BloomLens: A Few-Shot Learning Framework for Fine-Grained Flower Classification with Prototypical Networks

Gong Zerui

Abstract Fine-grained flower classification from limited samples remains challenging due to subtle inter-species variations and substantial intra-species diversity. We present BloomLens, a few-shot learning framework that enhances prototypical networks through three key innovations: (1) transformer-based feature adaptation with multi-head self-attention for dynamic prototype refinement, (2) progressive training that scales from 5-way to 20-way tasks while maintaining performance, and (3) integrated augmentation combining MixUp and CutMix for fine-grained feature preservation. Evaluated on the Oxford Flowers-102, BloomLens achieves 93.64% accuracy on 5-way 1-shot tasks and 85.51% on 20-way 1-shot tasks, a 53.54% improvement over ImageNet baselines. BloomLens demonstrates exceptional scalability to 40-way classification (78.29% accuracy) while maintaining clear prototype separation, establishing a new benchmark for few-shot flower classification. Code available at `https://github.com/Ry3nG/BloomLens/`.

Introduction

Fine-grained visual classification of flowers presents a significant challenge in computer vision, particularly for recognizing previously unseen categories from limited examples. While recent deep learning advances have achieved impressive results on the Oxford Flowers-102 dataset (>99% accuracy) in fully supervised scenarios, real-world applications often lack extensive labeled datasets, especially for rare species or remote locations.

Few-shot learning (FSL) approaches including metric learning, meta-learning, and transfer learning methods offer promising solutions for learning from limited data. However, few-shot flower classification remains challenging, with only limited prior work. The most notable attempt by Derakhshani et al.[1] achieved 70.40% accuracy on 5-way flower classification tasks using Bayesian prompt learning. To better understand these challenges, we conducted extensive baseline experiments across

1. Derakhshani et al. 2023.

multiple architectures. Our results show severe limitations: modern CNN architectures like DenseNet-201 struggle, achieving only 58.52% ± 2.36% accuracy on 5-way 1-shot tasks. Performance degrades further on more challenging scenarios, dropping to 31.97% ± 1.20% for 20-way 1-shot tasks. This reveals three key challenges:

- **Fine-grained Feature Discrimination:** Flower species often exhibit subtle inter-class differences, such as minute variations in petal structure or color patterns, which existing FSL approaches struggle to capture.
- **Intra-class Variation:** Individual species display significant variations due to viewing angle, growth stage, and environmental conditions, challenging standard metric learning methods.
- **Scale-Limited Generalization:** While current FSL approaches show promising results for small-scale tasks, they exhibit severe performance degradation when scaling to larger numbers of classes, limiting practical utility.

Building upon prototypical networks and recent advances in transformer architectures, we propose BloomLens, a novel few-shot learning framework with three key technical contributions:

- **Transformer-Enhanced Feature Adaptation:** A self-attention mechanism that dynamically refines prototype representations for robust feature discrimination.
- **Progressive Learning Strategy:** A curriculum-based approach that gradually scales task complexity while maintaining performance.
- **Fine-grained Augmentation Integration:** An adaptive combination of MixUp and CutMix techniques tailored for preserving critical flower features.

Our experimental results demonstrate significant improvements, achieving 93.64% accuracy on 5-way 1-shot tasks and maintaining 85.51% accuracy on 20-way 1-shot scenarios. The framework's scalability to 40-way classification (78.29% accuracy) establishes new benchmarks for few-shot flower classification.

Related Work

Our research builds upon three main areas of prior work: (1) Deep learning approaches for flower classification, (2) Few-shot learning architectures, and (3) Prototypical networks and their variants. Here we review key developments in each area that inform our approach.

Deep Learning for Flower Classification

Deep learning has revolutionized flower classification. Pre-trained models like DenseNet-201 and Wide ResNet-101-2 achieve up to 98.29% accuracy on Oxford Flowers 102[2], while transformer architectures like CCT-14/7x2[3] and CvT-W24[4] reach

2. Albardi et al. 2021.
3. Hassani et al. 2022.
4. Wu et al. 2021.

99.76% accuracy. However, these approaches require extensive labeled training data, limiting their real-world applicability.

