathcal{L}_{\text{ls}} = -(1-\varepsilon)\log p_t - \frac{\varepsilon}{C-1}\sum_{j\neq y}\log p_j,$$

where ε is the smoothing parameter.

Class Weights. Per-class weights were computed from the training distribution using the balanced heuristic:

$$w_c = \frac{N}{C\,n_c},$$

where N denotes the total number of training samples, C the number of classes, and n_c the sample count in class c. These weights were passed as α_y for Focal Loss or as the `weight` argument for Cross-Entropy. The indices were aligned with the training dataset's class order to ensure consistency across all computations. This weighting scheme is conceptually related to the Class-Balanced Loss formulation (Cui et al. 2019), which adjusts per-class importance based on the effective number of samples.

Optimization and Hyper-parameter Tuning

A Bayesian hyperparameter sweep was conducted using W&B, targeting maximization of validation accuracy all in 40 epochs to capture the future behavior after possible convergence - early experiences showed ConvNeXt tended to converge always before epoch 30. The search space covered optimization (optimizer, learning rate, betas, ϵ), regularization (weight decay, dropout), training schedule (epochs, warm-up ratio, backbone freeze duration), loss settings (label smoothing, class weights), and light label-preserving augmentations (rotation and brightness).
Learning_rate: $[1 \times 10^{-5},\ 1 \times 10^{-4}]$
Batch_size: $[32,\ 64]$ - samples per update
Weight_decay: $[0.0,\ 1 \times 10^{-5}]$ - regularization strength
Label smoothing: $[0.05,\ 0.1,\ 0.2]$
Warmup_ratio: $[0.05,\ 0.1,\ 0.2]$
Dropout_rate: $[0.0,\ 0.2,\ 0.5]$ - dropout regularization
Freeze_epochs: $[0,\ 3,\ 5]$ - freezing training during the first epochs
Rotation: $[0,\ 10,\ 15]$ - image rotation augmentation
Brightness: $[0.0,\ 0.2]$ - brightness adjustment for augmentation
Betas: $\{[0.9,\ 0.999],\ [0.95,\ 0.999],\ [0.9,\ 0.98]\}$ - momentum terms

Optimization employed the AdamW optimizer with weight decay for regularization. The learning rate followed a cosine decay with warm-up schedule (Kalra and Barkeshli 2024), stepped per batch. Let T be the total number of training steps and T_w the number of warm-up steps.

The learning rate at step t is given by:

$$\eta_t = \begin{cases} \eta_{\max} \cdot \frac{t}{T_w}, & 0 \leq t < T_w, \\ \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min}) \left(1 + \cos \frac{\pi(t-T_w)}{T-T_w}\right), & T_w \leq t \leq T. \end{cases}$$

This schedule prevents early training instability by slowly increasing learning rate and gradually anneals it toward a low terminal value, for a smoother convergence and less overfitting..

4 Experiments and Results

This section presents the experimental design and evaluation procedures adopted to assess the performance of the proposed Vision Transformer (ViT) and ConvNeXt models for patch-level germination classification. Experiments were designed to systematically evaluate the effect of model architecture, dataset composition, and loss function on classification accuracy and generalization across different cultivation conditions.

Experimental Overview

An initial series of experiments was conducted using a combined dataset that aggregated all available image patches. A hyperparameter tuning module was executed for both ViT and ConvNeXt models on this mixed dataset to identify optimal configurations. The tuning procedure explored learning rate, batch size, weight decay, and optimizer parameters through automated sweeps integrated into the training framework, with validation accuracy used as the primary selection criterion.
Following the tuning stage, the dataset was partitioned into two subsets: the *Red Daikon* and *Bright&Spicy mix*. Each model was then trained and evaluated separately on these subsets, as well as on the mixed dataset, to examine domain generalization and the effect of environmental diversity on model performance.

Experimental Protocol

All experiments followed a consistent protocol across models and datasets. The same random seed was used to ensure reproducibility of data splits and weight initialization. Each configuration was trained for 30 epochs (which was picked from (20, 30, 50) epochs after fine-tuning for both models) using the corresponding optimizer and scheduler setup defined in Section 3.4. Validation accuracy was recorded after every epoch.
This unified experimental design enabled systematic benchmarking of both architectures under identical training conditions, isolating the contribution of model structure and loss formulation to overall performance.

4.1 ViT

4.1.1 Hyperparameter Tuning

As it was mentioned in section 3.4.2, a sweep module has been implemented with the suggested settings to find the best configuration for ViT with CE loss function. Among all evaluated

configurations, the following setting consistently yielded the best validation performance on the mixed dataset:

Hyperparameter	Optimal Value
Training epochs	50
Warm-up ratio	0.0
Scheduler type	Cosine
Optimizer	Adam
Weight decay	0.0
Label smoothing	0.1
Learning rate	5×10^{-5}
Lambda	0.0

The use of the cosine scheduler with no warm-up allowed the model to stabilize early while smoothly reducing the learning rate toward the end of training. The standard Adam optimizer, despite its lack of decoupled weight decay, achieved more stable updates for this dataset compared to AdamW, particularly when weight decay was set to zero. Label smoothing ($\varepsilon = 0.1$) contributed to improved calibration of class probabilities and reduced overfitting on underrepresented germination stages. Despite the improvements, ViT still struggled to learn.
Figure 12 illustrates the validation accuracy curves for six representative hyperparameter configurations during the ViT sweep on the mixed dataset. Each curve corresponds to a unique combination of optimizer, learning rate schedule, and regularization parameters sampled during the Bayesian search.

