Leaf Cultivar Identification via Prototype-enhanced Learning

Yiyi Zhang[1], Zhiwen Ying[1], Ying Zheng[2], Cuiling Wu[1], Nannan Li[1], Jun Wang[1], Xianzhong Feng[3], Xiaogang Xu[4*]

[1] *Institute of Intelligent computing, Zhejiang Lab, Hangzhou, China*
[2] *AI Lab, Gientech, Hangzhou, China*
[3] *Northeast Institute of Geography and Agroecology, Chinese Academy of Sciences, Changchun, China*
[4] *School of Computer Science and Technology, Zhejiang Gongshang University, Hangzhou, China*
yiyi.zhang93@outlook.com, zhengying@hit.edu.cn, xxgang2013@163.com

Abstract

Plant leaf identification is crucial for biodiversity protection and conservation and has gradually attracted the attention of academia in recent years. Due to the high similarity among different varieties, leaf cultivar recognition is also considered to be an ultra-fine-grained visual classification (UFGVC) task, which is facing a huge challenge. In practice, an instance may be related to multiple varieties to varying degrees, especially in the UFGVC datasets. However, deep learning methods trained on one-hot labels fail to reflect patterns shared across categories and thus perform poorly on this task. To address this issue, we generate soft targets integrated with inter-class similarity information. Specifically, we continuously update the prototypical features for each category and then capture the similarity scores between instances and prototypes accordingly. Original one-hot labels and the similarity scores are incorporated to yield enhanced labels. Prototype-enhanced soft labels not only contain original one-hot label information, but also introduce rich inter-category semantic association information, thus providing more effective supervision for deep model training. Extensive experimental results on public datasets show that our method

---

* Corresponding author
First Author and Second Author contributed equally.

can significantly improve the performance on the UFGVC task of leaf cultivar identification.

1. Introduction

Cultivar identification plays a vital role in the evaluation, breeding, and production of plant multi-variety. In botany, plant leaves are widely utilized [1] to identify varieties due to their stable, persistent, and detective morphological characteristics. Conventional methods of plant identification often require empirical knowledge that is not readily available to non-experts. The process of manual variety classification is time-consuming and error-prone, making it difficult to meet the high demand for plant ecological research.

Over the past decade, researchers have successfully extracted topological features from leaves in terms of texture, shape, and venation. However, these hand-crafted traits are low-level features [2], which are not descriptive enough to distinguish cultivars belonging to the same species. Intuitively, deep learning is ideal to extract rich high-level features from leaf images. Existing leaf classification methods can be roughly divided into two sets. One set fuses multiple features extracted from different parts of leaves or through different manners to jointly identify leaf cultivar [3–6]. These hybrid methods require extra computational cost to calculate a group of inputs. The other set performs cascaded frameworks to encode features through a pipeline [7–9]. Nevertheless, these methods need additional network structures and some of them cannot be trained in an end-to-end manner. In brief, existing deep learning methods for leaf cultivar classification are not effective to meet the demands of practical scenarios.

Moreover, deep learning approaches rely heavily on large amounts of annotated data, where the quality of labels is critical to model performance. In many cases, errors in manual labeling are inevitable, directly leading to per-
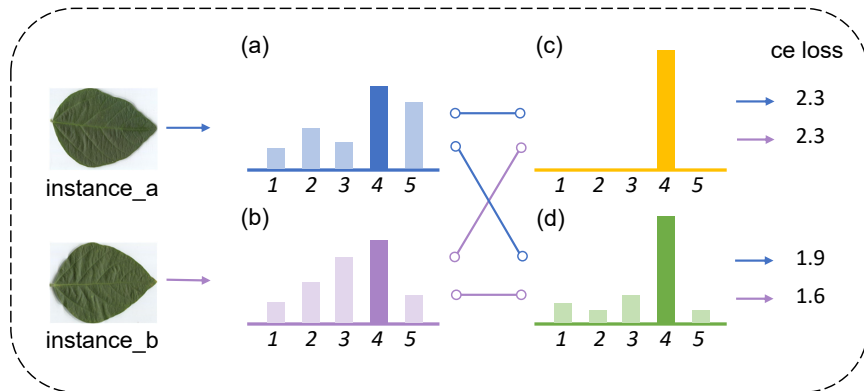
Figure 1: A case study comparing the effect of the soft and hard targets. (a) and (b) are predicted label distributions of two instances from the same class; (c) is the one-hot hard target; (d) is the simulated soft target. Though both instances are classified to the positive class 4 with the same ratio, negative classes carry different ratios of probability in (a) and (b). By computing the cross-entropy (ce) loss, the soft target reveals this information gap while the hard target treats (a) and (b) equally. o-o represents distance calculation between two distributions.

