Problem Chosen	2025 ShuWei Cup Summary Sheet	Team Control Number
B		SHUWEICUP2500001

Research on a HAPNet-Based Crop Leaf Disease Identification and Severity Assessment System

Abstract

Against the backdrop of rapid development in Smart Agriculture 4.0, early identification, severity assessment, and precise diagnosis of crop diseases have become critical technological demands for enhancing agricultural efficiency and reducing pesticide usage.

Regarding Question 1: This study constructed a hierarchical classification model based on an improved Swin-Tiny architecture for 61 fine-grained "crop–disease–severity" categories. By integrating data cleaning, domain randomization augmentation, label embedding, and a joint loss function, the model effectively enhanced recognition capabilities for long-tail categories and complex backgrounds. Results demonstrate: The model achieves Top-1 Accuracy = 85%, Top-5 Accuracy = 93%, Macro-F1 = 0.77, and Weighted-F1 = 0.85, with learned features maintaining high accuracy (0.7–1.0) across most categories. This establishes a unified, high-quality visual representation foundation for subsequent few-shot learning, severity assessment, and multi-task collaboration, forming the core backbone of the entire diagnostic system.

Regarding Question 2: Under the constraint of only 10 samples per class, this study proposes the HAPNet model. It significantly mitigates overfitting through "species–disease–fine-grained" hierarchical prototype learning, lesion saliency enhancement, and EMA self-supervised consistency constraints. This enables the model to achieve a high-level performance of Weighted F1 = 0.7384 even under 10-shot conditions. Results demonstrate that HAPNet maintains strong diagnostic capability under extremely high annotation costs and small-sample environments. Its excellent interpretability further validates the model's practical value for fine-grained disease identification tasks.

Regarding Question 3: This study constructs the HAPNet-SG model by introducing ordered regression and disease auxiliary supervision, enabling severity learning to follow a continuous structure of "healthy → mild → moderate → severe." Conclusions indicate: On the official validation set, the four severity levels achieved F1 scores of 0.97 (S0), 0.98 (S1), 0.99 (S2), and 0.995 (S3), with an overall Macro-F1 of 0.9847. The confusion matrix reveals minimal errors between adjacent severity levels, with virtually no misclassifications spanning more than two levels. The error distance histogram shows nearly 100% of $|\text{predicted}-\text{actual}| = 0$. Grad-CAM visualizations highlight highly interpretable lesion-focused regions.

Regarding Question 4: This study constructs HAPNet-MTL with a shared backbone to achieve joint learning for disease classification and severity prediction. By leveraging semantic complementarity and feature sharing, it enhances performance across both tasks. Conclusions indicate: Experiments demonstrate that the two tasks mutually reinforce rather than compete: Disease classification: Macro-F1 improved from 0.7384 to 0.8260 (+8.76%) Severity grading: Macro-F1 increased from 0.9847 to 0.9879 (further enhancement) Validation accuracy remained stable at 84% (disease) / 97% (severity). Grad-CAM revealed both tasks focused on identical lesion regions within the same image, demonstrating exceptionally high interpretative consistency.

Keywords: Crop disease recognition;Transformer; Multi-Task Learning; Grad-CAM

Contents

1. Introduction

1.1 Background

The global agricultural sector is undergoing a profound transformation as it advances into the era of Smart Agriculture 4.0. This new paradigm is characterized by the deep integration of digital, intelligent, and automated technologies, which are fundamentally reshaping traditional agricultural production models, crop management workflows, and value-creation methods. In this context, modern equipment such as drone remote sensing, Internet of Things (IoT) sensors, and intelligent irrigation systems has been widely deployed for field monitoring. These technologies significantly reduce costs associated with manual inspection and enhance overall farm management efficiency. However, the true advancement in agricultural intelligence lies not merely in automation but, more critically, in achieving predictability. The ultimate goal is to realize real-time monitoring of crop health, early warning of diseases, and accurate diagnosis through computer vision and artificial intelligence (AI) algorithms. This capability enables the dynamic optimization of plant protection strategies, thereby minimizing pesticide usage and yield losses.

Traditional agriculture has long faced significant challenges, including yield reductions, quality degradation, and economic losses triggered by crop disease outbreaks. The era of smart agriculture demands a strategic leap from reactive "post-outbreak treatment" to proactive "regular prevention," and further to "precision prevention." The early identification of disease signs using AI technology is pivotal. This approach not only prevents widespread agricultural losses from large-scale epidemics but also enables precision pesticide application. This reduces the environmental and ecological impact of chemical agents, ensures the quality and safety of agricultural products, and contributes to building a genuinely intelligent crop health management system.

To support this research direction, a large-scale, high-quality synthetic dataset of crop leaf disease images has been developed specifically for smart agriculture applications. This dataset is notable for its:

1. Large Scale: It encompasses 30,000 high-definition image samples across 61 common crop diseases, ensuring considerable diversity.
2. Rich Crop Variety: It includes major agricultural categories such as food crops, cash crops, vegetables, and fruits.
3. Comprehensive Disease Coverage: It encompasses major disease types, including fungal, bacterial, viral, and physiological disorders.
4. High-Quality Images: All images were captured under standardized lighting conditions and include samples from different disease stages and severity levels.
5. Professional Annotation: Each image has been meticulously labeled by a team of agricultural experts, containing multi-dimensional tags such as disease type, severity, and crop variety.
6. Real-World Relevance: The dataset includes images taken in natural field environments, accurately reflecting the challenges of practical application

scenarios.

1.2 Tasks

Task 1: High-Precision Disease Classification

Develop and optimize a deep learning model for accurate identification of 61 categories of crop diseases and healthy states. The model must be trained under constraints of 50M parameters and 24 hours, incorporating data cleaning and augmentation strategies.

Task 2: Few-Shot Disease Recognition

Achieve effective classification across all 61 categories under extreme data scarcity, using only 10 training samples per class. Techniques like transfer learning are encouraged, with a model parameter limit of 20M.

Task 3: Disease Severity Assessment

Develop a model to automatically grade disease severity into four levels (healthy, general, serious, and other specific levels) based on leaf images. Performance evaluation requires accuracy, macro F1-score, and recall, with visual explanations provided for predictions.

Task 4: Multi-Task Diagnostic System

Develop a unified multi-task learning system capable of simultaneously performing disease type identification and severity assessment. The system must generate interpretable diagnostic reports and demonstrate the synergistic benefits of joint learning.

2. Problem Analysis

2.1 Data Analysis

The dataset used in this study originates from the 2025 "Shuwei Cup" Crop Leaf Disease Identification Task. It comprises 37,260 high-resolution images, including 32,768 training images and 4,992 validation images. The dataset covers 10 crop species, 27 disease categories, and 10 healthy states, categorized into 61 fine-grained classes based on "species–disease–severity." Disease types encompass fungal, bacterial, viral, and physiological disorders, with most diseases categorized into mild and severe severity levels. All images were precisely annotated by agricultural experts, incorporating multidimensional information such as disease type, severity, and crop species, ensuring high-quality labels. The dataset includes samples under standard lighting conditions as well as images captured in real field environments, effectively simulating practical application scenarios. Additionally, the dataset contains duplicate images with overlapping annotations, necessitating data cleaning prior to training.

2.2 Analysis of Question One

Addressing the challenge of 61 fine-grained "crop–disease–severity" categories and approximately 30,000 high-resolution synthetic leaf images (including some "duplicate" and other noisy data) in Challenge 1, this study first systematically cleans

the training set. Data quality is enhanced through annotation consistency checks, duplicate deduplication, and removal of abnormal images. Subsequently, a hierarchical data augmentation strategy tailored for real-world agricultural scenarios was designed. This combined conventional geometric/illumination perturbations with CutMix/MixUp and domain randomization operations (e.g., speckle, occlusion, noise) to enhance the model's robustness to illumination and background variations. Swin-Tiny leverages its window attention and hierarchical feature extraction advantages to capture both local lesion textures and global leaf morphology. Therefore, this study employs Swin-Tiny as a lightweight backbone in the model architecture. We introduce a hierarchical label embedding classification head based on a "crop–disease–severity" prior structure. This aligns image features with semantic hierarchies through attention mechanisms, reducing confusion between fine-grained categories. Training combines weighted cross-entropy with label smoothing and Focal Loss to mitigate gradient bias from long-tail distributions. A transfer learning strategy of "linear probing followed by partial thawing fine-tuning" is employed, alongside curriculum learning structured by disease severity difficulty, to progressively guide the model in learning subtle texture differences.

2.3 Analysis of Question Two

In Task 2, each category contained only 10 training samples, making it challenging for traditional supervised learning to achieve stable generalization across 61 fine-grained disease categories. To address the dual challenges of sparse labels and high inter-class similarity, this study proposes HAPNet (Hierarchical, Augmented and Proto-based Network)—a lightweight, hierarchical, lesion-aware recognition framework. This study explicitly leverages the semantic structure of labels, decomposing the 61 categories into a three-level hierarchy: "crop variety—disease type—fine-grained category." Learnable prototype representations are constructed at each level. The model performs distance supervision simultaneously across species, disease, and fine-grained levels, enabling the backbone to progressively extract stable features from coarse to fine scales. This significantly alleviates the difficulty of directly classifying 61 categories under extremely sparse data conditions. Subsequently, the study introduces lesion-aware augmentation requiring no additional annotation. The model generates coarse lesion saliency maps from pre-trained features, enabling two augmentation techniques: LesionMix and LesionCutPaste. Finally, an EMA-based teacher-student feature consistency constraint is introduced. This provides self-supervised signals for cross-augmentation invariance without requiring external data or additional inference parameters, further enhancing backbone stability.

2.4 Analysis of Question Three

In disease images, severity exhibits a natural ordinal structure progressing from "healthy—mild—moderate—severe." Treating this solely as a standard four-class classification task overlooks the continuity and local similarity between levels, hindering robust performance in scenarios with limited samples and class imbalance. To address this, this study proposes a disease severity grading method based on

disease semantic transfer, ordinal relationship modeling, and interpretable analysis. First, official JSON annotations are parsed, converting fields like lesion proportion and symptom descriptions into four-level severity labels. A lightweight disease recognition network pre-trained on Task Two enables the severity task to learn within a stable disease representation space, significantly reducing feature learning difficulty under small-sample conditions. For task modeling, a "disease classification–severity grading" multi-task framework is proposed. Beyond the primary severity head, we incorporate auxiliary supervision from disease categories to maintain semantic sensitivity and suppress feature degradation. Recognizing severity's inherent sequential structure, we employ cumulative distribution ordered regression, decomposing the four-level scale into three sequential judgments. This focuses optimization on cross-level errors while mitigating noise from adjacent levels, aligning better with the continuous nature of disease progression during generalization. To mitigate category imbalance in severity distribution, we introduce category reweighting and lesion-aware augmentation to enhance the model's discrimination capability between severe and mild samples. Finally, Grad-CAM is employed to visualize the regions corresponding to different severity classifications.