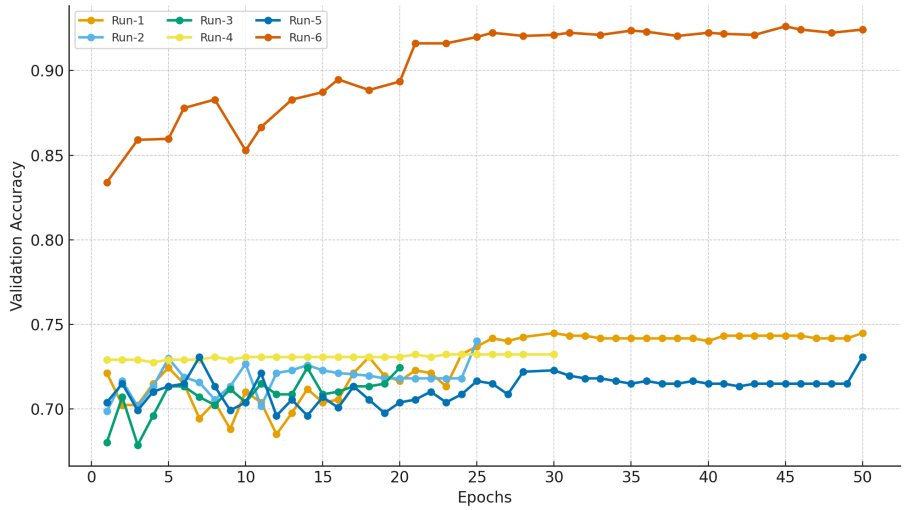


Figure 12: Validation accuracy across a few ViT sweep runs on the mixed dataset.

As shown in the figure, validation accuracy exhibits considerable fluctuation throughout training, with no promising convergence trend across most runs, however, longer trainings (50 epochs) show possible convergence. Despite the oscillatory behavior, the Bayesian optimization process identified a configuration that achieved the highest validation accuracy among all trials (see table in 4.1.1).
The observed fluctuations are typical in fine-tuning transformer architectures on limited datasets, where small gradient updates and stochastic regularization effects (label smoothing) can lead

to non-monotonic validation curves. The highest-accuracy configuration was selected as optimal setting, despite the lack of smooth curves in the first epochs.

CutMix Impact on Training

To investigate the effect of CutMix particularly, 3 sample runs has been separated in figure 13 to delve into details more precisely - 3 probabilities for CutMix 0.0, 0.5 and 0.85. As the figure illustrates, the model struggled to learn from the augmented dataset by CutMix and it did not lead to converge, while without its application, the model showed smoother learning curves which converged at the last epochs with higher accuracy. As a result, we decided to stop using CutMix for augmenting the data in the next steps.

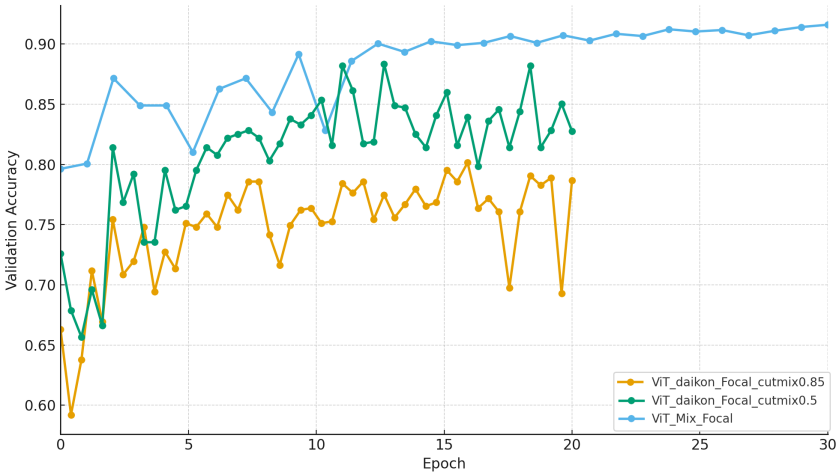
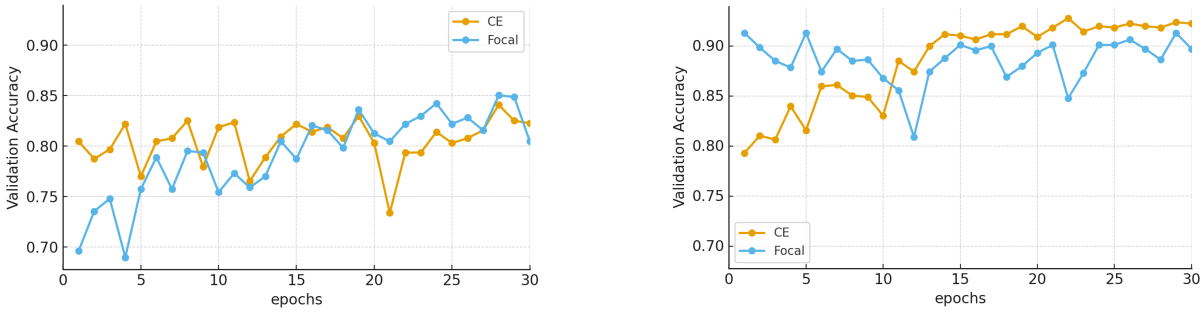


Figure 13: CutMix effect on training ViT

4.1.2 Training on Daikon set and Bright&Spicy set

To compare the role of Cross-Entropy (CE) and Focal Loss functions, we trained the model on two domain-specific subsets, *Red Daikon* and *Bright&Spicy Mix*. Figure 14 visualizes validation accuracy across 30 epochs for both losses on each dataset.