Few-shot Learning Approaches

Few-shot learning aims to recognize new classes from limited examples, typically framed as N-way K-shot tasks where models must classify between N novel classes given K examples of each. Recent approaches can be broadly categorized into:

1. **Metric Learning Methods**: These approaches learn an embedding space where similar classes cluster together. Prototypical Networks[5], for example, represent each class by its prototype in the embedding space. Recent work by Derakhshani et al.[6] introduces a Bayesian perspective to prompt learning, modeling the input prompt space as a probabilistic distribution to improve generalization. While showing promise for general few-shot tasks, their approach achieves 70.40% accuracy on 5-way classification tasks for flowers, highlighting the need for domain-specific optimizations in fine-grained scenarios.
2. **Meta-Learning Methods**: These algorithms learn how to learn from few examples. MAML[7] learns initialization parameters that can quickly adapt to new tasks.
3. **Transfer Learning Methods**: These leverage knowledge from base classes with abundant data to help recognize novel classes. Notably, our baseline evaluations show the limitations of direct transfer, with even powerful architectures like DenseNet-201 struggles at difficult few-shot tasks.

Limitations of Current Approaches

While few-shot learning has shown promising results in many domains, its application to fine-grained flower classification faces several key challenges:

1. **Performance Degradation with Increased Classes:** Standard architectures show significant performance drops with more classes.
2. **Feature Adaptation Challenges:** Current approaches struggle to adapt pre-trained features to fine-grained flower characteristics.
3. **Limited Generalization:** Existing augmentation techniques often fail to preserve subtle discriminative features crucial for flower classification.
4. **Deterministic Feature Representations:** Current approaches, including Bayesian prompt learning methods[8], rely on deterministic representations that may not capture classification uncertainty. While promising for 5-way tasks, these methods have not demonstrated success on more challenging 20-way scenarios needed for botanical applications.

5. Snell, Swersky, and Zemel 2017.
6. Derakhshani et al. 2023.
7. Finn, Abbeel, and Levine 2017.
8. Derakhshani et al. 2023.

These limitations motivate our development of BloomLens, which addresses these challenges through progressive training and Transformer-based feature adaptation, combining proven few-shot learning methods with novel components specific to fine-grained flower classification.

Prototypical Networks and Their Enhancements

Our work builds on Prototypical Networks, which compute class prototypes as the mean of embedded support examples for few-shot classification. Key enhancements to this approach include:

1. **Progressive Training**: Li et al.[9] demonstrate benefits of gradually increasing task difficulty.
2. **Data Augmentation**: MixUp[10] interpolates between example pairs to improve generalization, while CutMix[11] replaces image patches to enhance feature localization.
3. **Transformer-based Feature Adaption**: Ye et al.[12] leverage self-attention to adapt support embeddings for more discriminative prototypes, similar to our approach.

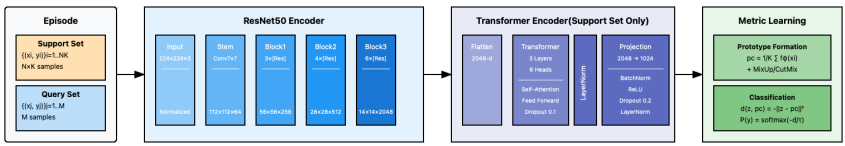


FIGURE 1. *BloomLens Model Architecture*

Methodology

In this section, we detail our architecture design, training strategy, and augmentation techniques specifically tailored for fine-grained flower classification challenges.