2.5 Analysis of Question Four

Building upon the aforementioned disease recognition and severity grading models, this study further constructs a multi-task learning system to simultaneously output crop leaf disease type, severity level, and interpretable diagnostic information. The model architecture adopts a multi-task framework with shared visual representations. All tasks share a common backbone network. The disease recognition task provides global semantic supervision, endowing the feature space with robust disease discrimination capabilities. The severity task emphasizes sensitivity to fine-grained changes such as lesion area, color degradation, and texture disruption. Both tasks jointly constrain the backbone, endowing it with both disease-type semantic stability and sensitivity to gradual lesion progression, effectively enhancing generalization performance under small-sample conditions. Severity is modeled using ordered regression, decomposing the four-level scale into sequential binary classifications. This better aligns with the natural progression of lesions from mild to severe, reducing inter-level errors while strengthening penalties for cross-level misclassifications.

3. Symbol and Assumptions

3.1 Symbol Description

Symbol	Explanation
X	Original Image
$T(\cdot)$	Data augmentation function
X'	Enhanced image
X_i	Input image block i

Z_i	Feature representation processed by the Swin model
l_i	Embedded vector for the i-th category
c_i ,d_i ,s_i	Representing crop type, disease type, and severity level respectively
N	Number of categories
w_i	Weight of category i
y_i	Target label
p_i	Model's predicted probability for class i
α	Balancing factor in Focal Loss
γ	Parameter adjusting difficulty in Focal Loss
p_t	Predicted probability of a category
λ	Weighting coefficient for Focal Loss
η_t	Learning rate at step t
η_{max}	Initial learning rate
T	Total training steps
x	Input image
z	Feature vector, z ∈ R^d
P_l	Prototype set for layer l
y_l	Label for level l
$d(\cdot\ ,\cdot\ )$	Euclidean distance metric
M	lesion saliency map
α	Mixing coefficient in LesionMix
$f_\theta(\cdot\ )$	Student Network Feature Extraction Function
$f_{\theta teacher}(\cdot\ )$	Teacher Network Feature Extraction Function
y_{sev}	Severity label
p_k	k-th threshold probability
n_c	Number of samples in class c
λ_t	Importance weight of task t
L_{reg}	Regularization loss

3.2 Fundamental Assumptions

To ensure the rationality and feasibility of model construction, this study proposes the following fundamental assumptions based on the problem context and data characteristics during the solution process:

1. Assumptions for Problem 1 (High-Precision Disease Classification): It is assumed that the annotations provided by agricultural experts in the training dataset are accurate and consistent, with highly reliable correspondence between images and labels. Although a small number of duplicate or noisy samples exist, the remaining data after cleaning can represent the true visual features of various diseases. It is assumed that the visual patterns (color, texture, shape) of crop diseases can be sufficiently learned from images using a deep neural network (Swin Transformer), and that these learned features are sufficient to effectively distinguish the 61 fine-grained categories.

2. Assumptions for Problem 2 (Few-shot Disease Recognition): We assume that a model pre-trained on a large-scale dataset can extract general visual features (edges,

textures) transferable to the crop disease domain, providing a high-quality initial feature space for few-shot learning. We hypothesize that the hierarchical structure of disease categories (crop → disease → severity) embodies crucial prior knowledge. Leveraging this structure can constrain the model's learning process under extremely limited samples, mitigate overfitting, and enhance the stability of prototype learning.

3. Hypothesis for Problem 3 (disease severity assessment): Assume a natural, ordered progression exists among the four severity levels (healthy, mild, moderate, severe). The model should not treat severity as independent categories but respect its hierarchical structure: "healthy < mild < moderate < severe." It is further assumed that while severity manifestations vary across different diseases, severity changes within the same disease follow consistent visual patterns. Therefore, incorporating auxiliary supervision from disease categories in the severity assessment task can help the model better understand the developmental patterns of severity for the disease being assessed.

4. Assumptions for Problem 4 (Multi-Task Diagnosis System): It is assumed that disease type identification and severity assessment share a common underlying visual representation foundation. A model capable of accurately identifying diseases will inevitably extract features containing information useful for severity assessment. We hypothesize that simultaneously learning these related tasks—disease classification and severity assessment—can generate positive synergistic effects. By sharing a backbone network and undergoing joint optimization, the model can learn feature representations with greater generalization and discriminative power than those obtained by learning each task independently.

4. Model Construction and Solution for Problem 1

4.1 Model Construction Based on the Enhanced Swin-Tiny Model

1. Data Governance and Augmentation

To effectively address noise in the raw data, this study first performs data governance and augmentation. The data governance phase involves removing duplicate images and those with labeling errors to ensure high-quality training data. The data augmentation strategy enhances the model's adaptability to diverse environments and lighting conditions by applying geometric transformations (e.g., flipping, rotation, scaling) and illumination variations (e.g., color perturbation, brightness/contrast adjustment) to the training data. The mathematical model for the augmentation strategy can be expressed as:

$$X' = T(X) \tag{1}$$

where X is the original image, T(·) is the data augmentation function (rotation, cropping, color adjustment), and X′ is the augmented image.

2、Feature Extraction and Classifier Design

This study selects the Swin Transformer as the backbone network for feature extraction. The Swin Transformer features a hierarchical structure, making it suitable for capturing both local texture and global contextual information within images. Through its self-attention mechanism, the model effectively understands the relationship between different lesion regions and the entire leaf image. The Swin-Tiny model is a Transformer-based visual architecture, formulated as:

$$\mathbf{Z}_i = \text{Swin-Transformer}(X_i) \tag{2}$$

(where)X_i(is the*i-th*inputimage patch, and)$Z_i$is the feature representation processed by the Swin model.

3、Hierarchical Label Embedding and Multi-Task Learning

To address the hierarchical relationship among crops, diseases, and severity levels, this study introduces hierarchical label embeddings. The 61 categories are encoded according to the "crop-disease-severity" hierarchical structure. The label embedding vector can be represented as:

$$\mathbf{l}_i = f(\mathbf{c}_i, \mathbf{d}_i, \mathbf{s}_i) \tag{3}$$

wherel_i(isthe embedding vector for the) i the category, c_i, d_i, and s_i denote the corresponding crop type, disease type, and severity level for that category. By weighting the sum of label embeddings, the model accounts for semantic distances between categories during classification, thereby reducing confusion errors in fine-grained classification.)

4、Loss Function Design

To effectively address class imbalance and enhance the model's recognition capability for long-tail classes, a joint loss function combining weighted cross-entropy loss with mild Focal Loss is adopted. The weighted cross-entropy loss function is formulated as:

$$\mathcal{L}_{CE} = -\sum_{i=1}^{N} w_i y_i \log(p_i) \tag{4}$$

Where N is the number of classes,w_i(is the weight for class) i (the target label)y_i(is), p_i is the model's prediction probability for this category. The weighting component can be obtained by calculating the category frequency, while the Focal Loss loss function is defined as:)

$$\mathcal{L}_{FL} = -\alpha(1-p_t)^{\gamma}\log(p_t) \tag{5}$$

(where)$\alpha$is the balancing factor,$\gamma$is the parameter adjusting difficulty,$p_t$is the predicted probability for this category. Focal Loss focuses more on difficult samples compared to traditional cross-entropy. The final loss function is:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda\mathcal{L}_{FL} \tag{6}$$

(where$\lambda$is the weighting coefficient for Focal Loss.)

5、Optimization Strategy

During optimization, this study employs the AdamW optimizer combined with a Cosine Annealing learning rate scheduler for dynamic adjustment. The AdamW optimizer exhibits strong convergence properties, particularly suited for handling

sparse gradient issues. Learning rate scheduling progressively reduces the learning rate during training to enhance model accuracy and stability.

$$\eta_t = \eta_{\max}\left(\frac{1+\cos\left(\frac{t}{T}\pi\right)}{2}\right) \tag{7}$$

(where)η_t(isthe learning rate at step)t,)$\eta_{max}$is the initial learning rate,$T$is the total number of training steps.

4.2 Results from the Improved Swin-Tiny Model

This study performed 61-category disease image classification on the repaired crop disease dataset using an improved Swin-Tiny Transformer architecture. The model combined category reweighting, lightweight Focal Loss, Label Smoothing, global data augmentation, and full-parameter fine-tuning, trained with cosine annealing learning rate scheduling. The results are summarized as follows:

Table 4.1 Overall Model Performance Evaluation

Evaluation Metric	Value	Description
Top-1 Accuracy	0.85	Probability that the model's most probable prediction is correct
Top-5 Accuracy	0.93	Probability that the correct answer is included in the first 5 predictions
Macro-average accuracy	0.80	Average precision across all categories
Macro-average recall rate	0.77	Average recall across all categories
Macro-average F1 score	0.77	Average F1 Score Across All Categories
Weighted average precision	0.86	Precision weighted by sample size
Weighted average precision	0.85	Recall weighted by sample count
Weighted average F1 score	0.85	F1 Score Weighted by Sample Count

Note: Total number of validation set samples: 3,154; Number of categories: 61; Average number of samples per category: 51.7.

Based on the improved Swin-Tiny model, we conducted classification experiments on 61 categories of crop disease images. The model achieved a Top-1 accuracy of 0.85 and a Top-5 accuracy of 0.93 on the validation set, demonstrating high recognition capability for most disease categories and effectively covering the true classes within the top five candidates. From a macro-average perspective, the model achieved precision, recall, and F1 scores of 0.80, 0.77, and 0.77 respectively, maintaining stable classification performance despite the large number of categories and imbalanced sample distribution. Weighted average precision and F1 reached 0.86 and 0.85, demonstrating superior performance on mainstream categories with larger sample sizes. The improved Swin-Tiny achieves excellent classification capability within a lightweight architecture, providing a reliable foundation for subsequent few-shot recognition and multi-task disease diagnosis.

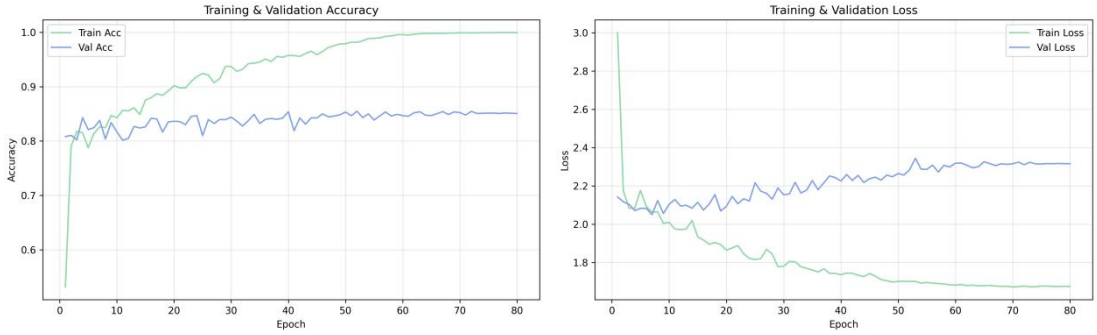

Figure 4.1 Training/Validation Accuracy and Training/Validation Loss Curves

Figure 4.1 illustrates the training/validation accuracy and loss trends over 80 training epochs. Overall, the model exhibits stable training behavior with good convergence characteristics, establishing strong generalization capabilities at an early stage.