(a) Red Daikon

(b) Bright&Spicy Mix

Figure 14: Validation accuracy for ViT trained with CE and Focal loss functions

On *Red Daikon* dataset, CE attains a higher mean validation accuracy over the first 30 epochs (0.805) and a higher final accuracy at epoch 30 (0.823) relative to focal (mean 0.792; final 0.805). Focal, however, reaches a slightly higher peak accuracy (0.850 vs. 0.841), suggesting occasional gains on harder mini-batches but with similar volatility (standard deviation of epoch-to-epoch accuracy deltas σ_Δ of 0.0294–0.0297). In contrast, on *Bright&Spicy* dataset, CE exhibits both a higher mean accuracy (0.886 vs. 0.887 for Focal; practically comparable) and a higher final accuracy (0.923 vs. 0.897), with notably lower volatility ($\sigma_\Delta = 0.0173$ for CE vs. 0.0253 for Focal). Overall, CE is more stable on *Bright&Spicy*, whereas Focal provides intermittent peaks on *Red Daikon* without improving the average trajectory. However, both CE and Focal do not seem to converge on both sets, although on *Bright&Spicy* it was converging at the last epochs close to 90% accuracy.
Comparing datasets, *Bright&Spicy* yields consistently higher validation accuracy and smoother learning curves than *Red Daikon* for both losses. Over 30 epochs, the mean accuracy is ≈ 0.886 on *Bright&Spicy* versus 0.792–0.805 on *Red Daikon*, and the volatility is lower for CE on *Bright&Spicy* (0.0173) than for CE on *Red Daikon* (0.0297). These trends indicate that the *Bright&Spicy* domain is easier for the model to learn.
Across both datasets, CE provides the most reliable performance, especially on *Bright&Spicy* where it achieves the highest final accuracy with the lowest fluctuation. On *Daikon*, Focal occasionally attains higher instantaneous peaks but does not improve the overall stability or final accuracy within the 30-epoch window considered. The results suggest a preference for CE under the present class weighting and augmentation regime, with Focal as a potential alternative when emphasizing rare or systematically hard examples.

4.1.3 Training on the Mixed set

We repeated CE vs. focal comparison on the mixed dataset, following the same protocol as for the splits. Figure 15 shows the validation accuracy trajectories.

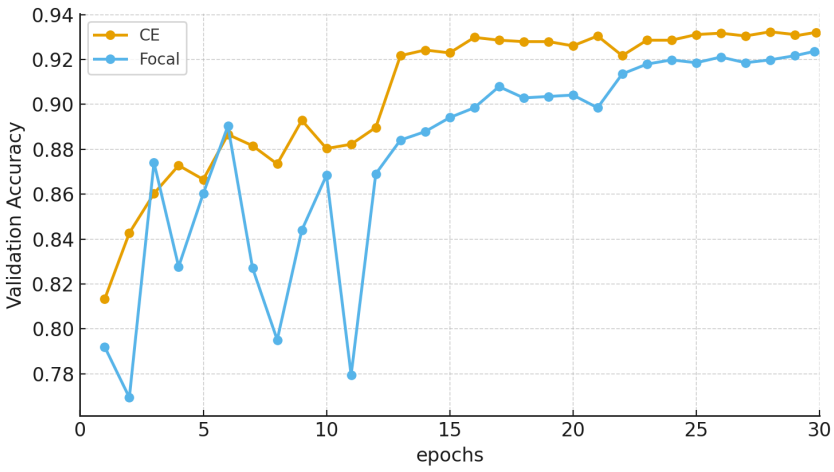


Figure 15: Validation accuracy for ViT trained with CE and Focal loss functions

Cross-Entropy attains higher mean validation accuracy and a higher plateau than Focal at later epochs, with substantially lower fluctuation. Over the common window, CE reaches a mean accuracy of 0.904 and a final value of 0.931, compared with Focal's mean of 0.877 and final of 0.922. The volatility of epoch-to-epoch accuracy changes (σ_Δ) is 0.0109 for CE versus

0.0383 for Focal, indicating that CE learns more stably on the heterogeneous mixed domain while also achieving a stronger final plateau.

Loss	**Mean Acc**	**Max Acc**	**Final Acc**	**Volatility (σ_Δ)**
Cross-Entropy	0.904	0.932	**0.931**	0.0109
Focal	0.877	0.922	0.922	0.0383

Table 2: Validation accuracy summary for ViT on the mixed dataset

On the mixed dataset, CE not only converges to a higher plateau but also exhibits markedly lower variability than Focal. Combined with the results on *Red Daikon* and *Bright&Spicy*, these findings support using CE as the default objective in subsequent experiments, reserving Focal for targeted ablations where emphasis on harder examples is explicitly desired.