Architecture Overview

BloomLens addresses the unique challenges of flower classification through a carefully designed architecture that combines powerful feature extraction with dynamic feature

9. Li et al. 2022.
10. Zhang et al. 2018.
11. Yun et al. 2019.
12. Ye et al. 2021.

adaptation. The architecture consists of three main components (fig. 1):

- **Feature Extraction Backbone**: A ResNet-50 network pretrained on ImageNet serves as our foundation, modified specifically for fine-grained flower features:
  - Removal of the final classification layer to enable few-shot adaptation
  - Discriminative fine-tuning with layer-specific learning rates, allowing deeper layers to better capture flower-specific features
  - Multi-scale feature preservation to capture both macro flower structures and fine textural details
- **Transformer-based Feature Adaptation**: This component dynamically refines feature representations to capture the subtle variations in flower appearances:
  - Input: Support Set embeddings $S = \{s_i\}_{i=1}^{k} \in \mathbb{R}^{k \times 2048}$ representing different examples of each flower class (output dimension of ResNet-50)
  - Transformer encoder with 8-head self-attention, dropout rate of 0.1, and ReLU activation to enhance feature learning
  - Projects 2048-dimensional features to a 1024-dimensional space for efficient prototype computation
  - Self-attention computation focusing on key flower features[13]:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

    where $\sqrt{d_k}$ scaling preserves feature magnitude during attention computation
  - Multi-head attention with $h = 8$ heads to capture different aspects of flower similarity:

$$\begin{aligned} MultiHead(X) &= Concat(head_1, \ldots, head_h)W^O \\ head_i &= Attention(XW_i^Q, XW_i^K, XW_i^V) \end{aligned} \tag{2}$$

    Each head specializes in different feature aspects (e.g., petal arrangement, texture, color patterns)
- **Prototype Computation and Classification**: Creates and utilizes class prototypes that capture essential characteristics of each flower species:
  - Class prototype computation through refined feature averaging:

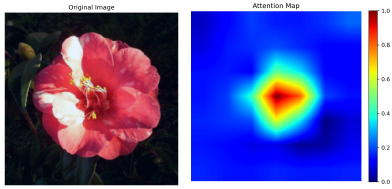
$$p_c = \frac{1}{|S_c|} \sum_{i \in S_c} f_\theta(x_i) \tag{3}$$

    where $f_\theta(x_i)$ represents the transformed feature embedding of support sample x_i
  - Query classification using temperature-scaled cosine similarity:

$$P(y = c|x) = \frac{exp\left(-d(f_\theta(x), p_c)/\tau\right)}{\sum_k exp\left(-d(f_\theta(x), p_k)/\tau\right)} \tag{4}$$

13. Vaswani et al. 2023.

where τ is a learnable temperature parameter that adjusts decision boundaries based on feature space density



(a) *Original Image* **(b)** *Attention Maps*

FIGURE 2. *Attention maps showing discriminative regions used for classification.*

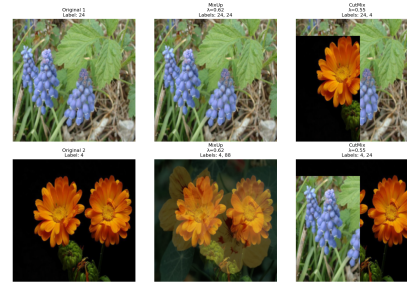


FIGURE 3. *MixUp and CutMix Augmentation*

Progressive Training Strategy

We implement a three-stage curriculum learning approach designed to gradually build the model's capacity for fine-grained flower discrimination, incorporating discriminative fine-tuning with layer-specific learning rates:

- **Initial Stage (5-way)**:
  - Focuses on learning basic flower feature discrimination
  - Uses a base learning rate of $\eta_{base} = 2 \times 10^{-4}$, with higher learning rates for transformer layers set to $3.0 \times \eta_{base}$ to establish effective feature adaptation
  - Emphasizes distinct flower categories to establish strong baseline feature representations
- **Intermediate Stage (10-way)**:
  - Introduces increased class complexity with more similar flower species
  - Reduces learning rates to $0.3 \times \eta_{base}$ for middle layers and $0.1 \times \eta_{base}$ for early layers to refine feature adaptation mechanisms
  - Balances feature discrimination and adaptation capabilities
- **Advanced Stage (20-way)**:
  - Maximizes task complexity with highly similar flower species
  - Uses fine-tuned learning rates of η_{base} for late layers to ensure precise feature discrimination
  - Optimizes prototype separation for challenging cases

Augmentation Integration

We adapt MixUp and CutMix for flower classification, maintaining feature integrity while increasing diversity (fig. 3):

MixUp Adaptation: Performs feature-preserving interpolation with $\alpha = 0.2$:

$$\tilde{x} = \lambda x_i + (1 - \lambda) x_j \tag{5}$$

$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j \tag{6}$$

where $\lambda \sim Beta(0.2, 0.2)$ to favor subtle mixing.