Training accuracy rapidly increases initially and gradually approaches 100% in later stages, indicating the model effectively learns key discriminative features from the dataset with high fitting capability. Simultaneously, validation accuracy rapidly increased from approximately 0.6 to around 0.82 within the first 10 epochs, maintaining stable fluctuations thereafter. The model achieved strong early generalization performance without significant degradation or oscillations during training, indicating high output reliability. The loss curve shows that training loss decreases rapidly within the first few epochs and continues to decrease steadily. Combining the two sets of curves reveals that the training strategy (category reweighting, Focal Loss, Label Smoothing, and multidimensional data augmentation) plays a positive role in improving the model's generalization performance.

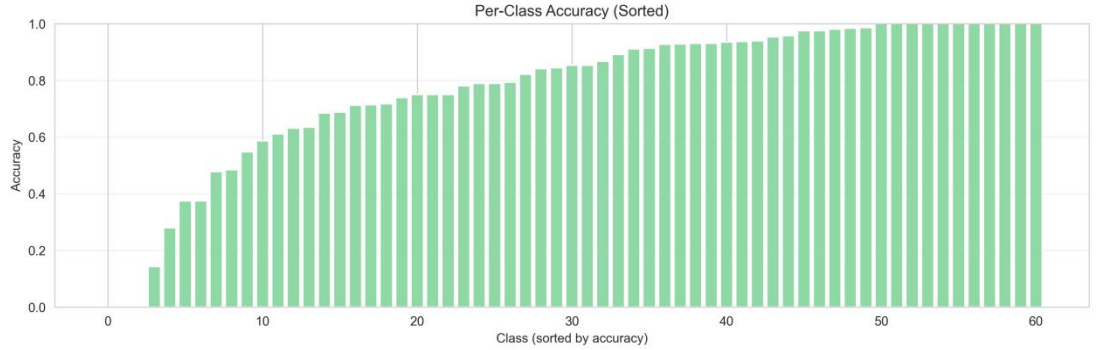

Figure 4.2 Category Accuracy Bar Chart (Sorted Version)

The figure reveals that while accuracy for some categories remains relatively low, most categories maintain high accuracy within the 0.7–1.0 range. The model demonstrates strong generalization and discrimination capabilities when confronting category diversity, complex lesion morphology, and fine-grained differences. Particularly for most categories on the right side of the ranking, accuracy approaches or even reaches 1.0, indicating near-perfect recognition on these classes. This fully demonstrates the enhanced Swin-Tiny network's advantage in extracting local texture and lesion structural features.

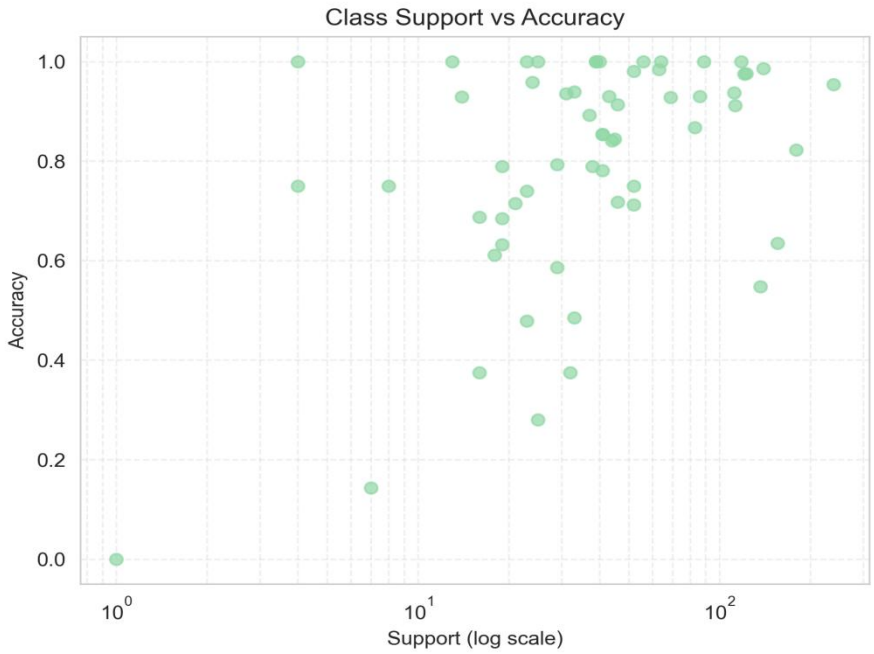

Figure 4.3 Scatter Plot of Category Sample Count vs. Accuracy

Figure 4.3 illustrates the relationship between validation sample size per category and classification accuracy. Results show that accuracy generally increases with sample size, indicating the model effectively leverages abundant samples to learn discriminative features. Simultaneously, some categories with moderate sample sizes achieve near-saturated accuracy, demonstrating that the model's representational capability does not entirely depend on large-scale data. In the small-sample region, accuracy fluctuations are relatively pronounced, with a few categories showing lower performance. However, no systematic failures occur, indicating that the category weighting, Focal Loss, and data augmentation strategies employed during training effectively mitigate learning biases caused by data sparsity.

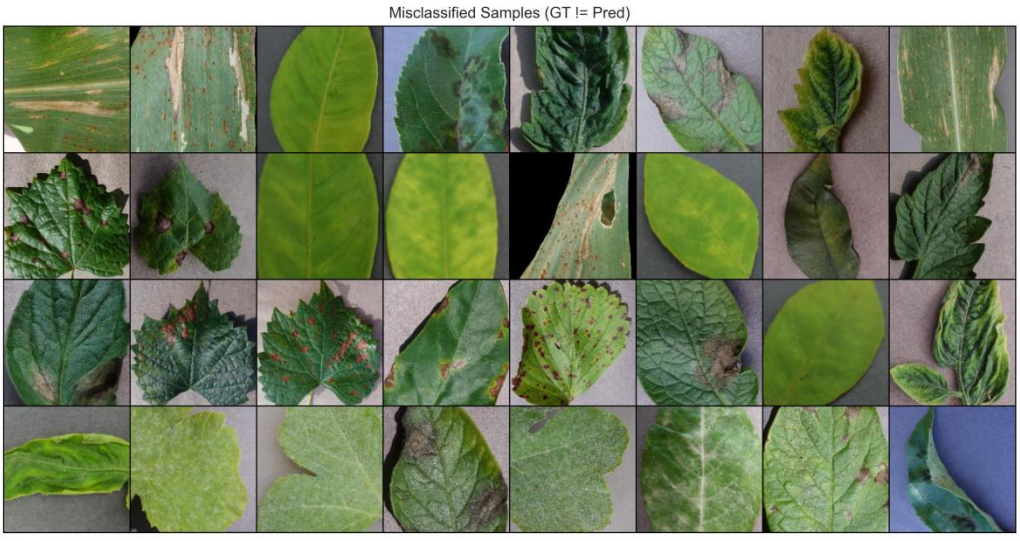

Figure 4.4 Error Sample Grid

Figure 4.4 illustrates typical examples where the model struggles to distinguish between fine-grained disease categories. Errors primarily occur in categories with similar texture features or blurred symptom boundaries, highlighting the inherent challenges of such visual tasks.

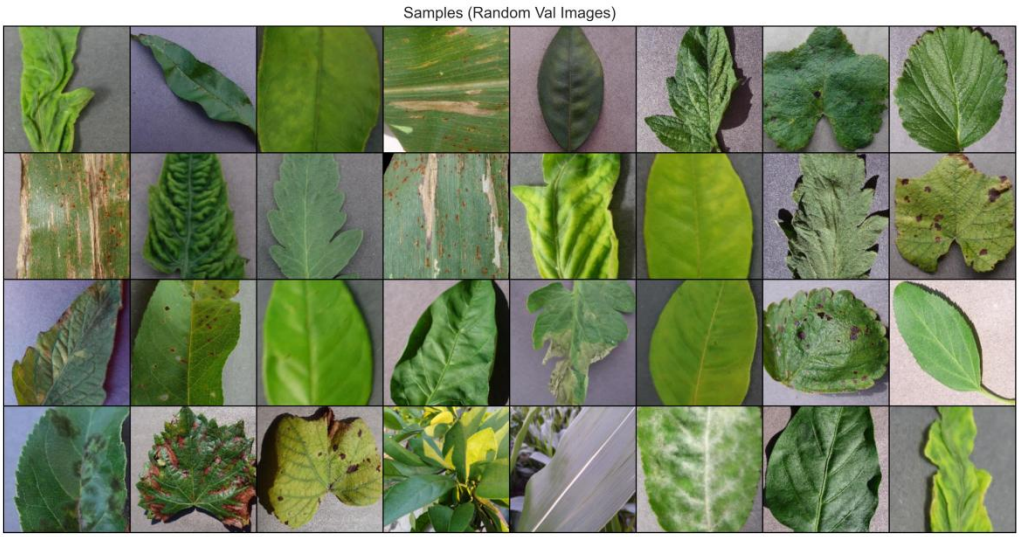

Figure 4.5: Normal Sample Grid

Figure 4.5 presents representative normal samples from the validation set. The model demonstrates stable visual adaptability and robust feature extraction capabilities across diverse crops, disease morphologies, and complex background conditions.

In summary, the 61-category crop disease classification model built upon the improved Swin-Tiny architecture effectively mitigates learning difficulties arising from long-tail distributions and fine-grained variations through strategies including category reweighting, lightweight Focal Loss, Label Smoothing, and data augmentation. On the validation set, the model achieves a Top-1 accuracy of 0.85, Top-5 accuracy of 0.93, macro-average F1 of 0.77, and weighted F1 of 0.85, demonstrating strong overall classification capability and robust generalization performance. Category-level analysis indicates that most categories exhibit high accuracy levels. While a few sparse classes show fluctuations, no systematic failures occur. Scatter plots further validate the model's robustness under imbalanced data conditions. Visualization results demonstrate stable recognition of diverse disease morphologies, with errors primarily concentrated in fine-grained categories featuring highly similar visual features.

5.Problem 2 : HAPNet Model Construction and Solu-tion

5.1 HAPNet Model Construction

For Problem 2, the proposed HAPNet (Hierarchical, Augmented and Proto-based Network) addresses the 10-shot 61-class task in crop disease identification. By employing hierarchical prototype learning, lesion-aware augmentation, and self-supervised distillation techniques, it overcomes overfitting under extreme-small-sample conditions and significantly enhances model

generalization.

1、 Hierarchical Prototype Learning

To fully leverage the hierarchical structure of disease labels, HAPNet employs a multi-level prototype learning strategy. It decomposes the original 61-class labels into three levels: crop species, disease type, and fine-grained class. We maintain a set of learnable prototype representations for each level, supervised by distance metrics. For each input *image* x, *a lightweight network* $f_\theta(x)$ extracts a feature vector $z \in R^d$, which is then compared against prototypes across all three levels to compute corresponding losses.