4.2 ConvNeXt

4.2.1 Hyperparameter Tuning

According to section 3.4.3, to fine-tune ConvNeXt and find the best configuration, a sweep module ran on W&B with the Bayesian search. Table 3 is the configuration that yielded the fastest stable convergence and the highest plateau accuracy.

Component	**Setting**
Optimizer	AdamW (betas $= [0.95, 0.999]$, $\epsilon = 1 \times 10^{-8}$)
Learning rate	5.74×10^{-5}
Weight decay	1×10^{-5}
Batch size	32
Epochs	40
Warm-up ratio	0.2
Loss	Cross-entropy with label smoothing ($\varepsilon = 0.05$)
Class weights	False
Dropout rate	0.5
Rotation	$15°$
Brightness jitter	0.2
Transfer schedule	`freeze_epochs` $= 5$ (head-only), then full fine-tuning

Table 3: Best ConvNeXt hyperparameter configuration

AdamW with modest weight decay and $(\beta_1, \beta_2) = (0.95, 0.999)$ stabilized early updates while maintaining momentum for the later plateau. A warm-up ratio of 0.2 reduced initial oscillations from the unfrozen head, and the 5-epoch freeze allowed the classifier to settle before adapting the backbone. Label smoothing ($\varepsilon = 0.05$) improved calibration and mitigated overconfidence without relying on class weights. Dropout 0.5 and mild spatial/photometric jitter (rotation $15°$, brightness 0.2) provided additional regularization consistent with the augmentation policy used for ViT.
Figure 16 shows validation accuracy for six sample runs from the sweep; each curve corresponds to a unique hyperparameter combination.

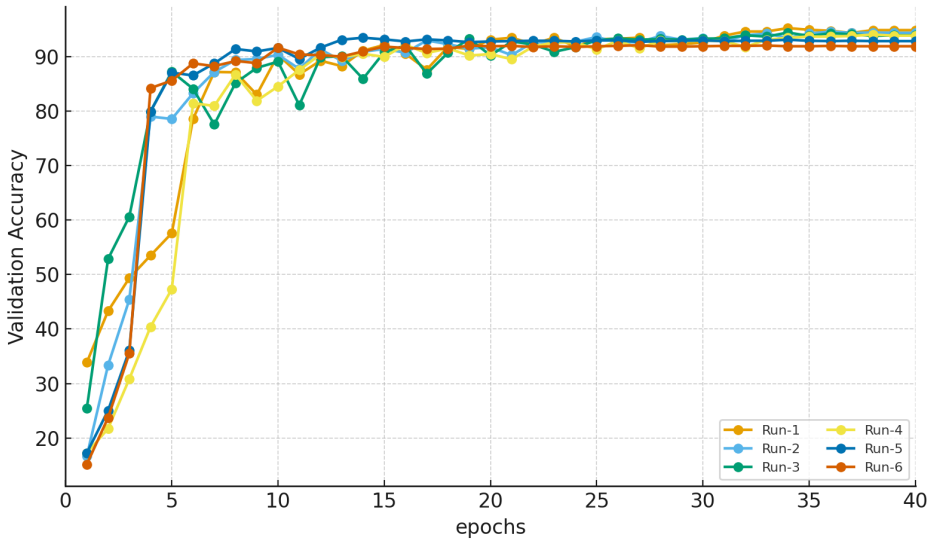


Figure 16: Validation accuracy across a few ConvNeXt sweep runs on the mixed dataset

All runs exhibit rapid gains during the first 8–12 epochs, followed by a stable plateau in the 92–95% range. Compared with the ViT sweeps, the ConvNeXt curves are notably smoother post warm-up, reflecting the stronger inductive bias of convolutional features under this data regime. The best-performing settings combine AdamW, label smoothing, and a short frozen phase, which together produce a higher, smoother plateau without resorting to heavy class weighting. Table 4 compares the similarities among the sweeps of the trained ConvNeXt on the mixed dataset. Volatility σ_Δ is the standard deviation of epoch-to-epoch accuracy changes; lower values indicate smoother learning. All figures mirror the ViT analysis protocol for comparability.

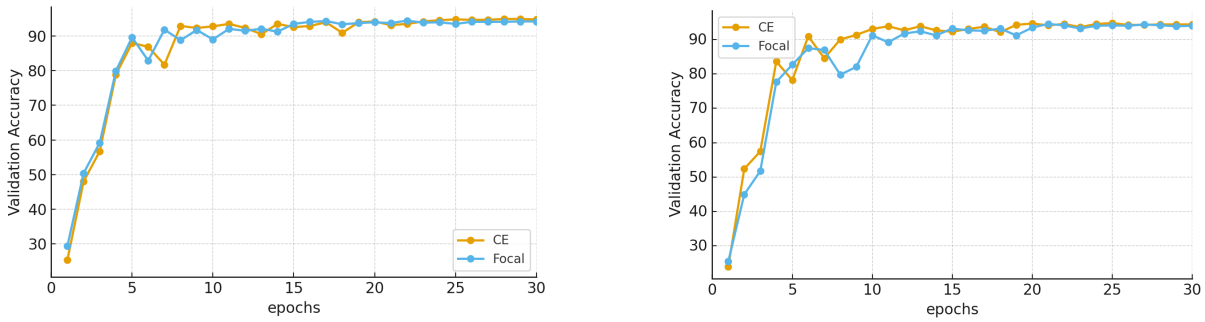
Run	**Mean Acc**	**Max Acc**	**Final Acc**	**Volatility (σ_Δ)**
Run-1	0.860	0.952	**0.948**	0.0436
Run-2	0.867	0.946	0.943	0.0626
Run-3	0.871	0.944	0.939	0.0627
Run-4	0.833	0.940	0.938	0.0600
Run-5	0.869	0.935	0.928	0.0732
Run-6	0.860	0.921	0.919	0.0804

