VOL. 11. NO. 3 FEBRUARY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480 /jitk.v11i3.7458

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

DATA AUGMENTATION EFFECTS ON PROTONET FEW-SHOT YELLOW DISEASE SEVERITY IN CHILI LEAVES

Rizal Amegia Saputra[1]*; Rusda Wajhillah[1]; Yusti Farlina[1]; Hani Noviani[1]; Saela Nurusysyifa[1]

Faculty of Engineering and Informatics[1]
Universitas Bina Sarana Informatika, Sukabumi, Indonesia[1]
http://bsi.ac.id[1]
rizal.rga@bsi.ac.id*, rusda.rwh@bsi.ac.id, yusti.yfa@bsi.ac.id, hani@gmail.com, saelanurus@gmail.com

(*) Corresponding Author
(Responsible for the Quality of Paper Content)

Abstract— *Yellow curling disease in chili plants is one of the leading causes of declining horticultural productivity because it reduces the quality and quantity of crops. Variations in symptoms at each level of severity make the identification process difficult, especially when labeled data is minimal. This study proposes a Prototypical Network-based Few-Shot Learning (FSL) approach with VGG16 architecture as a feature extractor. Five augmentation techniques, namely horizontal flip, rotation, zoom, brightness, and contrast adjustment, were used to increase data diversity in data-scarce conditions. Experiments were conducted with N-way K-shot configurations (2–5 classes; 1, 5, and 10 examples per class) to evaluate the impact of augmentation on prototype representation stability. Results show that increasing the number of examples per class consistently improves accuracy from 34.6% in 5-way 1-shot to 49.4% in 5-way 10-shot without augmentation. However, the use of augmentation decreases performance in higher N-way scenarios because it increases intra-class variability. The t-SNE visualization reinforces this study, where the healthy and severely diseased classes are clearly separated, while the intermediate class shows overlap. The novelty of this study is that it is the first to evaluate the impact of augmentation strategies on prototype representation stability in the agricultural domain with limited data. The results of this Few-Shot Learning approach are effective for plant disease classification despite the limited dataset*.

Keywords: *Few-Shot Learning, Prototypical Network, VGG16, Yellow Leaf Curl Disease*

Intisari— *Penyakit keriting kuning pada tanaman cabai merupakan salah satu penyebab utama penurunan produktivitas hortikultura karena menurunkan kualitas dan kuantitas hasil panen. Variasi gejala pada setiap tingkat keparahan menyebabkan proses identifikasi sulit dilakukan, terutama saat data berlabel sangat terbatas. Penelitian ini mengusulkan pendekatan Few-Shot Learning (FSL) berbasis Prototypical Network dengan arsitektur VGG16 sebagai ekstraktor fitur. Lima teknik augmentasi yaitu horizontal flip, rotation, zoom, brightness, dan contrast adjustment digunakan untuk meningkatkan diversity data dalam kondisi data-scarce. Eksperimen dilakukan dengan konfigurasi N-way K-shot (2–5 kelas; 1, 5, dan 10 contoh per kelas) untuk mengevaluasi dampak augmentasi terhadap stabilitas representasi prototipe. Hasil menunjukkan bahwa peningkatan jumlah contoh per kelas secara konsisten memperbaiki akurasi dari 34.6% pada 5-way 1-shot menjadi 49.4% pada 5-way 10-shot tanpa augmentasi. Namun penggunaan augmentasi menurunkan performa pada skenario N-way yang lebih tinggi karena meningkatkan intra-class variability. Visualisasi t-SNE memperkuat penelitian ini, di mana kelas sehat dan sangat parah terpisah jelas, sedangkan kelas menengah menunjukkan tumpang tindih. Kebaharuan pada penelitian ini menjadi pertama dalam mengevaluasi dampak strategi augmentasi terhadap stabilitas representasi prototipe dalam domain pertanian dengan keterbatasan data. Hasilnya dari pendekatan Few-Shot Learning ini, efektif untuk klasifikasi penyakit tanaman meskipun data terbatas dataset terbatas.*

Kata Kunci: *Pembelajaran Few-Shot, Jaringan Prototipikal, VGG16, Penyakit Keriting Daun Kuning*.

INTRODUCTION

Chilli (Capsicum annuum L.) is one of the primary horticultural commodities in Indonesia, with high economic value [1]. This plant plays an important role in the food sector and processing industries; therefore, its production's stability directly impacts food price inflation and national food security [2]. However, chilli productivity is highly vulnerable to diseases. One of the most damaging diseases is yellow leaf curl, which is caused by Begomovirus infection and is transmitted by the whitefly pest (Bemisia tabaci) [3]. This disease causes morphological changes, such as leaves rolling upward, stunting, and changes in leaf colour to bright yellow, with symptoms varying from mild to severe[4]. Severe infections can reduce crop yields by up to 70% if not appropriately handled [5].