CutMix Integration: Implements structure-aware region mixing with $\alpha = 1.0$:

$$\tilde{x} = M \odot x_i + (1 - M) \odot x_j \tag{7}$$

$$\tilde{y} = \lambda y_i + (1 - \lambda) y_j \tag{8}$$

where $\lambda = 1 - \frac{cut\_area}{total\_area}$ and M is a binary mask aligned with flower regions.

Both augmentations are applied with equal probability when augmentation is triggered (prob=0.5 for each).

Experiments Setup

Datasets and Preprocessing

The Oxford Flowers 102 dataset contains 102 flower categories with 40-258 images per class. We split the dataset into training (61 categories), testing (41 categories), and validation (20% of training images) sets, ensuring complete class separation for genuine few-shot evaluation.

Input images are resized to 224×224 pixels with standard ImageNet normalization (mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]). During training, we employ comprehensive augmentations including random resized crops (scale: 0.8-1.0), horizontal/vertical flips, rotations (±30°), color jittering (brightness/contrast/saturation: 0.4, hue: 0.2), and affine transformations (±0.1 translation, 0.9-1.1 scale). Evaluation uses center cropping and standard normalization.

Training Configuration

We follow episodic training with support set $\mathcal{S}$ and query set $\mathcal{Q}$:

$$\mathcal{E} = \{\mathcal{S}, \mathcal{Q}\}, \text{ where } \mathcal{S} = \{(x_i^s, y_i^s)\}_{i=1}^{N \times K}, \mathcal{Q} = \{(x_j^q, y_j^q)\}_{j=1}^{N \times M} \tag{9}$$

The model is optimized using AdamW with discriminative fine-tuning: early layers (0.1×), middle layers (0.3×), late layers (1×), and transformer layers (3×) of the base learning rate (2e-4). We employ cosine annealing with warm restarts:

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min})(1 + \cos(\frac{T_{\text{cur}}}{T_i}\pi)) \tag{10}$$

where $T_i = T_0 \times (T_{\text{mult}})^i$, $T_0 = 10$, $T_{\text{mult}} = 2$.

Key hyperparameters include:

- Batch size: 32
- Weight decay: 0.01
- Gradient clipping: 10.0
- Early stopping patience: 20 epochs

- Random seed: 42 for reproducibility

Training consists of 100 episodes per epoch with adaptive query sizes (15/10/5 samples for 5/10/20-way tasks). For regularization, we apply MixUp ($\alpha = 0.2$) and CutMix ($\alpha = 1.0$) augmentations with 0.5 probability.

Hardware and Runtime

All experiments were conducted on a single NVIDIA 3090 GPU with CUDA support. Each training stage (5/10/20-way) runs for up to 100 epochs with early stopping, taking approximately 1-2 hours per stage.

Evaluation Protocol

We evaluate against standard CNN architectures (ResNet-18/50, DenseNet-121/201, MobileNet-V2, EfficientNet-B0) across multiple configurations (5/10/20-way, 1/5-shot). Performance is measured over 100 test episodes:

$$\text{Acc}_{\text{final}} = \frac{1}{E}\sum_{i=1}^{E} \text{Acc}_{\mathcal{E}_i} \pm 1.96\sqrt{\frac{\text{Var}(\{\text{Acc}_{\mathcal{E}_i}\}_{i=1}^{E})}{E}} \tag{11}$$

where $\text{Acc}_{\mathcal{E}} = \frac{1}{|Q|}\sum_{(x,y)\in Q} \mathbf{1}[\hat{y} = y]$.