Table 4: Validation accuracy summary for ConvNeXt across six sweep runs

Comparison to ViT tuning. Relative to ViT, ConvNeXt reaches its plateau in fewer epochs and with lower post-warm-up variance, consistent with its convolutional inductive bias and the staged fine-tuning (freeze then unfreeze). While focal loss (with γ) was explored in the sweep, the best configuration used *cross-entropy with label smoothing*, which delivered superior calibration and stability under the chosen augmentation scheme.

4.2.2 Training on Daikon set and Bright&Spicy set

We trained ConvNeXt with Cross-Entropy (CE) and Focal Loss on the *Red Daikon* and *Bright&Spicy* subsets. Figure 17 shows epoch-wise validation accuracy for each loss on both datasets.



(a) Red Daikon

(b) Bright&Spicy Mix

Figure 17: Validation accuracy over epochs for ConvNeXt trained with CE and Focal

On *Red Daikon*, CE and Focal achieve similar mean accuracy over the first 30 epochs (87.22 vs. 87.44), with comparable volatility (standard deviation of epoch-to-epoch changes σ_Δ of 6.51 vs. 6.18). CE ends slightly higher at epoch 30 (94.81 vs. 94.22), while Focal attains a similar peak (94.46). On *Bright&Spicy*, CE maintains a higher mean (87.42 vs. 85.72) and a higher final accuracy at epoch 30 (94.41 vs. 94.02), albeit with somewhat higher volatility (7.64 vs. 6.43). Overall, CE provides the more reliable trajectory on *Bright&Spicy*, whereas both losses are largely comparable on *Daikon*.
Comparing datasets, both losses reach $\approx$94–95% by epoch 30 on *Red Daikon* and *Bright&Spicy*. The early-epoch ramp is slightly steeper on *Red Daikon*, but *Bright&Spicy* shows a similarly high plateau by mid-training. Mean accuracy over the window is marginally higher for *Red Daikon* under Focal and nearly identical under CE, suggesting that both domains are comparably learnable for ConvNeXt, with a small stability edge for CE on *Bright&Spicy*.

Dataset	**Loss**	**Mean Acc**	**Max Acc**	**Final Acc**	**Volatility (σ_Δ)**
Daikon	Cross-Entropy	0.872	0.949	**0.948**	0.0651
Daikon	Focal	0.874	0.945	0.942	0.0618
Bright&Spicy	Cross-Entropy	0.874	0.948	**0.944**	0.0764
Bright&Spicy	Focal	0.857	0.945	0.940	0.0643

Table 5: Validation accuracy summary for ConvNeXt on *Daikon* and *Bright&Spicy*

Both losses reach high plateaus on both datasets. CE is slightly more consistent on *Bright&Spicy* and finishes higher on both splits, while Focal yields a comparable average on *Red Daikon*. Given these results and the tuning outcomes, CE remains the default objective for ConvNeXt in subsequent experiments, with Focal reserved for ablations emphasizing hard examples.

4.2.3 Training on the Mixed set

As figure 18 and table 6 show, CE exhibits a slightly higher mean validation accuracy and a marginally higher final plateau compared with Focal, with comparable volatility. Over the first 30 epochs, CE achieves a mean of 0.881 with a final accuracy of 0.953, whereas Focal attains a mean of 0.871 and a final accuracy of 0.951. Volatility of epoch-to-epoch accuracy changes (σ_Δ) is modest and similar across the two losses (5.72 for CE vs. 6.19 for Focal), indicating broadly stable convergence once past the initial ramp.

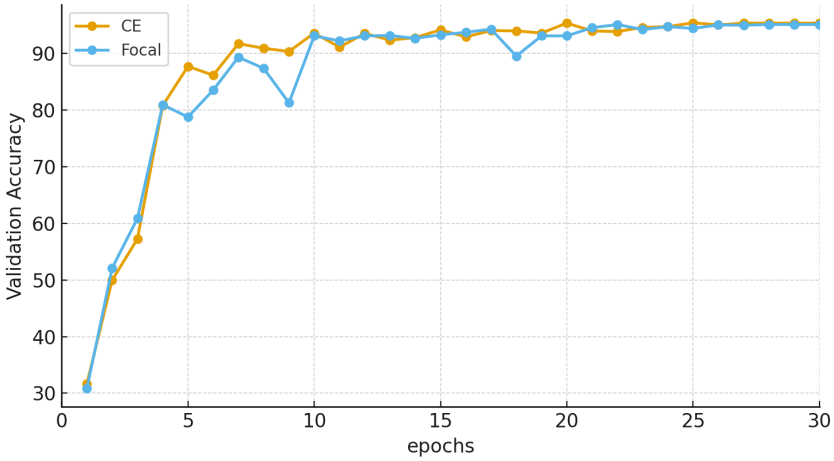


Figure 18: Validation accuracy for ConvNeXt trained with CE and Focal loss functions