Determining disease severity is important for early diagnosis and also plays a role in determining appropriate control strategies, such as the intensity of insecticide spraying or selective pruning [6][7]. However, the manual identification process by experts requires considerable time, effort, and cost [8]. Thus, this becomes the main challenge in developing digital image-based systems owing to the limited amount of labelled data, especially for classifying disease severity levels, which requires expert annotation [9].

Deep learning methods have been widely used for plant disease detection and have proven effective on large and uniform datasets, such as PlantVillage [10]. However, this method is data-hungry; therefore, its performance decreases significantly when applied to limited datasets [11].

Several previous studies have discussed the application of deep learning for detecting chilli leaf disease. Research [12] using ResNet50 achieved an accuracy of 99.6% on the PlantVillage dataset; however, the dataset used was homogeneous and focused on multiple diseases. The research [13] applied EfficientNetB0 to 1,600 field images and achieved an accuracy of 88%; however, this model still relies on large datasets and only classifies disease types.

The research [14] ested three pre-trained models (VGG16, InceptionV3, and EfficientNetB0) on 3,000 field images and achieved an accuracy of 98.8%; however, this approach still relies on conventional augmentation and has not yet assessed the differences between levels of symptom severity. In addition, [15] introduced the COLD dataset with more than 10,000 field images, an important benchmark source for chilli disease research. However, this descriptive study has not developed new methods for cases with limited data. Research [16] proposed Enhanced MobileNet for chilli leaf disease detection and achieved an accuracy of 97.18%; however, its focus remained on binary disease classification with general augmentation. The research [17] is one of the studies that has implemented a meta-learning approach (MAML, RMAML, and Reptile) for chilli variety identification and has shown an increase in accuracy of up to 84%. However, this study focused on variety identification, not disease severity classification, and has not yet integrated metric-based few-shot learning.

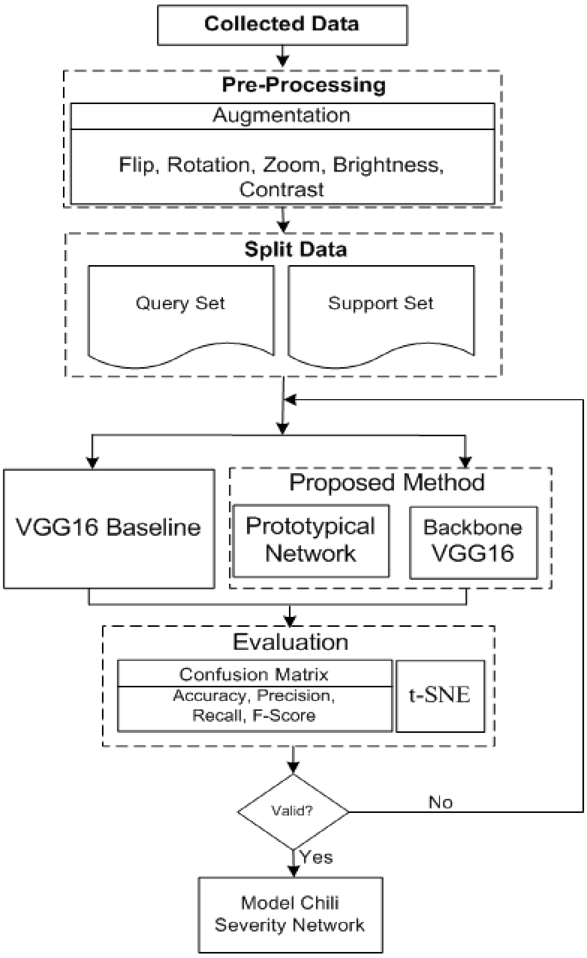
From various previous studies, most deep learning and transfer learning approaches still show limitations when dealing with limited data, visual similarities between levels of symptom severity, and the lack of scientific interpretation of the feature representations produced by the models. There has not yet been any research that analyses explicitly the impact of augmentation techniques on the performance of Few-Shot Learning (FSL) in the context of multi-level classification of chilli leaf disease severity.

This study proposes a Few-Shot Learning (FSL) approach based on a Prototypical Network with a VGG16 backbone for classifying yellow leaf curl disease severity in chilli plants. This approach was chosen because FSL is capable of performing effective classification even when the amount of training data is limited [18] by utilising representation-based learning that produces a prototype or feature centre for each class [19][20].