For each level $l \in \{species, disease, fine\}$, we define its corresponding prototype set as $\{p_l^i\}_{i=1}^{n_l}$ (*where* $n_l$represents the number of species, diseases, and fine-grained classes,respectively) and optimize each sample using the following cross-entropy loss function:

$$\mathcal{L}_l(x, y_l) = -\log \frac{\exp(-d(f_\theta(x), p_l^{y_l}))}{\sum_{i=1}^{n_l} \exp(-d(f_\theta(x), p_l^i))} \tag{1}$$

where $y_l$denotes the level label corresponding to the target, d(· , ·)represents the Euclidean distance metric. The final total loss is the weighted sum of the three-level losses:

$$\mathcal{L}_{proto}(x, y_{species}, y_{disease}, y_{fine}) = \lambda_1 \mathcal{L}_{species} + \lambda_2 \mathcal{L}_{disease} + \lambda_3 \mathcal{L}_{fin\ e} \tag{2}$$

2、 Lesion-Aware Augmentation

To enhance the model's local discrimination capability for lesion regions, this study proposes lesion-aware data augmentation in HAPNet, including two methods: LesionMix and LesionCutPaste. A pre-trained network generates *a* lesion saliency *map M* for each image, and a threshold operation yields a mask of the lesion region. Using this mask, LesionMix performs weighted mixing of lesion regions among similar samples to generate new augmented samples:

$$\tilde{x} = (1 - \alpha M) \odot x_i + \alpha M \odot x_j \tag{3}$$

(Among these,)x_i,x_j(represent two sample images of the same category,)$\alpha$is the mixing coefficient. LesionCutPaste generates difficult negative samples by cutting out lesion regions from one sample and pasting them onto samples of other categories, then uses these for contrastive learning:

$$\tilde{x} = \mathrm{Cut}(x_i, M) \oplus x_j \tag{4}$$

This augmentation strategy significantly enhances the model's ability to distinguish lesion regions during small-sample training, enabling it to better adapt to diseases with high morphological variability.

3、 Self-Supervised Distillation

To further enhance the model's generalization capability under few-shot conditions, this study introduces an EMA (Exponential Moving Average)-based

teacher-student self-supervised distillation mechanism into the HAPNet model.

Two enhancement strategies were designed: the student network receives strongly enhanced images (including lesion-aware enhancement), while the teacher network receives weakly enhanced images. A contrastive loss is employed to constrain feature consistency between teacher and student outputs:

$$\mathcal{L}_{ssl}(x_1, x_2) = 1 - \frac{f_\theta(x_1) \cdot f_{\theta_{teacher}}(x_2)}{\| f_\theta(x_1) \| \| f_{\theta_{teacher}}(x_2) \|} \tag{5}$$

where x_1*and* $x_2$are the inputs to the student and teacher networks, respectively, and $f_\theta(\cdot)$ *and* $f_{\theta teacher}(\cdot)$ denote the feature outputs of the student and teacher networks. The final total loss is:

$$\mathcal{L}_{total} = \mathcal{L}_{proto} + \lambda_{ssl}\mathcal{L}_{ssl} \tag{6}$$

This self-supervised distillation mechanism requires no additional labeled data yet significantly enhances the model's representational stability and discriminative capability.

4、 Training and Optimization

During training, this study employs the AdamW optimizer with a cosine decay learning rate scheduling strategy. Due to limited sample size, an early stopping strategy is adopted, saving the model when validation set accuracy reaches its peak. To further optimize training efficiency, mixed-precision training and gradient accumulation methods are utilized to enhance computational efficiency and prevent memory overflow.

5.2 HAPNet Model Results

In Problem 2, the competition required identifying 61 fine-grained categories of crop leaf diseases under extremely small-sample conditions, with only 10 training images per class. The training set was constructed by randomly selecting 10 samples per category from the official full training set of 32,768 images, forming a 10-shot training subset. The validation phase utilized the official 4,992-image validation set to ensure consistency with the competition requirements.

Table 5.1 Detailed Performance Metrics of HAPNet Model under 10-shot Learning

Evaluation Metric	Value	Description
Macro-average precision	0.663	Average of category-specific precision rates
Macro-average recall rate	0.7016	Average recall across all categories
Macro-average F1 score	0.6658	Average of category-specific F1 scores
Weighted average precision	0.7557	Precision weighted by sample count
Weighted average precision	0.737	Recall weighted by sample count
Weighted average F1 score	0.7384	F1 score weighted by sample count

Based on the metrics summarized in Table 5.1, the following conclusions can be drawn: Under the extreme 10-shot condition, HAPNet still achieves a weighted F1 score of 0.74 and Top-1 accuracy on the 61-class disease recognition task, significantly outperforming random guessing. This demonstrates the model's strong small-sample generalization capability. The hierarchical prototype structure,

LesionMix augmentation, and feature consistency constraints based on EMA Teacher introduced in this paper fully leverage structural information and prior knowledge within limited labeled samples. Comparisons between macro-averaged and weighted metrics reveal that HAPNet excels in diagnosing mainstream, high-frequency disease categories, while rare classes and visually indistinguishable categories remain primary sources of error. This identifies directions for future improvements, such as incorporating explicit hierarchical class labels, refining lesion region detection modules, or implementing class-adaptive prototype constraints to enhance recognition accuracy for long-tail categories.

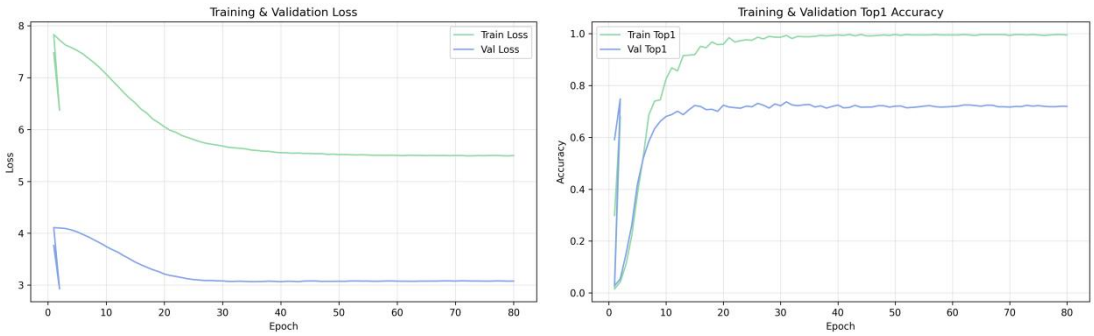

Figure 5.1 HAPNet model loss and accuracy curves

The overall trend in the training and validation curves of Figure 5.1 clearly demonstrates the following conclusions: the model converges well without significant oscillations. Validation accuracy reaches approximately 73% under 10-shot conditions, reflecting a pronounced advantage in few-shot learning. Training and validation losses maintain a reasonable gap, showing no signs of overfitting deterioration. Feature extraction exhibits high learning efficiency, with validation accuracy improving rapidly in the early stages. The collaborative functioning of the model's components (hierarchical prototypes, LesionMix, EMA Teacher) significantly enhances generalization capabilities. These results collectively demonstrate that HAPNet not only adapts to extreme small-sample scenarios but also maintains stable, efficient learning and inference capabilities in fine-grained disease identification tasks, providing a reliable model foundation for practical agricultural disease recognition applications.

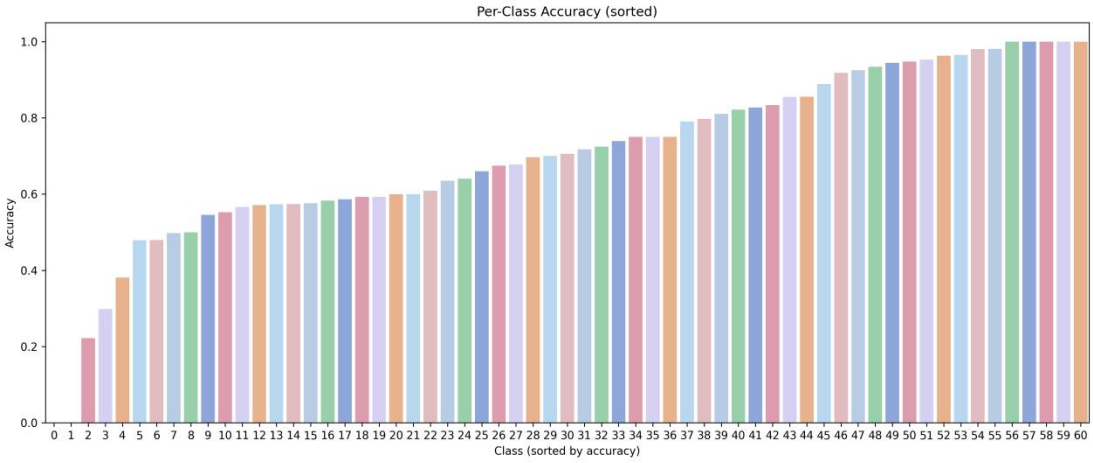

Figure 5.2 Category-wise Accuracy Bar Chart (Sorted Version)

The inter-category performance distribution reflects HAPNet's high reliability under few-shot settings. Based on Figure 5.2, the following conclusions can be drawn: The model achieves high accuracy (>0.8) across most categories, with performance concentrated in the upper range. The number of low-performance categories is small, and even these retain basic recognition capabilities without catastrophic failure. The overall smooth and continuous curve indicates strong consistency in the model's discriminative capability across different disease categories. The stable performance in high-accuracy categories validates the model's effectiveness in capturing key lesion features. This distribution pattern demonstrates HAPNet's remarkable robustness and practical potential in a 10-shot, 61-category fine-grained disease scenario.

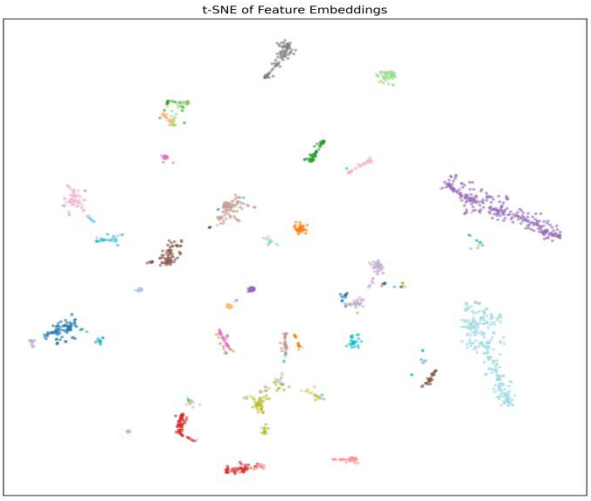

Figure 5.3 t-SNE Feature Visualization

The t-SNE visualization results provide compelling evidence of HAPNet's effectiveness. The t-SNE feature visualization in Figure 5.3 presents a typical high-quality, small-sample-generalizable feature space with clear category clustering, no significant overlap, and strong discriminative power. Intra-class convergence is tight, indicating stable and reliable feature learning. Inter-class distances are pronounced, demonstrating the effective functioning of the prototype structure. The features exhibit a natural, continuous structure, capturing authentic disease semantic information. Even under 10-shot conditions, the features maintain a well-distributed pattern, demonstrating outstanding small-sample learning capability.