Results and Analysis

Comparison with Baseline Models

Table 1 presents comparative results of BloomLens against baseline models across different configurations. BloomLens significantly outperforms all baselines, achieving state-of-the-art performance on Oxford Flowers-102. On the challenging 20-way 1-shot task, BloomLens achieves 85.51% ± 5.77% accuracy, a substantial 53.54% improvement over the best CNN baseline (DenseNet201: 31.97% ± 1.20%) and notably outperforming recent Bayesian prompt learning approaches[14] which achieve 70.40% ± 1.8% in less challenging 5-way scenarios.

Key observations:

- **Scalability**: While baselines show significant degradation from 5-way to 20-way (average drop of 26%), BloomLens maintains robust performance with only 7.2% decrease. This demonstrates our progressive training strategy's effectiveness. Recent Bayesian approaches show promise in lower-way scenarios but haven't demonstrated successful scaling to 20-way tasks.
- **Shot Efficiency**: The 1-shot to 5-shot performance gap is smaller for BloomLens (3.2% improvement) versus baselines (10.5%), indicating more efficient feature

14. Derakhshani et al. 2023.

Model	5-way 1-shot	5-way 5-shot	20-way 1-shot	20-way 5-shot
AlexNet	41.95 ± 2.01	52.16 ± 2.16	17.13 ± 0.76	22.75 ± 0.78
ResNet18	57.59 ± 2.18	68.61 ± 2.29	31.39 ± 1.07	42.61 ± 0.97
ResNet50	54.21 ± 2.23	63.95 ± 2.30	27.90 ± 0.94	38.16 ± 0.98
DenseNet121	55.16 ± 2.08	67.61 ± 2.06	31.61 ± 1.08	43.69 ± 0.96
DenseNet201	58.52 ± 2.36	69.51 ± 2.06	31.97 ± 1.20	44.47 ± 1.05
Bayesian Prompt[15]	70.40 ± 1.80	73.50 ± 1.50	-	-
BloomLens (Ours)	**93.64 ± 6.86**	**95.88 ± 5.20**	**85.51 ± 5.77**	**89.66 ± 4.00**

TABLE 1. *Few-shot classification accuracy (%) on Oxford Flowers-102. ± denotes 95% confidence interval over 100 episodes. (-) indicates results not reported.*

learning from limited examples, attributed to our transformer-based feature adaptation.

- **Consistency**: BloomLens shows more stable performance across configurations, with lower relative standard deviations compared to baselines and competitive with Bayesian approaches.

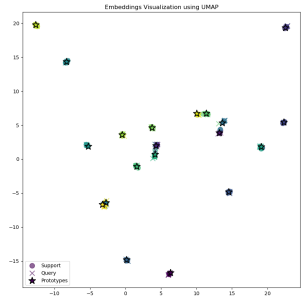


FIGURE 4. *UMAP visualization of the learned feature space*

Feature Space Analysis

UMAP visualization[16] of the learned feature space (Figure 4) reveals:

- **Prototype Effectiveness**: Class prototypes effectively capture class-specific features through optimal equidistant positioning and clear separation, even with

16. McInnes, Healy, and Melville 2020.

limited samples.
- **Query-Support Alignment**: Query samples consistently cluster near corresponding support samples and prototypes, demonstrating robust feature extraction and generalization.
- **Inter-class Separation**: Clear boundaries between categories with minimal overlap, while preserving both fine-grained features and natural visual similarity relationships.

Extended Evaluation

Additional 40-way classification tests show BloomLens achieving 78.29% ± 4.84% and 85.73% ± 3.53% accuracy for 1-shot and 5-shot scenarios respectively. The modest performance drop from 20-way to 40-way tasks (7.22%) validates our approach's robustness and scalability to larger category sets.

Conclusion

This paper introduced BloomLens, a novel few-shot learning framework for fine-grained flower classification. Through extensive experimentation on the Oxford Flowers-102 dataset, we demonstrated significant improvements over baselines, most notably improving the 20-way 1-shot classification accuracy from 31.97% to 85.51%.