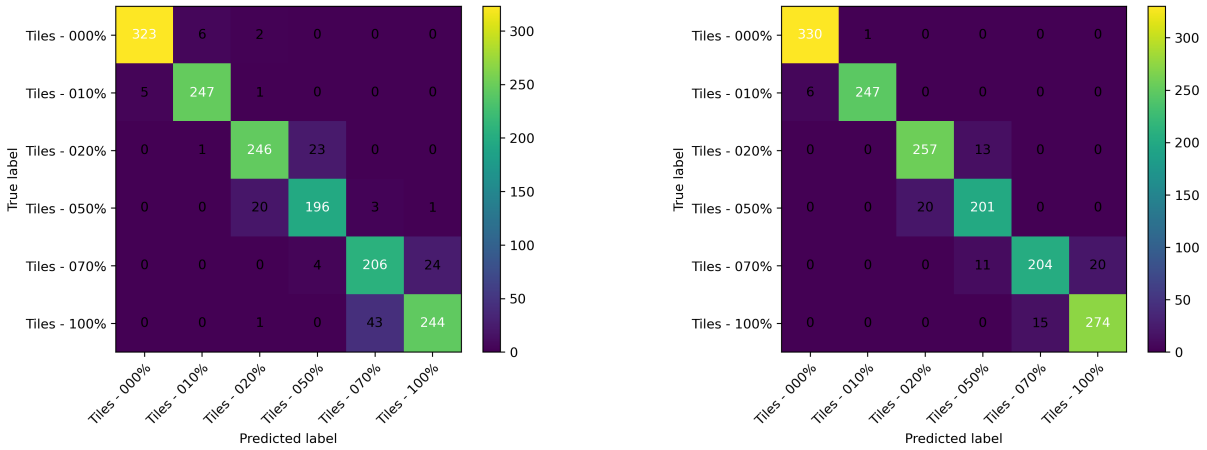
On the heterogeneous mixed domain, CE outperforms Focal with a slightly higher mean and final accuracy, while both losses converge smoothly after the initial 6–10 epochs. Taken together with the Red Daikon and Bright&Spicy results, these findings support CE as the default objective for ConvNeXt in subsequent experiments, with Focal reserved for targeted ablations.

Loss	Mean Acc	Max Acc	Final Acc	Volatility (σ_Δ)
Cross-Entropy	0.881	0.954	**0.953**	5.72
Focal	0.871	0.951	0.951	6.19

Table 6: Validation accuracy summary for ConvNeXt on the mixed dataset.

Comparing ViT and ConvNeXt results. In this section, the results of most accurate ViT and ConvNeXt trained on the mixed dataset are discussed, based on their confusion matrices, which is a table that shows how well a classification model performs by comparing the model's predicted labels with the actual labels.
The confusion matrices shown in figure 19 explains that **ConvNeXt** performs slightly better than ViT overall on validation set. ConvNeXt has more correct predictions along the diagonal and fewer misclassifications, especially for higher tile percentages (e.g., class–070% and class–100%). In contrast, ViT shows more confusion between neighboring classes, such as class–020% and class–050%. This indicates that ConvNeXt provides more consistent and accurate class separation on the mixed dataset. It is also important to note that a certain level of

confusion between adjacent classes is acceptable, as neighboring categories (e.g., class–050% and class–070%) may share similar morphological features. However, the model should not confuse labels that are morphologically very different. For instance, ViT incorrectly classified two samples from class–000% as class–020%, which differ significantly in appearance—an unexpected and undesired error pattern.



(a) Vit

(b) ConvNeXt

Figure 19: Comparing the models' confusion matrices trained on the mixed dataset

5 Discussion

Although it was expected that the *focal loss* function would perform better than *cross-entropy* due to the imbalanced nature of the dataset, the results showed the opposite trend. The CE loss achieved higher overall accuracy and produced smoother learning curves during training. This suggests that, in this context, CE provided a more stable optimization process and was less sensitive to variations in class frequency. Additionally, while the *Vision Transformer (ViT)* architecture was anticipated to struggle with convergence because of its data-hungry nature, the use of effective data augmentation techniques helped it learn successfully. These augmentations increased data diversity and regularized training, allowing ViT to reach reliable convergence despite the relatively limited dataset size.

6 Limitations

Manual data collection and annotation required significant time and effort, as each image patch had to be carefully captured and labeled. This manual process limited the overall dataset size and slowed experimentation. In future work, an accurate automated pipeline could continuously collect and label high-quality data with minimal human input. This would make the system more efficient, scalable, and suitable for large-scale commercial applications.

7 Future Work

Explainable AI (XAI) can be used to better understand how both architectures make their predictions. For ConvNeXt, gradient-based visualization methods such as Grad-CAM or Grad-CAM++ can highlight which parts of the image the model focuses on. This helps verify whether the network pays attention to meaningful biological features, rather than unrelated background patterns. For ViT, attention-based visualization methods can show how image regions interact and whether the model uses overall spatial context, such as the distribution of seedlings, when making decisions.