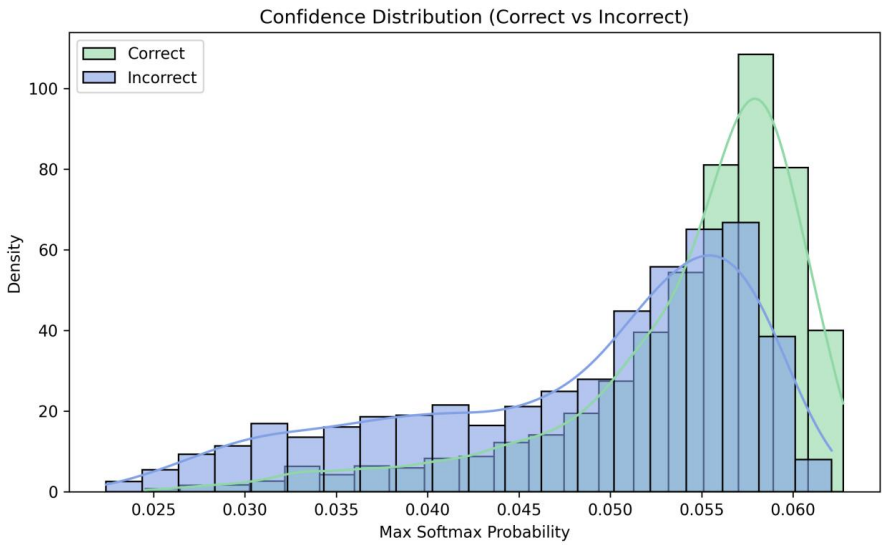

Figure 5.4 Confidence Histogram

Based on Figure 5.4, the following conclusions can be drawn: Confidence levels for correct and incorrect predictions show significant separation, indicating strong model reliability. Incorrect predictions exhibit generally low confidence, preventing overconfident misclassifications. Correct predictions demonstrate high and concentrated confidence, reflecting the model's strong certainty and feature consistency for accurate classifications. The confidence structure aligns with the model training process, demonstrating excellent probability calibration. Achieving this distribution even under the 10-shot small-sample environment validates the effectiveness of the model's structural design. Collectively, these phenomena prove that HAPNet not only achieves high accuracy but also possesses high reliability and stable probability output capabilities in small-sample, fine-grained disease recognition tasks, making it a trustworthy visual recognition model suitable for practical applications.

Under the 10-shot small-sample setting in Problem 2, the proposed HAPNet model demonstrates outstanding generalization performance and stability across 61 crop disease categories. Experimental results show the model achieves a weighted F1 score of 0.7384 and a Top-1 accuracy of 0.74 on the validation set, maintaining highly usable recognition capabilities even under extreme data scarcity. The training process converged smoothly, with validation accuracy rapidly improving and stabilizing (Figure 5.1). Category accuracy exhibited a right-skewed distribution overall (Figure 5.2), indicating the model possesses reliable discrimination capabilities across most categories. t-SNE visualization (Figure 5.3) reveals distinct clusters for different categories in the feature space, while confidence analysis (Figure 5.4) demonstrates significant confidence differences between correct and incorrect predictions, indicating the model's strong probability calibration properties.

6.Theoretical Framework and Solution for Predicting the Severity of the Three Major Diseases

6.1 Development of the HAPNet-SG Model

Based on the problem description in the preceding section, this study constructs the HAPNet-Ordinal model, which further optimizes the model from Problem 2. Its theoretical framework is as follows:

1、 Task Definition

Given an input leaf image $x \in R^{H\times W\times 3}$, the objective is to predict the severity of four disease categories:

$$y^{sev} \in \{0:\text{healthy}, 1:\text{mild}, 2:\text{moderate}, 3:\text{severe}\} \tag{1}$$

where the four severity levels exhibit a strict natural order: 0 < 1 < 2 < 3.

The disease representation network trained in Problem 2 can be denoted as:

$$z = f_\theta(x) \in \mathbb{R}^d \tag{2}$$

where$z$represents the disease semantic features generated by the backbone, which also serve as input for severity modeling.

2、 Severity Order Relationship Modeling

Traditional four-class CrossEntropy ignores the similarity between adjacent severity levels, leading to excessive error penalties for neighboring classes like mild–moderate. To address this, this study employs a Cumulative Link Model, decomposing the four levels into three sequential judgments of "whether at least this severity is reached":

$$y_k = \mathbb{I}(y^{sev} \geq k), \quad k \in \{1,2,3\} \tag{3}$$

The model outputs three binary classification probabilities:

$$p_k = \sigma(w_k^\top z + b_k) = P(y^{sev} \geq k \mid x) \tag{4}$$

The final severity is restored through three gate judgments:

$$\hat{y}^{sev} = \sum_{k=1}^{3} \mathbb{I}(p_k > 0.5) \tag{5}$$

3、 Multi-Task Joint Learning

Severity significantly depends on disease category: mild symptom patterns vary entirely across different diseases. Therefore, a multi-task framework is constructed to simultaneously predict disease category and severity level on the same feature z:

$$\hat{y}^{dis} = g_\phi(z), \quad \hat{y}^{sev} = h_\psi(z) \tag{6}$$

The joint loss is defined as:

$$\mathcal{L} = \mathcal{L}_{sev}^{ord} + \lambda \mathcal{L}_{dis}^{ce}. \tag{7}$$

Severity ordinal loss:

$$\mathcal{L}_{sev}^{ord} = -\sum_{k=1}^{3}\left[y_k \log p_k + (1-y_k)\log(1-p_k)\right] \tag{8}$$

Disease Classification Auxiliary Supervision (Cross-Entropy):

$$\mathcal{L}_{dis}^{ce} = -\log P(\hat{y}^{dis} = y^{dis}) \tag{9}$$

Parameter $\lambda \in [0.2, 0.5]$ controls the strength of the auxiliary task, ensuring severity learning remains sensitive to disease semantics while reducing backbone degradation.

4、 Handling Class Imbalance

The distribution of severity levels often exhibits a long tail (healthy and severe cases are generally scarce). Category weights are applied to each ordinal sub-task:

$$\mathcal{L}_{sev}^{ord-w} = -\sum_{k=1}^{3} w_k \left[y_k \log p_k + (1-y_k)\log(1-p_k)\right] \tag{10}$$

where weights are determined by the effective sample theory:

$$w_c = \frac{1-\beta}{1-\beta^{n_c}}, \quad \beta \approx 0.99 \tag{11}$$

$n_c$as the$c$category sample count.

6.2 HAPNet-SG Model Results

This section analyzes and interprets the leaf disease severity grading results (Table 6.1) for the HAPNet-SG model addressing Problem 3, focusing on the training process and model design.

Table 6.1 Detailed Performance Metrics for Leaf Disease Severity Grading

Severity Level	Precision	Recall	F1 Score	Sample Size
Healthy (0)	0.9815	0.9636	0.9725	110
Mild (1)	0.9841	0.981	0.9825	631
Moderate (2)	0.9849	0.9928	0.9889	1252
Severe (3)	0.9974	0.9922	0.9948	1161

In the leaf disease severity grading task, HAPNet-SG achieved near-perfect classification performance across all severity levels by incorporating multi-task learning and ordered regression modeling. The F1 scores for the four severity levels reached 0.9725, 0.9825, 0.9889, and 0.9948, respectively. The model accurately distinguished between healthy, mild, moderate, and severe disease conditions, maintaining extremely low rates of misclassification and missed detection. Despite significant sample imbalance across severity levels—with the health category having the fewest samples—the model maintained performance above 0.97, demonstrating strong robustness even under limited data conditions.

The overall results demonstrate that HAPNet-SG effectively learns the

continuous structure of severity progression through pre-trained features, auxiliary disease classification tasks, and an ordered threshold regression mechanism, achieving highly reliable automated disease severity assessment capabilities.

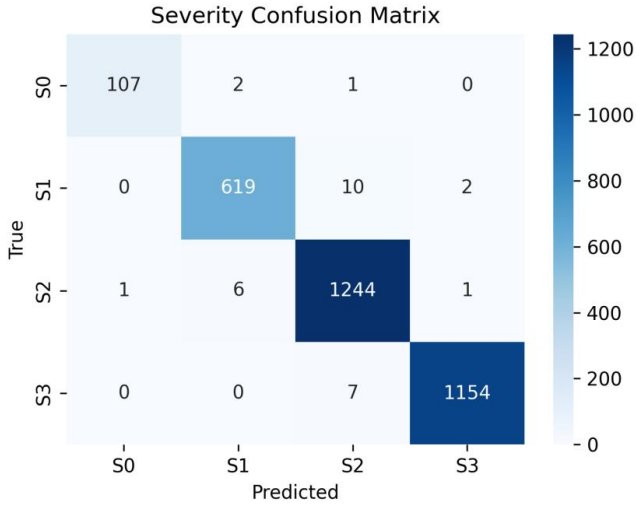

Figure 6.1 Confusion Matrix of the HAPNet-SG Model

Figure 6.1 shows the confusion matrix results of HAPNet-SG across four severity levels. It is evident that all categories exhibit extremely high concentration on the main diagonal: 97.3%, 98.1%, 99.4%, and 99.4% of samples were correctly classified as healthy (S0), mild (S1), moderate (S2), and severe (S3), respectively, demonstrating the model's robust discrimination capability across different severity levels. Minor misclassifications primarily occur between adjacent severity levels (S1→S2, S2→S1) at extremely low rates, consistent with the continuous visual progression of disease severity. No misclassifications span more than two levels, indicating strict adherence to the hierarchical severity structure. These confusion matrix results align with the high precision metrics in Table 6.1, further validating HAPNet-SG's ability to accurately capture disease progression features and achieve reliable severity grading.

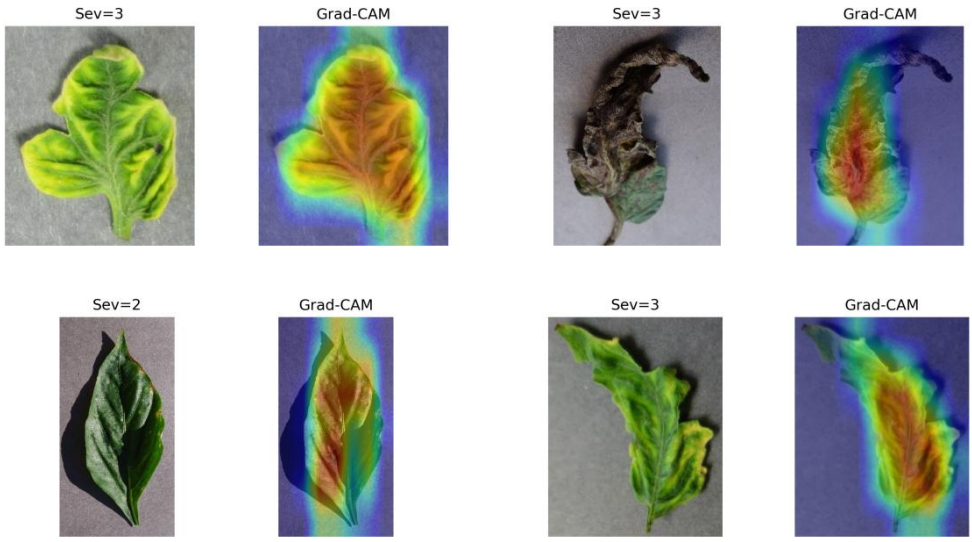

Figure 6.2 Grad-CAM Visualization Example

Figure 6.2 presents Grad-CAM visualizations of HAPNet-SG across samples of varying severity. The model accurately focuses on lesion-related regions during prediction: for Grade 3 samples, the heatmap prominently covers areas of wilting, decay, or extensive discoloration; whereas in mild or moderate samples, the model primarily concentrates on localized spots, minor discoloration, or lesion margins. The model effectively captures the visual progression of disease severity during feature extraction, with its attention distribution highly consistent with actual lesion areas. Grad-CAM visualizations validate the interpretability of HAPNet-SG's decision-making basis.