This approach combines symptom-based augmentation to enrich the data variation and improve the model's generalisation capability. Subsequently, a comprehensive evaluation was conducted using accuracy, precision, recall, and F1-score metrics, and t-SNE visualization for feature representation analysis. This study is expected to contribute significantly to developing efficient, interpretable, and adaptive plant disease diagnosis systems under limited data conditions.

MATERIALS AND METHODS

The model development process is carried out systematically, from data collection and augmentation, formation of support and query sets, and model training, to the performance evaluation stage. The following flowchart illustrates the overall process performed in this study.



Source: (Research Results, 2025)

Figure 1. Research Framework

1 Data Collection

The dataset is in the form of chilli leaf images with five categories of disease severity: healthy, mild, moderate, severe, and very severe. Data was acquired independently as well as from already published sources.

2 Pre-processing and Augmentation

This stage aims to improve the quality and diversity of the data, particularly for minor classes. Augmentation techniques were used during the training phase, and the augmentation process was performed automatically (on-the-fly) during training; therefore, the augmented images were not saved as a new dataset. The augmentation techniques include Flip, Rotation, Zoom, Brightness, and Contrast.

3 Split Data

The dataset was divided into two parts according to the few-shot learning paradigm: the support set as learning examples and the query set as test data in each training session.

4 *Few-Shot Learning* with *Prototypical Network*

The model was built using the Prototypical Network approach, which utilizes VGG16 as a feature extractor. This model calculates the distance between the query embedding and each class's prototype, then classifies it into the nearest class based on Euclidean distance.

5 Evaluation

In this study, the metrics used were confusion matrix, accuracy, precision, recall, and F1-score [21], as well as t-SNE visualization to understand the feature separability between classes.

6 Validation

If the evaluation results showed good performance, the model was confirmed as the final model. Conversely, if the performance is suboptimal, the previous stage is iterated.

Prototypical Network

In the Few-Shot Learning approach, a Prototypical Network model is designed to form a vector representation (prototype) of each class based on the available support set. For each class m, the prototype c_m is calculated as the average of the feature vector generated by the embedding function $f(x)$, over all examples in that class [22]:

$$c_m = \frac{1}{n}\sum_{x \in x_p} f(x) \tag{1}$$

Once the prototype representations are obtained, the model measures the Euclidean distance between the embedding representation of the query image x_q and each class prototype c_m to assess semantic closeness:

$$d(x_q, c_m) = \left\| f(x_q) - c_m \right\| 2 \tag{2}$$

Class prediction for each query image is performed by selecting the prototype that has the closest distance to the embedding representation of that image:

$$\hat{\mathcal{Y}}_q = argmin_m d(x_q, c_m) \tag{3}$$

To convert distances into a probabilistic distribution among classes, the model uses the softmax function, which calculates the likelihood of the query image belonging to each class:

$$p(\mathcal{Y} = m | x_q) = \frac{\exp(-d(f(x_q), c_m))}{\sum_m \exp(-d(f(x_q), c_m))} \tag{4}$$

During training, the model optimizes classification performance by minimizing the cross-entropy loss function, which evaluates how closely the predicted distribution matches the true class labels:

$$L_{CE} = -\sum_{m=1}^{K} \mathcal{Y}m \log(p(\mathcal{Y} = m|x_q)) \qquad (5)$$

Model VGG 16

The baseline model used for feature extraction in this research is VGG16, which has been adapted to classify Chilli leaf images into five classes of disease severity. The complete architecture structure is presented in Table 1. Input images sized 224 × 224 ×3 are processed through five stacked convolutional blocks, each consisting of two to three convolutional layers with 3×3 filter sizes and a stride of 1, followed by 2×2 max pooling with a stride of 2 to reduce spatial dimensions. The number of feature channels gradually increases from 64 to 512. After the feature extraction process, Global Average Pooling (GAP) is used to flatten the spatial features into a one-dimensional vector, followed by a dropout layer to reduce overfitting. Finally, a fully connected layer with five output neurons produces the final classification.

Table 1. Architecture of the Improved VGG16 Baseline

Layer	Feature Map	Size	Stride
Input	Image	224 × 224 × 3	-
1	2 × Conv	224 × 224 × 64	1
	MaxPooling	112 × 112 × 64	2
2	2 × Conv	112 × 112 × 128	1
	MaxPooling	56 × 56 × 128	2
3	3 × Conv	56 × 56 × 256	1
	MaxPooling	28 × 28 × 256	2
4	3 × Conv	28 × 28 × 512	1
	MaxPooling	14 × 14 × 512	2
5	3 × Conv	14 × 14 × 512	1
	MaxPooling	7 × 7 × 512	2
GAP	GlobalAveragePooling2D	512	-
Drop	Dropout	512	-
FC	Dence	5	-

Source: (Research Results, 2025)

Model Training Configuration

The experiment was conducted on Kaggle Notebook using an NVIDIA Tesla P100 GPU (16 GB VRAM) with Python 3.10, TensorFlow, and Scikit-learn. Images were resized to 224 × 224 pixels with a batch size 32 and a fixed random seed (42). Within the Few-Shot Learning framework, several N-way × K-shot configurations were evaluated (K = 1, 5, 10; query = 5 per class) over 500 training episodes using the Euclidean distance metric. To reduce computation, t-SNE visualization was applied only to the 5-way configuration (perplexity = 30, iterations = 1000).