formance degradation. The mislabeling problem is exacerbated especially in UFGVC datasets where each class is highly similar. Meanwhile, the typical one-hot labeling used to train deep models assigns the full probability to one single class, making the model particularly vulnerable to mislabeled instances and appearing overconfident. As expected, works of knowledge distilling [10, 11] have claimed the advantages of soft targets. When the soft targets have high entropy, they are much more likely to provide richer information per training case than hard targets (one-hot labels). As illustrated in Figure 1, all negative classes are treated equally in hard targets, while soft targets can measure the difference behind negative classes, leading to different gradient directions in the training phase. A label smoothing(LS) method [12] was proposed to alleviate the limitation of one-hot labels through a "confidence penalty". However, [13] demonstrated that LS loses information in the logits about resemblances among different classes. In light of these pros and cons, we thus ask: how to generate soft targets for improved ultra-fine-grained classification?

As an analogy to natural language processing (NLP), by capturing the co-relation between labels, label embedding [14] can select the most informative words and neglect irrelevant ones when predicting different labels for multi-label text classification. Based on this intuition, we propose a novel Prototype-enhanced Learning (PEL) method for leaf cultivar identification. PEL is predicated on the assumption that label embedding encoded with the inter-class relationships would force the image classification model to focus on discriminative patterns. Unlike classical label embedding in NLP, we capture inter-class co-relation at the feature level by generalizing categorical prototypes. The idea of prototypes is related to prototypical networks [15, 16] in few-shot learning. They assign instance labels to the closest prototypes in the classifier. However, we use prototypes to augment original labels, which is significantly different from them. In PEL, the prototypes are updated via a moving average during training, thus continuously approaching the corresponding class centers on-the-fly. We conduct similarity scoring between input features and prototypes. Similarity scores are fused into one-hot label representations to generate enhanced labels. We use the obtained logits to replace original labels for supervised model learning.

With the help of PEL, a deep network not only learns to distinguish varieties but also grasps the semantic relationship between each label. The proposed PEL is compared with 22 state-of-the-art methods including hand-crafted feature descriptors, CNN-based and transformer-based deep learning methods. Encouraging experimental results are reported on 7 ultra-fine-grained image datasets, demonstrating the effectiveness of PEL for the UFGVC tasks. The main contributions are summarized as follows:

- A novel method named Prototype-enhanced Learning (PEL) is proposed for leaf cultivar identification. Compared to existing methods, PEL is end-to-end trainable and adds no extra parameters, which is light and effective with negligible computational overhead.

- We develop a new prototype update module to learn inter-class relations

4

by capturing label semantic overlap and iteratively update prototypes to generate continuously enhanced soft targets.

- Extensive experiments on 7 benchmarks adequately illustrate the superiority of PEL for UFGVC. State-of-the-art performances are reported on two widely used backbones including ResNet and DenseNet.

2. Related Work

This section reviews various plant leaf recognition methods from the species level and cultivar level separately. In detail, some representative methods are surveyed, including hand-crafted feature-based approaches and deep learning-based approaches.

2.1. Plant Species Recognition

Hand-crafted descriptors. Traditional hand-crafted descriptors manually extract leaf visual characteristics, such as leaf shape, texture, vein, and color, to identify plant species. Leaf shapes provide significant clues for botanists to identify species[17]. [18] proposed a novel contour-based shape descriptor to capture robust shape geometry which can be invariant to translation, rotation, scaling, and bilateral symmetry. [19] made the first attempt to introduce the idea of bag-of-words (BoW) for shape representation in which the shapes are decomposed into contour fragments. [20] utilized a novel feature that captures global and local shape information independently. Furthermore, methods based on leaf texture and veins also have been presented in the past few decades. For example, [21] explored to use Gabor co-occurrences in plant texture classification. [22] presented a texture description approach by combining LBP feature with a gray-level co-occurrence matrix (GLCM) for tea leaf classification. [23] adopted a procedure for segmenting and classifying scanned legume leaves based only on the analysis of their veins. [24] designed a new descriptor named Eagle which characterizes the overall venation structure using the edge patterns

5

among neighboring regions. Recently, [25] put forward a feature fusion framework by integrating the shape information and the texture feature for plant leaf recognition.