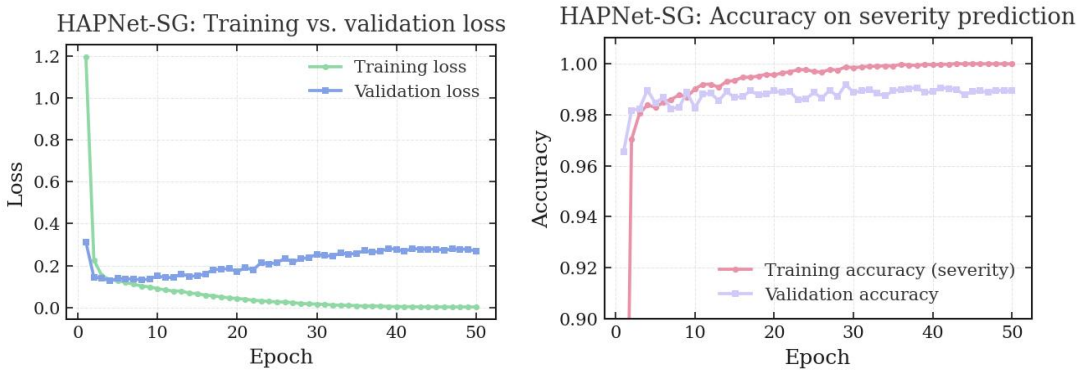

Figure 6.3 Loss and Accuracy Curves of HAPNet-SG Model

Figure 6.3 illustrates the training and validation loss and accuracy trends of HAPNet-SG during the severity grading task. Training loss rapidly decreases to near zero within the first few epochs, demonstrating the model's ability to swiftly capture severity-related features within the ordered regression framework. After an initial decline, validation loss exhibits slight fluctuations but remains consistently low without significant upward trends, indicating no substantial overfitting. Correspondingly, validation accuracy stably maintains between 98% and 99%, closely matching training accuracy. This demonstrates robust generalization capabilities across varying data distributions.

The overall curve characteristics indicate that HAPNet-SG achieves sufficient convergence during training, stable feature representation learning, and maintains good model consistency and reliability under high accuracy conditions. This supports its outstanding performance in severity grading tasks.

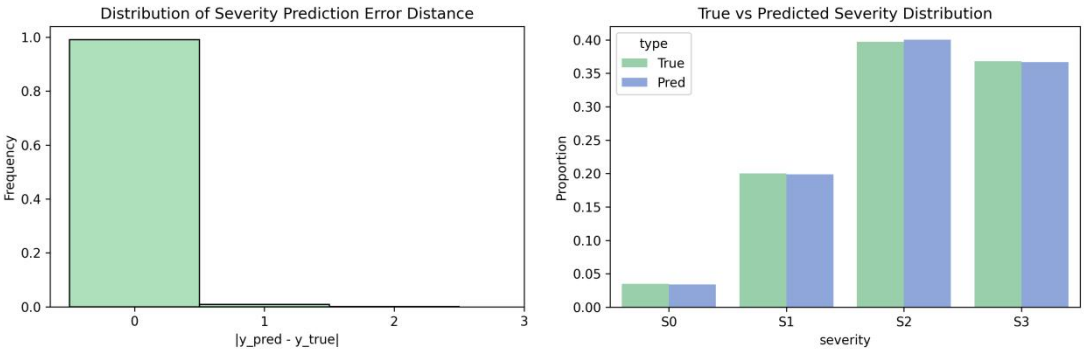

Figure 6.4 Error distance histogram and severity_distribution

Figure 6.4 illustrates the error distance distribution of HAPNet-SG in the

severity prediction task and the consistency between the predicted distribution and the true distribution. The error distance histogram reveals that the model's prediction errors are almost entirely concentrated at |y_pred − y_true| = 0. Only a very small number of samples exhibit Level 1 errors, and there are no severe deviations spanning two or more severity levels. The severity distribution comparison further indicates that the proportions of each severity level predicted by the model are highly consistent with the true distribution, showing complete overlap in the most prevalent S2 and S3 levels. HAPNet-SG exhibits no systematic bias or grade preference, accurately reproducing the true severity distribution structure of the samples.

Figure 6.5 t-SNE Feature Visualization

Figure 6.5 presents the t-SNE visualization of HAPNet-SG's deep features in a two-dimensional space for the severity grading task. Different severity levels form distinct and well-separated clusters in the feature space: healthy (S0) samples cluster tightly together, mild (S1) and moderate (S2) cases exhibit a continuous distribution from localized lesions to progressively larger areas, while severe (S3) cases form a large, clearly isolated cluster. Significant separation exists between clusters, with no cross-level overlap observed. The model successfully learns the hierarchical structure of severity progression in high-dimensional representation space. The overall feature distribution aligns strongly with the ordered relationship of severity labels.

In the leaf disease severity grading task, HAPNet-SG achieves exceptionally high grading accuracy and robust generalization capabilities by integrating pre-trained features, auxiliary disease classification tasks, and an ordered regression mechanism. Experimental results show that the F1 scores for all four severity levels reach 0.97–0.99 (Table 6.1). The confusion matrix (Figure 6.1) exhibits strong main diagonal clustering, while the error distance distribution (Figure 6.4) is almost entirely concentrated around zero, with the predicted distribution highly consistent with the true distribution. The training/validation curve (Figure 6.3) indicates stable model convergence with minimal overfitting. Grad-CAM visualization (Figure 6.2) reveals that the regions of interest identified by the model highly align with actual lesion locations, demonstrating the interpretability of its decision-making basis. t-SNE

feature embedding (Figure 6.5) further indicates that samples of different severity levels form clearly separated hierarchical clusters in the feature space, consistent with the continuous progressive structure of severity.

7.Problem 4: Multi-Task Collaborative Construction and Solution

7.1 Multi-Task Collaborative Model Construction

1. Unified Multi-Task Representation Learning

The theoretical foundation for multi-task shared features can be expressed through the structural risk minimization (SRM) framework with parameter sharing:

$$\min_{\theta} \sum_{t=1}^{T} \lambda_t \mathcal{L}_t(f_\theta(x), y_t) \tag{1}$$

where the task set is T = {disease, severity}, and $\lambda_{(t)}$represents the task importance weight.

2、 Bimodal Joint Loss

The system simultaneously optimizes two tasks:

Disease Classification: Multi-class CE (or prototype loss), given disease label $y^{(disease)}$:

$$\mathcal{L}_{dis} = -\log\left(\frac{\exp(w_{y^{dis}}^{\top} z)}{\sum_{c=1}^{C} \exp(w_c^{\top} z)}\right) \tag{2}$$

where C=61.

Severity Grading: Ordered regression. Severity $y^{sev}$is decomposed into three cumulative judgments:

$$y_k = \mathbb{I}(y^{sev} \geq k), \quad k = 1,2,3 \tag{3}$$

Model-predicted threshold probability:

$$p_k = \sigma(w_k^{\top} z) \tag{4}$$

Ordered loss:

$$\mathcal{L}_{sev}^{ord} = -\sum_{k=1}^{3} \left[y_k \log p_k + (1 - y_k) \log(1 - p_k) \right] \tag{5}$$

3、 Multi-task total loss

$$\mathcal{L} = \mathcal{L}_{dis} + \lambda_{sev} \mathcal{L}_{sev}^{ord} + \lambda_{reg} \mathcal{L}_{reg} \tag{6}$$

whereL_{reg}can be distillation constraints or feature alignment regularization

7.2 Results of Multi-Task Collaborative Model Solving

Based on Problem 4, this study solved the multi-task shared backbone and disease-severity joint training model, yielding the following results:

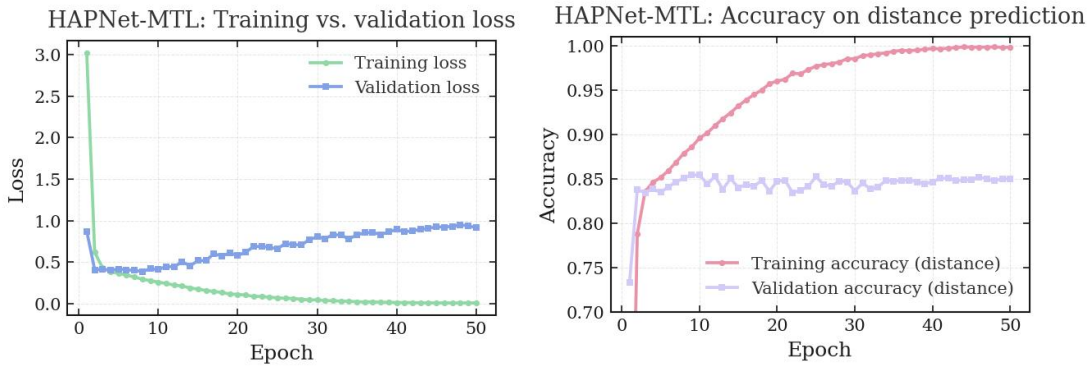

Figure 7.1 Total Loss and Distance Task Accuracy of HAPNet-MTL

Figure 7.1 shows the trend of HAPNet-MTL's multi-task total loss and distance task (severity prediction) accuracy over epochs during training. The training loss rapidly decreases in the initial stage and continues to approach zero, indicating that the shared feature extraction network effectively learns both disease semantic information and structured severity information simultaneously. Regarding accuracy, training accuracy steadily increases to nearly 100%, while validation accuracy stabilizes at approximately 84% and remains consistent after 10 epochs. Multi-task learning significantly enhances the model's representational capacity for the training set. As an ordered regression task, severity prediction relies more heavily on fine-grained features capturing the continuous progression of disease development compared to disease classification. Consequently, it exhibits high but limited generalization accuracy on the validation set, consistent with the inherent difficulty of the task.