RESULTS AND DISCUSSION

This section presents the experimental results and performance analysis of the Prototypical Network-based Few-Shot Learning (FSL) model with VGG16 on various N-way and K-shot configurations, as well as the impact of data augmentation.

Data Description and Experimental Scenario

The dataset comprised 250 chili leaf images divided into five classes: Healthy, Mild, Moderate, Severe, and Very Severe. It combined secondary data from Widi [23] with 55 primary samples collected through an inoculation process. The distribution was imbalanced, with only 35 images in the Moderate class and 70 in the Healthy class, supporting the relevance of the Few-Shot Learning (FSL) approach. In the FSL framework, data are organized into episodic tasks consisting of a support set (K-shot) for learning and a query set for evaluation rather than a conventional train–test split [24]. The following scenario design can be seen in Table 2.

Table 2. Experimental Data Scenario

N-Way	Class Definition	K-Shot	Augmentation
2-way	Healthy, Sick (combination of Mild+Moderate+Severe+Very Severe)	1,5, 10	With & without
3-way	Healthy, Moderate (Mild+Moderate), Severe (Severe+Very Severe)	1,5, 10	With & without
4-way	Healthy, Mild+Moderate, Severe, Very Severe	1,5, 10	With & without
5-way	Healthy, Mild, Moderate, Severe, Very Severe	1,5, 10	With & without

Source: (Research Results, 2025)

Experiment Results

The experimental results, which can be seen in Table 3, show that as the number of classes (N-way) increases, the model accuracy decreases, while increasing the number of examples per class (K-shot) gradually improves the stability of the prototype and increases accuracy. With 10-shot 2-way without augmentation, the accuracy is 88.6% and the F1-score is 0.88, while the lowest performance occurs in 5-way 1-shot with augmentation, which only reaches 32.24% and an F1-score of 0.31. This pattern shows that the use of augmentation does not always improve performance in multi-class scenarios and highlights the difference in complexity between healthy–sick binary classification and multi-level severity classification.

Table 3. Prototypical Network Performance with VGG16 on various N-way, K-shot, and augmentation configurations

N-way	K-shot	Augmentasi	Akurasi (%)	Precision	Recall	F1-score
2	1	No Aug	70.4	0.7171	0.704	0.685
2	1	With Aug	64.6	0.6648	0.646	0.6178
2	5	No Aug	81.8	0.8468	0.818	0.8116
2	5	With Aug	79	0.8230	0.79	0.7831
2	10	No Aug	88.6	0.9135	0.886	0.8803
2	10	With Aug	84.2	0.8782	0.842	0.8343
3	1	No Aug	45.6	0.4898	0.4547	0.4381
3	1	With Aug	49.6	0.5215	0.4960	0.4702
3	5	No Aug	56.9	0.6254	0.5693	0.5666
3	5	With Aug	58.7	0.6464	0.5867	0.5801
3	10	No Aug	67.6	0.7287	0.6751	0.6729
3	10	With Aug	63.6	0.6889	0.6351	0.6342
4	1	No Aug	34.8	0.4034	0.348	0.3355
4	1	With Aug	34.9	0.3746	0.349	0.3209
4	5	No Aug	50.4	0.5375	0.504	0.4954
4	5	With Aug	49.5	0.5307	0.495	0.4830
4	10	No Aug	55.5	0.6035	0.555	0.5521
4	10	With Aug	56.9	0.6238	0.569	0.5694
5	1	No Aug	34.6	0.3865	0.3456	0.3302
5	1	With Aug	32.2	0.3702	0.3224	0.3111
5	5	No Aug	45.6	0.4911	0.456	0.4498
5	5	With Aug	40.2	0.4385	0.4024	0.3948
5	10	No Aug	49.4	0.5281	0.4944	0.4866
5	10	With Aug	45.5	0.4965	0.4552	0.4479