The current results indicate that the proposed pipeline can be adapted to *different crop species*, as long as image collection accounts for specific plant characteristics. Future datasets should include multiple cultivars and leaf shapes for each crop and basic preprocessing tailored to each crop, such as background normalization, glare reduction, and color calibration. These steps would make the model more robust and improve its generalization across species and growing facilities.

Because ViT is more sensitive to data augmentation and training settings, several strategies could be explored. Label-preserving augmentations like RandAugment or AugMix can increase data diversity without changing class meanings. MixUp, which combines images by blending their intensity values, may be more suitable for germination labels than CutMix, as it avoids mixing distinct spatial regions. A gradual augmentation schedule — starting with mild transformations and increasing strength during training — could also improve model stability.

Finally, the data cleaning process, which was done manually, can also be automated, as an example in the way that Qin et al. (2023) suggested. A simple object detection model can be trained to identify and remove unusable image patches. This automated filtering would make dataset preparation faster, more consistent, and easier to scale for larger experiments.

8 Conclusion

This study evaluated two modern architectures—Vision Transformer (ViT) and ConvNeXt—for patch-level germination classification. Across (*Red Daikon*, *Bright&Spicy Mix*) and mixed datasets, ConvNeXt exhibited smoother optimization dynamics and reached high plateaus rapidly, while ViT required more training loops and showed greater epoch-to-epoch variability. In loss-function comparisons, cross-entropy (CE) consistently delivered higher final accuracy and lower volatility than Focal Loss, particularly on the mixed set. These findings suggest that, within the current data and augmentation regime, CE provides a more reliable baseline, and ConvNeXt offers stronger inductive bias for stable convergence.

A central design decision was the use of CutMix for ViT as a regularization mechanism. Although CutMix improved robustness by exposing the model to mixed spatial contexts, it also introduced a conceptual mismatch with the discrete nature of germination stages: the interpolated label targets do not correspond to existing categories.

To conclude, the comparative evidence favors CE as the default objective and highlights ConvNeXt's stability under modest data, while also ViT proves its promising predictions where it is not expected to perform accurately.

References

Andreas Kamilaris, Francesc X. Prenafeta-Boldú (2018). "Deep learning in agriculture: A survey". In: *Computers and Electronics in Agriculture* 147, pp. 70–90. DOI: 10.1016/j.compag.2018.02.016.

Ashok, Anjali and Mary Adesoba (2023). "Plant yield prediction in indoor farming using machine learning". PhD thesis. University of Skövde, School of Informatics. URL: https://urn.kb.se/resolve?urn=urn:nbn:se:his:diva-23201.

Beacham, Andrew M., Lucy H. Vickers, and James M. Monaghan (2019). "Vertical farming: a summary of approaches to growing skywards". In: *The Journal of Horticultural Science and Biotechnology* 94.3, pp. 277–283. DOI: 10.1080/14620316.2019.1574214.

Benke, Kurt and Bruce Tomkins (2017). "Future food-production systems: vertical farming and controlled-environment agriculture". In: *Sustainability: Science, Practice and Policy* 13.1, pp. 13–26. DOI: 10.1080/15487733.2017.1394054.

Al-Chalabi, Malek (2015). "Vertical farming: Skyscraper sustainability?" In: *Sustainable Cities and Society* 18, pp. 74–77. DOI: 10.1016/j.scs.2015.06.003.

Coll, Bartomeu and Jean-Michel Morel (Sept. 2011). "Non-Local Means Denoising". In: *Image Processing On Line* 1. DOI: 10.5201/ipol.2011.bcm_nlm.

Cui, Yin et al. (2019). "Class-Balanced Loss Based on Effective Number of Samples". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9268–9277.

Despommier, Dickson (2010). *The Vertical Farm: Feeding the World in the 21st Century*. Thomas Dunne Books.

Dias, M. C. et al. (2011). "Computer vision for monitoring seed germination from dry state to young seedlings". In: *Seed Testing International*.

Dosovitskiy, A., L. Beyer, A. Kolesnikov, et al. (2020). "An image is worth 16x16 words: Transformers for image recognition at scale". In: *International Conference on Learning Representations (ICLR)*.

Fuentes-Peñailillo, Fernando et al. (2023). "Automating Seedling Counts in Horticulture Using Computer Vision and AI". In: *Horticulturae* 9.10. ISSN: 2311-7524. DOI: 10.3390/horticulturae9101134. URL: https://www.mdpi.com/2311-7524/9/10/1134.

Han, L. et al. (2025). "Application of Deep Learning Technology in Monitoring and Managing Agricultural Crops". In: *Sustainability* 17.17, p. 7602. DOI: 10.3390/su17177602.

He, Kaiming et al. (2015). "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778. URL: https://api.semanticscholar.org/CorpusID:206594692.

He, L. et al. (2025). *New computer vision system can guide specialty crops monitoring*. [Online; accessed 21-October-2025]. URL: https://www.psu.edu/news/research/story/new-computer-vision-system-can-guide-specialty-crops-monitoring.