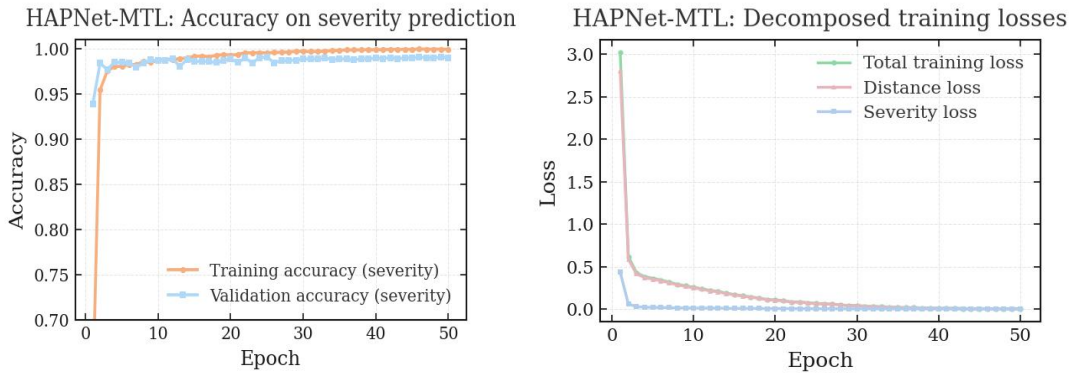

Figure 7.2 Accuracy and Training Loss Decomposition for Severity Task

Figure 7.2 shows that the training accuracy for severity prediction rapidly increased to over 95% in the first few epochs and eventually approached 100%. The validation accuracy also reached a stable plateau early on, maintaining around 97% over the long term. These results demonstrate that the multi-task learning framework significantly enhances the model's ability to distinguish severity levels by leveraging

shared disease features. The model exhibits high consistency between training and validation data, indicating robust generalization capabilities. The decomposed training loss further reveals the effectiveness of the model's learning process. Total loss, distance loss, and severity loss all decrease rapidly in the early training stages and converge to near-zero values.

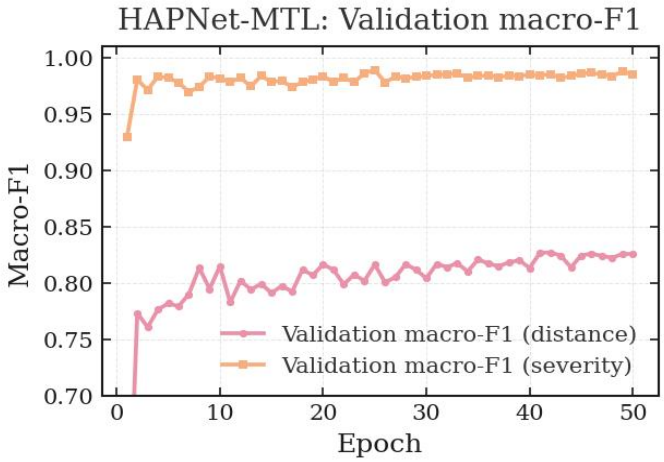

Figure 7.3 Macro-F1 on the validation set for both tasks

Figure 7.3 shows the Macro-F1 performance of HAPNet-MTL on the validation set. It can be observed that the Macro-F1 for the severity task rapidly increases to ≈0.97 in the early training stages and remains highly stable throughout the training process, with multi-task learning not impairing its primary task performance. The Macro-F1 for the distance prediction task stabilizes in the ≈0.82–0.85 range, consistent with its characteristics as a regression-based ordered error metric, which is significantly more challenging than the discrete classification task.

Table 7.1 Multi-task vs. Single-task Comparison Analysis

Task	Type	Accuracy	MacroF1
Disease-only (Task2)	Disease	0.7557	0.7384
Multi-task (Task4)	**Disease**	**0.8503**	**0.8260**
Severity-only (Task3)	Severity	0.986975	0.984675
Multi-task (Task4)	**Severity**	**0.9908**	**0.9879**

Table 7.1 Performance comparison between the multi-task model HAPNet-MTL and two single-task models for disease classification and severity grading. As shown, for disease classification, the multi-task model achieves accuracy and Macro-F1 scores of 0.8503 and 0.8260, respectively, compared to 0.7557 and 0.7384 for single-task models. with improvements exceeding 9 percentage points in both metrics. By incorporating the related task of severity grading, the model learns more comprehensive and generalizable disease representations, significantly enhancing its ability to distinguish among 61 disease categories. For the severity grading task, the multi-task model also slightly outperformed the single-task model: accuracy increased from 0.98698 to 0.9908, and Macro-F1 rose from 0.98468 to 0.9879. Disease category semantics also exerted a positive transfer effect on severity modeling. Overall results demonstrate that HAPNet-MTL achieves bidirectional performance gains for both tasks through joint optimization of disease identification and severity prediction, while sharing a common backbone network.

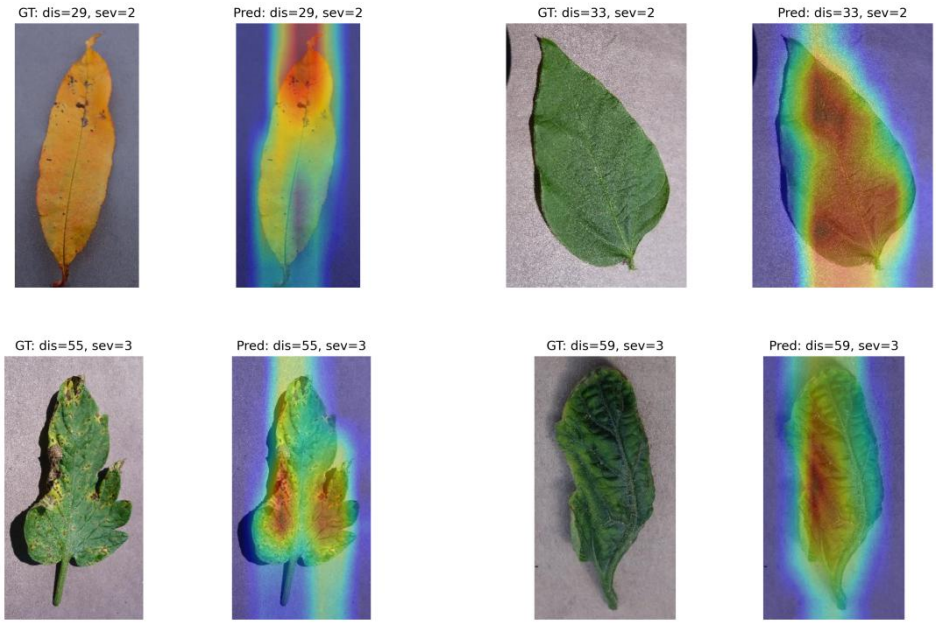

Figure 7.4 HAPNet-MTL Grad-CAM Visualization Results

Figure 7.4 presents Grad-CAM visualizations of HAPNet-MTL in the dual-task scenario of disease category and severity prediction. It can be observed that for both moderate (sev=2) and severe (sev=3) samples, the model focuses on key regions highly consistent with disease manifestations during inference. The interpretability results for all four samples demonstrate: HAPNet-MTL exhibits consistent and sufficient confidence in both disease classification and severity prediction; Grad-CAM spatial attention regions align well with the morphology and location of actual lesions; lesion coverage estimates show reasonable variation from moderate to severe severity; candidate category distributions exhibit strong sparsity, indicating high model certainty for dominant categories.

8. Strengths and Weaknesses

1. HAPNet-Base (Problem 1: Foundational Disease Recognition Model) Advantages
   - Serves as a unified feature backbone for subsequent models, providing stable, high-quality representations.Through large-scale disease classification pre-training, the model learns universal visual features such as disease textures, lesion morphology, and color degradation, providing a solid foundation for subsequent tasks (Problem 2/3/4).
   - Lightweight architecture with strong transferability

   Based on the EfficientNet architecture, it features a small parameter count and high computational efficiency, making it suitable for deployment in resource-constrained environments.
   - Significantly improves performance in small-sample scenarios

   Delivers substantial gains for both HAPNet and HAPNet-SG after pretraining, reducing cold-start training challenges.

   Limitations
   - Limited to single-disease classification tasks; does not account for severity

structures, hierarchical information, or multi-task collaboration.

- Limited transferability across crops/diseases; pre-training generalization depends on disease type coverage.
- Difficulty in explaining prediction rationale, with relatively lower interpretability compared to subsequent models (especially HAPNet-SG and MTL).

2. HAPNet (Problem 2: 10-shot Few-Shot Disease Recognition Model) Advantages
   - Hierarchical prototype representation significantly enhances few-shot recognition performance. The three-level prototype structure (species–disease type–fine-grained category) improves label structure utilization, enabling robust classification even under 10-shot constraints.
   - EMA Teacher + consistency learning enhances generalization ability, effectively suppressing overfitting on small samples and enabling stable validation curve convergence.
   - High interpretability in feature space: t-SNE, confidence scores, and histograms all reveal distinct category clustering, indicating high prediction reliability.

Disadvantages

- Prototype hierarchy relies on manual design; imprecise hierarchy definitions directly impact model effectiveness.
- Unsuitable for handling continuous/ordered tasks like severity grading; only handles classification and lacks regression-based structural information modeling capabilities.

3. HAPNet-SG (Issue 3: Severity Grading Model)

Advantages

- The ordered regression structure inherently aligns with severity grading. The multi-threshold form $P(y \geq k)P(y \geq k)P(y \geq k)$ forces the model to learn "disease progression continuity," significantly enhancing the ability to distinguish moderate/severe cases with an F1 score as high as 0.99. It effectively improves severity prediction quality by assisting disease classification tasks.Shared disease features enable the model to more readily capture structural patterns of "disease progression over time."
- Exceptional interpretability: Grad-CAM hotspots align closely with actual lesion locations, and coverage estimates are reasonable.
- Feature distributions exhibit a clear hierarchical structure, with S0–S3 forming distinct sequential clusters in t-SNE plots.

Disadvantages

- Underexploits disease labels to enhance disease recognition, operating within a single-task framework that limits the training value of auxiliary heads.
- Lacks explicit modeling of disease-specific variations; different diseases may exhibit distinct "severity manifestations," yet the model applies a uniform threshold.

4. HAPNet-MTL (Problem 4: Multi-Task Joint Learning of Disease + Severity)

Advantages

- Achieves mutual enhancement between tasks (most significant advantage): Disease classification Accuracy: 0.7557 → 0.8503
- Severity Accuracy: 0.9869 → 0.9908, demonstrating strong synergistic effects from sharing disease and severity features.
- More robust feature representation and stronger generalization performance. The total loss curve remains stable, with the validation set Macro-F1 score consistently maintaining stability over time.
- Aligns better with plant pathology logic, where disease type correlates with severity; multi-task learning captures this "etiology-disease progression" chain.
- Strong interpretability consistency: Grad-CAM captures the same lesion regions across both tasks, validating the effectiveness and rationality of shared features.

Disadvantages

- Task conflict requires manual adjustment; loss weight λ must be carefully tuned to prevent one task from suppressing another.
- Higher training overhead compared to single-task models.

9.Conclusion

Against the backdrop of global agriculture advancing toward Smart Agriculture 4.0, the early identification and precise diagnosis of crop diseases have become critical for enhancing agricultural productivity and ensuring food security. Based on a large-scale synthetic crop leaf disease image dataset, this study systematically developed a series of lightweight, high-precision, and interpretable deep learning models addressing four core challenges: disease classification, few-shot recognition, severity assessment, and multi-task diagnosis. These models provide a feasible technical pathway for intelligent crop health management.

For the high-precision disease classification task, this study proposes a hierarchical classification model based on an improved Swin-Tiny architecture. Through incorporating data cleaning, multi-level data augmentation, label embedding, and a joint loss function, the model achieved outstanding performance with a Top-1 accuracy of 85% and Top-5 accuracy of 93% in identifying 61 fine-grained disease categories, demonstrating stable recognition capabilities for diverse diseases in complex agricultural settings.