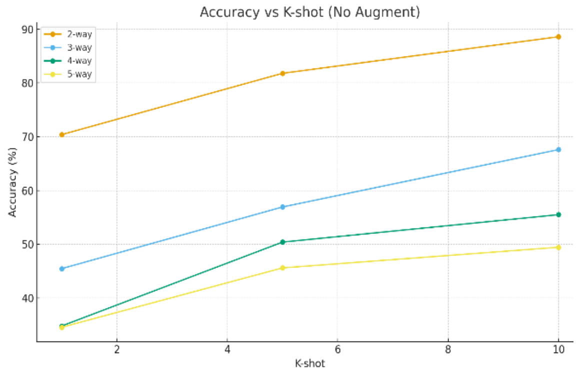
Source: (Research Results, 2025)

Analysis of the Influence of the Number of Classes (N-way)

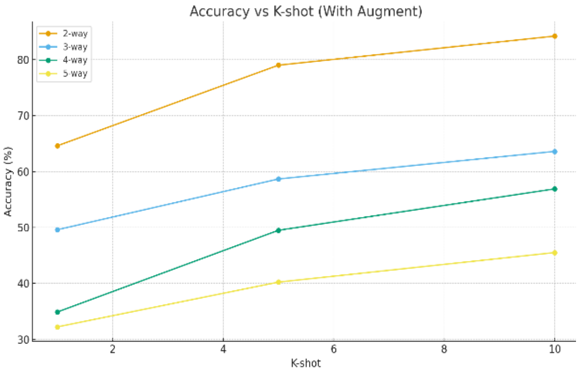
An increase in the number of classes (N-way) directly increases the complexity of the decision space, because the model must distinguish between more categories that have visual similarities, especially at moderate severity levels, such as Mild, Moderate, and Severe. A larger number of classes (N-way) increases the complexity of the decision space. In such circumstances, the distance between classes becomes smaller and the distribution of features between them tends to overlap. Synthetic variations such as changes in brightness, contrast, rotation, or zoom often increase intra-class variability faster than they increase separability between classes. As a result, images generated by augmentation can shift closer to the distribution of other classes, causing feature stacking, especially in 4-way and 5-way configurations. This condition makes the prototypes generated by the prototypical network less stable and makes it more difficult to describe the unique characteristics of each class. Therefore, lower accuracy at higher N-ways is caused by an increase in the number of classes that must be separated. Moreover, the biological characteristics of chili leaf disease symptoms do not entirely align with the model's sensitivity to generic augmentation.

Analysis of the Effect of Sample Size (K-shot)

The number of examples in each class (K-shot) has a direct effect on prototype stability in few-shot learning. At low K-shots, prototype representations are highly sensitive to noise and intra-class variation, causing the model to produce inconsistent decisions, especially for classes with overlapping intermediate features. Figures 2 and 3 show that increasing the number of examples from 1-shot to 5-shot significantly strengthens feature concentration, resulting in consistent accuracy improvements across all N-way configurations. However, the increase from 5-shot to 10-shot shows a pattern of diminishing returns, indicating that once the prototype is stable, additional examples only provide marginal improvements. The difference between the two graphs serves to clarify the impact of augmentation. In the no-augmentation condition, increasing K-shots results in a significant improvement in accuracy, while in the augmentation condition, the overall accuracy tends to be lower in N ≥ 3 configurations. This difference occurs because generic augmentation expands intra-class variability more than it increases between-class separability, so that the stability improvement effect of larger K-shots is partially offset by representation shift.

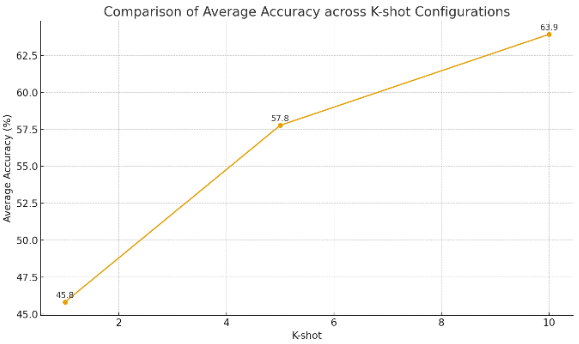


Source: (Research Results, 2025)
Figure 2. Accuracy vs. K-shot graph for various N-way configurations without augmentation



Source: (Research Results, 2025)
Figure 3. Accuracy vs. K-shot graph for various N-way configurations with augmentation

Figure 4 shows that although performance improved from 45.9% (1-shot) to 63.9% (10-shot), the improvement slowed down when the prototype had reached sufficient consistency. This condition emphasizes that the number of examples plays an important role in forming a stable class representation, while augmentation needs to be designed to be more adaptive so as not to disrupt the feature space structure that has been formed.