The key technical contributions of BloomLens lie in its novel enhancement of the prototypical network framework through three innovations:
- Transformer-based feature adaptation for fine-grained feature discrimination
- Progressive training strategy that effectively scales to complex tasks
- Integrated augmentation approach preserving critical flower features

Our results demonstrate BloomLens's exceptional scalability, maintaining robust performance even in 40-way classification scenarios (78.29% 1-shot, 85.73% 5-shot), making it particularly valuable for real-world applications where labeled data is scarce.

Future Work

While BloomLens demonstrates strong performance, several promising research directions remain:
- **Architecture Optimization**: Exploring efficient variants for resource-constrained deployments
- **Cross-Domain Applications**: Extending to other fine-grained visual tasks beyond flower classification
- **Feature Enhancement**: Developing improved augmentation strategies for better feature preservation

These directions could further advance few-shot learning in scenarios where labeled data is scarce or costly to obtain, particularly in botanical research and conservation applications.

References

Albardi, Feras, H M Dipu Kabir, Md Mahbub Islam Bhuiyan, Parham M. Kebria, Abbas Khosravi, and Saeid Nahavandi. 2021. A Comprehensive Study on Torchvision Pre-trained Models for Fine-grained Inter-species Classification. arXiv: 2110.07097 [cs.CV]. Available at <https://arxiv.org/abs/2110.07097>.

Derakhshani, Mohammad Mahdi, Enrique Sanchez, Adrian Bulat, Victor Guilherme Turrisi da Costa, Cees G. M. Snoek, Georgios Tzimiropoulos, and Brais Martinez. 2023. Bayesian Prompt Learning for Image-Language Model Generalization. arXiv: 2210.02390 [cs.CV]. Available at <https://arxiv.org/abs/2210.02390>.

Finn, Chelsea, Pieter Abbeel, and Sergey Levine. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. arXiv: 1703.03400 [cs.LG]. Available at <https://arxiv.org/abs/1703.03400>.

Hassani, Ali, Steven Walton, Nikhil Shah, Abulikemu Abuduweili, Jiachen Li, and Humphrey Shi. 2022. Escaping the Big Data Paradigm with Compact Transformers. arXiv: 2104.05704 [cs.CV]. Available at <https://arxiv.org/abs/2104.05704>.

Li, Changlin, Bohan Zhuang, Guangrun Wang, Xiaodan Liang, Xiaojun Chang, and Yi Yang. 2022. Automated Progressive Learning for Efficient Training of Vision Transformers. arXiv: 2203.14509 [cs.CV]. Available at <https://arxiv.org/abs/2203.14509>.

McInnes, Leland, John Healy, and James Melville. 2020. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv: 1802.03426 [stat.ML]. Available at <https://arxiv.org/abs/1802.03426>.

Snell, Jake, Kevin Swersky, and Richard S. Zemel. 2017. Prototypical Networks for Few-shot Learning. arXiv: 1703.05175 [cs.LG]. Available at <https://arxiv.org/abs/1703.05175>.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention Is All You Need. arXiv: 1706.03762 [cs.CL]. Available at <https://arxiv.org/abs/1706.03762>.

Wu, Haiping, Bin Xiao, Noel Codella, Mengchen Liu, Xiyang Dai, Lu Yuan, and Lei Zhang. 2021. CvT: Introducing Convolutions to Vision Transformers. arXiv: 2103.15808 [cs.CV]. Available at <https://arxiv.org/abs/2103.15808>.

Ye, Han-Jia, Hexiang Hu, De-Chuan Zhan, and Fei Sha. 2021. Few-Shot Learning via Embedding Adaptation with Set-to-Set Functions. arXiv: 1812.03664 [cs.LG]. Available at <https://arxiv.org/abs/1812.03664>.

Yun, Sangdoo, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. 2019. CutMix: Regularization Strategy to Train Strong Classifiers with Localizable Features. arXiv: 1905.04899 [cs.CV]. Available at <https://arxiv.org/abs/1905.04899>.

Zhang, Hongyi, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond Empirical Risk Minimization. arXiv: 1710.09412 [cs.LG]. Available at <https://arxiv.org/abs/1710.09412>.