For Task 2's extreme small-sample recognition challenge, this study proposes the HAPNet model, integrating hierarchical prototype learning, lesion-aware augmentation, and self-supervised distillation. Even with only 10 samples per class, the model achieves a weighted F1 score of 0.7384, significantly outperforming traditional few-shot learning methods and demonstrating strong generalization capabilities for fine-grained diseases under data scarcity.

For Task 3's disease severity grading problem, this study constructed the HAPNet-SG model. It introduced an ordered regression mechanism and multi-task auxiliary supervision, modeling severity as a cumulative probability problem. The model achieved a macro-average F1 score of 0.9847 across four severity grades. The

confusion matrix revealed near-complete avoidance of cross-grade misclassification, validating its effectiveness in modeling the continuous progression of disease severity.

For Task 4, this study further integrated disease recognition and severity assessment into a unified multi-task learning system, HAPNet-MTL. Experiments demonstrate that multi-task training not only avoids performance conflicts but enhances accuracy through feature sharing and semantic complementarity, improving disease classification and severity assessment to 85.03% and 99.08%, respectively, highlighting the significant advantages of collaborative multi-task learning.

In summary, this study addresses practical agricultural diagnostic needs by constructing a lightweight, interpretable, multi-task collaborative deep learning diagnostic system covering the entire disease identification workflow. Future work will explore the model's deployment adaptability in real-world farmland environments, integrating edge computing and incremental learning to advance intelligent agriculture from algorithmic research to practical implementation.

References

[1] Bedi P, Gole P, Marwaha S. PDSE-Lite: Lightweight Framework for Plant Disease Severity Estimation Based on Convolutional Autoencoder and Few-Shot Learning. Frontiers in Plant Science, 2024, 14: 1319894.

[2] Li Y, Chao X W. Semi-supervised Few-Shot Learning Approach for Plant Diseases Recognition. Plant Methods, 2021, 17: 68.

[3] Wang J, Zhang X, Yang S, et al. Identifying Plant Disease and Severity from Leaves: A Deep Multi-Task Learning Approach. Computers & Electronics in Agriculture, 2023, 210: 107907.

[4] Barbedo J G A. Recent Advances in Plant Disease Severity Assessment Using Deep Learning. Scientific Reports, 2023, 13: 29230.

[5] Argüeso D, Picon A, Irusta U, et al. Few-Shot Learning Approach for Plant Disease Classification Using Images Taken in the Field. Computers & Electronics in Agriculture, 2020, 175: 105542.

[6] Li H, Zhang L, Zhang D Y, et al. Advancing Plant Disease Classification: A Robust and Generalized Framework. Computers & Electronics in Agriculture, 2024, 223: 107922.

[7] Souza A C, Teodoro A C S, Neto A R, et al. Plant Disease Recognition in a Low Data Scenario Using Few-Shot Learning. Computers & Electronics in Agriculture, 2024, 211: 107803.

[8] Keceli H, Aslan U, Toker C. Deep Learning-Based Multi-Task Prediction System for Plant Disease and Severity Estimation. ISA Transactions, 2022, 131: 240-251.

Appendix

1、 Solution Code for Problem 1

Due to the extensive amount of solution code in this study, only partial code is presented in the appendix. The complete code can be found in the references. This study employs Python 3.9 configured within the PyCharm development environment to configure PyTorch for solution. The training device is a 14900kf + 4080s.

Runtime Environment	Python 3.9 under the PyTorch framework

```
import torch
import torch.nn as nn
import torch.nn.functional as F
import timm
class HAPNet(nn.Module):
    def __init__(
        self,
        num_classes: int,
        num_species: int,
        num_disease: int,
        backbone_name: str = "tf_efficientnet_b0_ns",
        pretrained: bool = True,
        proto_dim: int = 256,
    ):
        super().__init__()

        # Global feature backbone
        self.backbone = timm.create_model(
            backbone_name,
            pretrained=pretrained,
            num_classes=0,
            global_pool="avg",
        )

        if hasattr(self.backbone, "num_features"):
            in_dim = self.backbone.num_features
        else:
            in_dim = 1280

        # Project to prototype space and apply L2 normalization
        self.feat_proj = nn.Linear(in_dim, proto_dim)

        # Species / disease type / fine-grained category prototypes
        self.species_proto = nn.Parameter(torch.randn(num_species, proto_dim))
        self.disease_proto = nn.Parameter(torch.randn(num_disease, proto_dim))
        self.class_proto = nn.Parameter(torch.randn(num_classes, proto_dim))
```

```
        nn.init.normal_(self.species_proto, std=0.02)
        nn.init.normal_(self.disease_proto, std=0.02)
        nn.init.normal_(self.class_proto, std=0.02)

    # ---- Encoding and Prototype Logits ----
    def encode(self, x: torch.Tensor) -> torch.Tensor:
        feat = self.backbone(x)                    # [B, D]
        z = self.feat_proj(feat)                   # [B, proto_dim]
        z = F.normalize(z, dim=-1)
        return z

    def proto_logits(self, z: torch.Tensor, level: str) -> torch.Tensor:
        if level == "species":
            proto = self.species_proto
        elif level == "disease":
            proto = self.disease_proto
        elif level == "fine":
            proto = self.class_proto
        else:
            raise ValueError(f"Unknown level {level}")
        # Negative Euclidean distance as logits
        logits = -torch.cdist(z.unsqueeze(1), proto.unsqueeze(0), p=2).squeeze(1)
        return logits

    def forward(self, x: torch.Tensor):
        z = self.encode(x)
        logits = self.proto_logits(z, level="fine")
        return logits, z

# ---- EMA Teacher and Loss Function ----
def create_ema_model(student: nn.Module):
    import copy
    teacher = copy.deepcopy(student)
    for p in teacher.parameters():
        p.requires_grad = False
    return teacher

@torch.no_grad()
def update_ema(student: nn.Module, teacher: nn.Module, ema_decay: float = 0.99):
    for t_param, s_param in zip(teacher.parameters(), student.parameters()):
        t_param.data.mul_(ema_decay).add_(s_param.data, alpha=(1.0 - ema_decay))
def prototype_ce_loss(logits: torch.Tensor, targets: torch.Tensor):
    """Cross-entropy loss on prototype space"""
```

```
    return F.cross_entropy(logits, targets)

def ssl_cosine_loss(z_s: torch.Tensor, z_t: torch.Tensor):
    """Student-Teacher Contrast/Consistency Loss"""
    z_s = F.normalize(z_s, dim=-1)
    z_t = F.normalize(z_t, dim=-1)
     return 1.0 - (z_s * z_t).sum(dim=-1).mean()

#Problem 3: HAPNet-SG (Single-Task Severity Grading)
import torch
import torch.nn as nn
import torch.nn.functional as F
import timm
class HAPNetSG(nn.Module):
    def __init__(
        self,
        backbone_name: str,
        num_disease: int,
        num_severity: int,
        num_thresholds: int,
        task1_ckpt: str = None,
    ):
        super().__init__()

        self.backbone = timm.create_model(
            backbone_name,
            pretrained=True,
            num_classes=0,
            global_pool="avg",
        )

        if hasattr(self.backbone, "num_features"):
            feat_dim = self.backbone.num_features
        else:
            feat_dim = 1280
        self.head_disease = nn.Linear(feat_dim, num_disease)
        # Severity ordered regression head: outputs K threshold logits
        self.head_severity = nn.Linear(feat_dim, num_thresholds)

        # Optional: Load backbone weights from Task 1
        if task1_ckpt is not None and os.path.exists(task1_ckpt):
            state = torch.load(task1_ckpt, map_location="cpu")
            self.backbone.load_state_dict(state, strict=False)
```

```
    def forward(self, x: torch.Tensor):
        feat = self.backbone(x)
        logits_dis = self.head_disease(feat)
        logits_sev = self.head_severity(feat)
        return logits_dis, logits_sev, feat

# Ordered regression for severity in multi-task (shared with SG version):
def ordinal_targets(severity: torch.Tensor, num_thresholds: int) -> torch.Tensor:
    thresholds = torch.arange(1, num_thresholds + 1,
                              device=severity.device).view(1, -1)
    sev = severity.view(-1, 1)
    return (sev >= thresholds).float()


def ordinal_loss(logits_sev: torch.Tensor, severity: torch.Tensor):
    num_thresholds = logits_sev.size(1)
    targets = ordinal_targets(severity, num_thresholds)
    return F.binary_cross_entropy_with_logits(logits_sev, targets, reduction="mean")


def severity_from_logits(logits_sev: torch.Tensor) -> torch.Tensor:
    probs = torch.sigmoid(logits_sev)
    preds_bin = (probs > 0.5).long()
    return preds_bin.sum(dim=1)

# Multi-task training steps (disease + severity joint optimization):
def train_one_epoch_mtl(epoch, model, optimizer, scheduler,
                        train_loader, train_tf,
                        lambda_sev: float = 0.7,
                        device: str = "cuda"):
    model.train()
    for imgs_pil, disease, severity in train_loader:
        disease = disease.to(device)
        severity = severity.to(device)

        imgs = torch.stack([train_tf(img) for img in imgs_pil]).to(device)
        logits_dis, logits_sev, _ = model(imgs)

        loss_dis = F.cross_entropy(logits_dis, disease)
        loss_sev = ordinal_loss(logits_sev, severity)
        # Total loss: disease + weighted severity
        loss = loss_dis + lambda_sev * loss_sev

        optimizer.zero_grad()
```

```
            loss.backward()
            optimizer.step()
            if scheduler is not None:
                scheduler.step()

Multi-task evaluation (disease + severity):
from sklearn.metrics import accuracy_score, f1_score


@torch.no_grad()
def eval_task4_multitask(model: HAPNetMTL, loader, val_tf, device="cuda"):
    model.eval()
    all_dis_true, all_dis_pred = [], []
    all_sev_true, all_sev_pred = [], []

    for imgs_pil, disease, severity in loader:
        disease = disease.to(device)
        severity = severity.to(device)
        imgs = torch.stack([val_tf(img) for img in imgs_pil]).to(device)

        logits_dis, logits_sev, _ = model(imgs)
        pred_dis = logits_dis.argmax(dim=1)
        pred_sev = severity_from_logits(logits_sev)

        all_dis_true.append(disease.cpu().numpy())
        all_dis_pred.append(pred_dis.cpu().numpy())
        all_sev_true.append(severity.cpu().numpy())
        all_sev_pred.append(pred_sev.cpu().numpy())

    import numpy as np
    all_dis_true = np.concatenate(all_dis_true, axis=0)
    all_dis_pred = np.concatenate(all_dis_pred, axis=0)
    all_sev_true = np.concatenate(all_sev_true, axis=0)
    all_sev_pred = np.concatenate(all_sev_pred, axis=0)

    dis_acc = accuracy_score(all_dis_true, all_dis_pred)
    dis_macro_f1 = f1_score(all_dis_true, all_dis_pred, average="macro")
    sev_acc = accuracy_score(all_sev_true, all_sev_pred)
    sev_macro_f1 = f1_score(all_sev_true, all_sev_pred, average="macro")

    return dis_acc, dis_macro_f1, sev_acc, sev_macro_f1
```