Source: (Research Results, 2025)
Figure 4. Comparison of Average Accuracy

Comparison of accuracy and time with the Baseline

The comparison in Table 4 shows a striking contrast between the performance of the VGG16 model as a baseline and the Prototypical Network approach in a 5-way classification scenario. When the amount of training data is very limited, VGG16 can only achieve an accuracy of 16.6% with an F1-score of 0.0571. This low result may occur because VGG16 requires many examples to learn large internal parameters, so the model overfits and fails to form general feature representations. The situation is different with the prototypical network.

This model shows a stable performance improvement as the number of examples per class increases. In the 1-shot condition, its accuracy was 34.6%, then rose to 45.6% in 5-shot, and reached 49.4% in 10-shot. This improvement demonstrates the nature of metric-based learning (Prototypical Network), which relies on comparing distances between classes rather than learning complex classification parameters.

Thus, the prototypical network is much better at dealing with data limitations and is more adaptable to variations in disease severity, which can sometimes be very subtle. Prototypical Network also has advantages in terms of efficiency because its inference process only requires distance calculations to the prototype of each class, which saves significant computing time compared to VGG16, which must perform full propagation through the classification layer. These results show that the few-shot learning approach is not only more computationally efficient than conventional CNNs but also more accurate in agricultural domains with limited data.

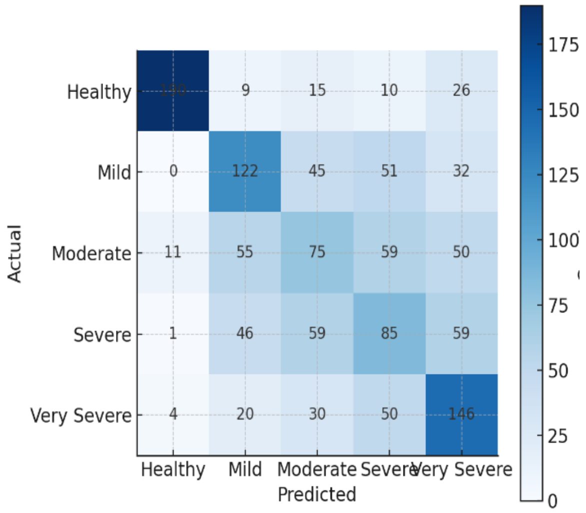
Table 4 Comparison of Baseline VGG16 and Prototypical Network (5-way Few-Shot Classification)

Model	Accuracy	Precision	Recall	F1-score
VGG16 (Baseline, Full-data, 5-class)	16.6	0.0333	0.2	0.0571
PrototypicalNet (5-way 1-shot, No Aug)	34.6	0.3865	0.3456	0.3302
PrototypicalNet (5-way 5-shot, No Aug)	45.6	0.4911	0.4560	0.4498
PrototypicalNet (5-way 10-shot, No Aug)	49.4	0.5281	0.4944	0.4866

Source: (Research Results, 2025)

Confusion Matrix Analysis

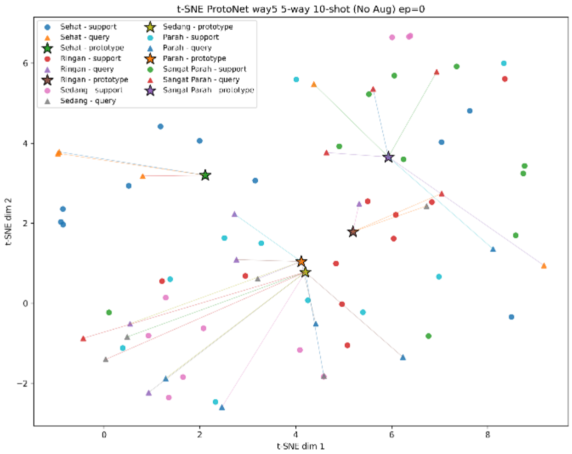
The confusion matrix in Figure 5 shows the classification results for the 5-way 10-shot configuration without augmentation. It indicates that the model is most accurate in recognizing the Healthy and Very Severe classes, which each have the most visible visual characteristics and can therefore be distinguished. Conversely, the Mild, Moderate, and Severe classes overlap, as indicated by the high number of cross predictions among the three classes. Due to similar color and texture variations, this pattern shows that the model still has difficulty distinguishing between closely related severity levels. As a result, the boundaries between classes in the feature space are not yet completely separated.



Source: (Research Results, 2025)
Figure 5. Confusion Matrix of the 5-Way 10-Shot

Feature Space Visualization

The t-SNE visualization in Figure 6 with a 5-way configuration and 10-shot without augmentation shows how the feature representations of each class are distributed in the latent space. The star symbols indicate that the prototypes are quite close to the center of their respective clusters, suggesting that a larger number of examples per class helps stabilize their formation. The model can distinguish between the Healthy and Very Severe classes because they have clear cluster separation. On the other hand, the Mild, Moderate, and Severe classes appear to be interrelated, and at some points, they merge into other classes. This pattern shows that these three severity levels have very similar visual characteristics. Overall, this t-SNE visualization shows that, although the prototypes have been formed stably, the separation of intermediate class features still requires a more contextual feature representation strategy.