Irmawati, Irmawati et al. (Dec. 2023). "Early Detection of Potato Leaf Pest and Disease Using EfficientNet and ConvNeXt Architecture". In: pp. 167–172. DOI: 10.1109/CONMEDIA60526.2023.10428527.

Kalra, Dayal and Maissam Barkeshli (June 2024). *Why Warmup the Learning Rate? Underlying Mechanisms and Improvements*. DOI: 10.48550/arXiv.2406.09405.

Kozai, Toyoki (2018). *Smart Plant Factory: The Next Generation Indoor Vertical Farms*. Springer. DOI: 10.1007/978-981-13-1065-2.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey Hinton (Jan. 2012). "ImageNet Classification with Deep Convolutional Neural Networks". In: *Neural Information Processing Systems* 25. DOI: 10.1145/3065386.

Liakos, Konstantinos et al. (2018). "Machine learning in agriculture: A review". In: *Sensors* 18.8, p. 2674. DOI: 10.3390/s18082674.

Lin, T.-Y. et al. (2017). "Focal loss for dense object detection". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988.

Liu, Wei et al. (2016). *SSD: Single shot multibox detector*. URL: https://doi.org/10.1007/978-3-319-46448-0_2.

Liu, Zhuang et al. (2022). "A ConvNet for the 2020s". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11966–11976. DOI: 10.1109/CVPR52688.2022.01167.

Mohanty, Sharada P., David P. Hughes, and Marcel Salathé (2016). "Using Deep Learning for Image-Based Plant Disease Detection". In: *Frontiers in Plant Science* Volume 7 - 2016. DOI: 10.3389/fpls.2016.01419. URL: https://www.frontiersin.org/journals/plant-science/articles/10.3389/fpls.2016.01419.

Nico Heider, Lorenz Gunreben (2025). "A survey of datasets for computer vision in agriculture". In: *Gesellschaft für Informatik e.V.* DOI: 10.18420/GILJT2025_02.

OpenCV (2025). *Template Matching*. [Online; accessed 21-October-2025]. URL: https://docs.opencv.org/4.x/d4/dc6/tutorial_py_template_matching.html.

Padhiary, Mrutyunjay et al. (Mar. 2025). "Advances in Vertical Farming: The Role of Artificial Intelligence and Automation in Sustainable Agriculture". In: *LatIA* 3, p. 131. DOI: 10.62486/latia2025131.

Panotra, B. R. et al. (2024). "Vertical Farming: Addressing the Challenges of 21st Century Agriculture through Innovation". In: *International Journal of Environment and Climate Change*.

Qin, Jiale et al. (2023). "Deep-Learning-Based Rice Phenological Stage Recognition". In: *Remote Sensing* 15.11. ISSN: 2072-4292. DOI: 10.3390/rs15112891. URL: https://www.mdpi.com/2072-4292/15/11/2891.

Redmon, Joseph et al. (2015). "You Only Look Once: Unified, Real-Time Object Detection". In: *CoRR* abs/1506.02640. URL: http://arxiv.org/abs/1506.02640.

Ren, Shaoqing et al. (2015). "Faster R-CNN: Towards real-time object detection with region proposal networks". In: *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 91–99.

Schroeder, Alexandra B. et al. (2021). "The ImageJ ecosystem: Open-source software for image visualization, processing, and analysis". In: *Protein Science* 30.1, pp. 234–249. URL: https://doi.org/10.1002/pro.3993.

Simonyan, Karen and Andrew Zisserman (2015). *Very Deep Convolutional Networks for Large-Scale Image Recognition*. arXiv: 1409.1556 [cs.CV]. URL: https://doi.org/10.48550/arXiv.1409.1556.

Song, Chang et al. (2023). "Maize seed appearance quality assessment based on improved Inception-ResNet". In: *Frontiers in Plant Science* Volume 14 - 2023. ISSN: 1664-462X. DOI: 10.3389/fpls.2023.1249989.

Sun, Wenbin et al. (2025). "CSGD-YOLO: A Corn Seed Germination Status Detection Model Based on YOLOv8n". In: *Agronomy* 15.1. DOI: 10.3390/agronomy15010128.

Wang, Hao and Peng Gao (2024). "Survey Of Small Object Detection Methods Based On Deep Learning". In: *2024 9th International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*. Vol. 9, pp. 221–224. DOI: 10.1109/ICIIBMS62405.2024.10792837.

Wang, Xiaoqi et al. (2023). "ECA-ConvNeXt: A Rice Leaf Disease Identification Model Based on ConvNeXt". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 6235–6243. DOI: 10.1109/CVPRW59228.2023.00663.

Wiener, Norbert (Aug. 1949). *Extrapolation, Interpolation, and Smoothing of Stationary Time Series: With Engineering Applications*. The MIT Press. DOI: 10.7551/mitpress/2946.001.0001.

Yao, Qiong et al. (2024). "SGR-YOLO: a method for detecting seed germination rate in wild rice". In: *Frontiers in Plant Science* Volume 14 - 2023. DOI: 10.3389/fpls.2023.1305081.

Yun, Sangdoo et al. (2019). "CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6023–6032.

Zhao, Jinfeng et al. (2023). "Deep-learning-based automatic evaluation of rice seed germination rate". In: *Journal of the Science of Food and Agriculture* 103.4, pp. 1912–1924. URL: https://doi.org/10.1002/jsfa.12318.