Source: Research Results (2025)
Figure 6. t-SNE Plot of the 5-Way 10-Shot Configuration (No Augmentation)

CONCLUSION

This study indicates that prototypical network-based few-shot learning with VGG16 as a feature extractor is an effective approach for classifying the severity of yellow curl disease in chili leaves under limited data conditions. In general, increasing the number of examples per class improves the representation structure and model accuracy, while increasing the number of classes increases the risk of feature overlap due to symptom similarities between categories. In the best configuration, the model achieved 88.6% accuracy (2-way 10-shot), while the lowest performance occurred at 34.6% (5-way 1-shot). However, this study also found that the effectiveness of augmentation is not universal. In the 3-way 1-shot configuration, the augmentation technique improved performance by +4.0%, but decreased it in the 3-way 10-shot scenario by –4.0%.

This analysis found that synthetic variations that do not match the visual characteristics of the disease can increase intra-class variability and cause feature confusion, especially when the number of classes is large. t-SNE visualization and prototype distance analysis show that the healthy and severely diseased classes have the most stable cluster separation, while the moderate class tends to overlap. These findings provide evidence that prototype stability is strongly influenced by the semantic relevance of augmentation and symptom heterogeneity. Overall, this study makes an important contribution as the first to empirically evaluate the impact of augmentation strategies on prototype representation stability in few-shot learning for limited-data agricultural domains. The results also suggest new areas for research, such as creating distribution-based adaptive augmentation, using stronger backbones, and adding attention mechanisms to make features more distinct.

REFERENCE

[1] Apriyangsa Gulo, Listari Harefa, and Fitri Julianti Gea, “Efektivitas Penggunaan Mulsa Plastik dalam Meningkatkan Produktivitas Tanaman Cabai (Capsicum annuum L.),” *Flora J. Kaji. Ilmu Pertan. dan Perkeb.*, vol. 2, no. 1, pp. 152–159, Jan. 2025, doi: 10.62951/flora.v2i1.258.

[2] E. P. Yudha and G. C. Vanessa, “Analisis Kinerja Ekspor Cabai Hijau Di Indonesia,” *J. Apresiasi Ekon.*, vol. 10, no. 3, pp. 340–345, 2022, doi: 10.31846/jae.v10i3.506.

[3] R. Albar, M. Chatri, M. Des, D. H. Putri, and Y. B. Paradisa, “Penyakit Geminivirus (PepYLCV) pada Tanaman Cabai (Capsicum sp .) yang Disebabkan oleh Hama Kutu Kebul (Bemisia tabaci),” *J. Serambi Biol.*, vol. 8, no. 3, pp. 391–396, 2023.

[4] S. Suprayogi, N. Widyawati, and M. M. Herawati, “The Impact of Yellow Leaf Curl Disease Stage During Vegetative and Generative Phases on The Growth and Yield of Curly Red Chili Pepper OR Twist 42,” *J. Tek. Pertan. Lampung*, vol. 14, no. 2, pp. 483–493, 2025.

[5] N. Hardina, “Identifikasi Virus Penyebab Penyakit Kuning Keriting pada Cabai di Kabupaten Gowa , Sulawesi Selatan Identification of Virus that Causes Yellow Leaf Curl Disease in Chili in,” *J. Agrik.*, vol. 34, no. 3, pp. 427–434, 2023, [Online]. Available: https://jurnal.unpad.ac.id/agrikultura/article/viewFile/42833/21773.

[6] K. Ye, “Key Intelligent Pesticide Prescription Spraying Technologies for the Control of Pests , Diseases , and Weeds : A Review,” *Agric.*, vol. 15, no. 81, 2025, doi: https://doi.org/10.3390/agriculture15010081.

[7] S. Verma, A. Chug, A. P. Singh, and D. Singh, “Plant Disease Detection and Severity Assessment Using Image Processing and Deep Learning Techniques,” *SN Comput. Sci.*, vol. 5, no. 1, p. 83, Dec. 2023, doi: 10.1007/s42979-023-02417-5.

[8] S. Neto *et al.*, “Machine learning and deep learning techniques to support clinical diagnosis of arboviral diseases : A systematic review,” *PLoS Negl. Trop. Dis.*, vol. 16, no. 1, pp. 1–37, 2022, doi: https://doi.org/10.1371/journal.pntd.0010061.

[9] K. Singh and D. Malhotra, “Meta-Health: Learning-to-Learn (Meta-learning) as a Next Generation of Deep Learning Exploring Healthcare Challenges and Solutions for Rare Disorders: A Systematic Analysis,” *Arch. Comput. Methods Eng.*, vol. 30, no. 7, pp. 4081–4112, 2023, doi: 10.1007/s11831-023-09927-8.

[10] F. A. P. R. Putri, J. P. Tanjung, and N. P. Dharshini, “Implementation of Transfer Learning on CNN using DenseNet121 and ResNet50 for Brain Tumor Classification,” *J. Informatics Telecommun. Eng.*, vol. 7, no. 1, pp. 102–111, 2023, [Online]. Available: https://www.researchgate.net/publication/335117624_Malang_City_Polytechnic_Web_Based_Student_Attendance_Information_System_Telecommunications_Engineering_Study_Program_Using_Fingerprint/fulltext/5d515fe34585153e594ef214/Malang-City-Polytechnic-Web-Based-S.

[11] I. H. Rather, S. Kumar, and A. H. Gandomi, *Breaking the data barrier: a review of deep learning techniques for democratizing AI with small datasets*, vol. 57, no. 9. Springer Netherlands, 2024.

[12] S. Karthik, P. Revathy, M. S. Kavitha, and R. Sabitha, “Chili Crop Leaf Disease Detection Using Deep Feature Extraction Using Transfer Learning with ResNet50,” *2024 Asian Conf. Intell. Technol. ACOIT 2024*, pp. 1–4, 2024, doi: 10.1109/ACOIT62457.2024.10939738.

[13] V. K. Velpula, S. V. S. Prasad, J. Vadlamudi, S. Gurrala, A. V. Priya, and K. Srinidhi, “Chilli Leaf Disease Prediction System using Deep Learning Model,” *2025 Int. Conf. Emerg. Smart Comput. Informatics, ESCI 2025*, pp. 1–6, 2025, doi: 10.1109/ESCI63694.2025.10988404.

[14] S. Rozlan and M. Hanafi, “Efficacy of chili plant diseases classification using deep learning: A preliminary study,” *Indones. J. Electr. Eng. Comput. Sci.*, vol. 25, no. 3, pp. 1442–1449, 2022, doi: 10.11591/ijeecs.v25.i3.pp1442-1449.

[15] M. P. Aishwarya and A. P. Reddy, “Dataset of chilli and onion plant leaf images for classification and detection,” *Data in Brief*, vol. 54. 2024, doi: 10.1016/j.dib.2024.110524.

[16] S. A. Hamim and A. I. Jony, “Enhanced deep learning model architecture for plant disease detection in Chilli plants,” *J. Edge*

Comput., vol. 3, no. 2, pp. 136–146, 2024, doi: 10.55056/jec.758.

[17] W. Suwarningsih *et al.*, “Emphasizing Data Quality for the Identification of Chili Varieties in the Context of Smart Agriculture,” *Interdiscip. J. Information, Knowledge, Manag.*, vol. 19, no. 1, pp. 292–304, 2024, doi: 10.28945/5257.

[18] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, “Generalizing from a Few Examples,” *ACM Comput. Surv.*, vol. 53, no. 3, pp. 1–34, May 2021, doi: 10.1145/3386252.

[19] M. Fazaeli-Javan, R. Monsefi, and K. Ghiasi-Shirazi, “Hard-Positive Prototypical Networks for Few-shot Classification,” *IEEE Access*, vol. 13, no. February, pp. 41054–41067, 2025, doi: 10.1109/ACCESS.2025.3544638.

[20] W. Fu, L. Zhou, and J. Chen, “Bidirectional Matching Prototypical Network for Few-Shot Image Classification,” *IEEE Signal Process. Lett.*, vol. 29, pp. 982–986, 2022, doi: 10.1109/LSP.2022.3152686.

[21] A. Rachmad, M. Fuad, and E. M. S. Rochman, “Convolutional Neural Network-Based Classification Model of Corn Leaf Disease,” *Math. Model. Eng. Probl.*, vol. 10, no. 2, pp. 530–536, 2023, doi: 10.18280/mmep.100220.

[22] Y. Wu *et al.*, “Improved prototypical network for active few-shot learning,” *Pattern Recognit. Lett.*, vol. 172, pp. 188–194, 2023, doi: 10.1016/j.patrec.2023.06.015.

[23] W. Naftani, “Response of Several Chili Accessions and Molecular Markers to Yellow Leaf Curl Disease,” IPB University, 2023.

[24] M. R. Zarei and M. Komeili, “Interpretable few-shot learning with online attribute selection,” *Neurocomputing*, vol. 614, p. 128755, Jan. 2025, doi: 10.1016/j.neucom.2024.128755.