Deep learning methods. In the past few years, several leaf identification approaches using deep learning have been developed. [26] designed a framework by first segmenting the leaf from the background, extracting features representing the curvature of the leaf's contour to identify numerous tree species. [27] attempted using a deep convolutional neural network (CNN) for the problem of plant identification from leaf vein patterns. [6] adopted a hybrid global-local leaf feature extraction method for plant classification. [28] developed a deep learning system to learn discriminative features from leaf images along with a classifier for species identification of plants. A procedure for segmenting and classifying scanned legume leaves based only on the analysis of their veins was proposed [23]. [29] presented a dual-path deep convolutional neural network (CNN) to learn joint feature representations for leaf images by exploiting their shape and texture characteristics. [30] proposed a multi-organ plant identification approach based on a CNN and recurrent neural network. They analyzed features of leaf and other organs, such as fruit, steam, or flower for plant species identification. [7] put forward a framework for simulating botanist behaviors through three deep learning-based models. Nevertheless, these methods tend to perform poorly when transferred from species classification to cultivar classification. This is due to large intraclass distances and small inter-class distances among cultivars.

2.2. Plant Cultivar Recognition

Using leaf image patterns as clues for identifying plant species has achieved great success in the past decades. Recently, there is an increasing concern about whether leaf image patterns can also provide powerful discriminative information for cultivar-level recognition.

Hand-crafted descriptors [1] proposed a novel multi-scale sliding chord matching approach to extract leaf patterns that are distinctive for soybean culti-

6

var identification. [31] put forward a novel Multi-Orientation Region Transform (MORT), which can effectively characterize both contour and structure features simultaneously. The proposed MORT can extract local structural features at various scales and orientations for comprehensive shape description. [2] introduced a leaf cultivar classification model based on forest representation learning and multi-scale contour feature learning. [32] designed a novel local binary pattern, named pairwise rotation-difference LBP (PRDLBP), for the characterization of leaf image patterns. [33] attempted a new strategy of depicting leaf shapes by convolving the contour vector functions with Gaussian functions of different widths. [39] proposed a novel local R-symmetry co-occurrence method (RsCoM) for characterizing discriminative local symmetry patterns to distinguish subtle differences among cultivars.

Deep learning methods [5] combined leaf hand-crafted features and deep learning extracted features together to identify 5000 leaf images from 100 soybean cultivars. [3] explored to fuse deep learning features of triplet leaves from different parts of soybean plants for effective cultivar recognition. [34] proposed an efficient and convenient method for the classification of apple cultivars using a deep convolutional neural network, which is the delicate symmetry of human brain learning. [4] integrated an auxiliary self-supervised learning module (MaskCOV) with a powerful in-image data augmentation scheme for cultivar classification. [51] presented SPARE, a self-supervised part-erasing framework for ultra-fine-grained visual categorization. The key insight of SPARE is to learn discriminative representations by encoding a self-supervised module to predict the position of the erased parts. [9] introduced a novel mixing attentive vision transformer (Mix-ViT) incorporated with a self-supervised module to address the UFGVC tasks. [35] concluded that most work of plant classification is performed on the author's dataset, which makes a comparison of different works difficult. In order to achieve an adequate comparison among existing methods, we conduct comprehensive experiments with 22 competing methods on 7 benchmark datasets. Experimental results fully validate the contribution of our proposed method.
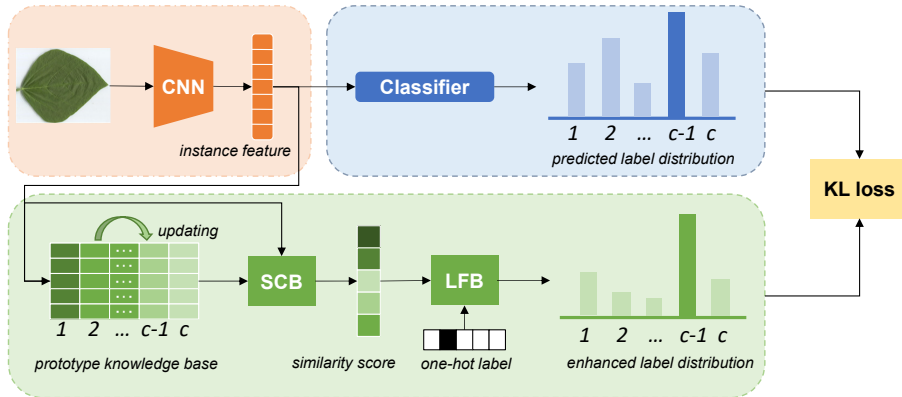
Figure 2: Framework of the proposed Prototype-enhanced Learning(PEL) method. Given an input image, the CNN layers extract the instance feature representation. The classifier takes it to predict the instance label distribution. In parallel, we use the instance feature to update the prototype knowledge base, then compute the similarity score through Similarity Computing Block(SCB). The similarity score is then fused into the original one-hot label by Label Fusing Block(LFB), resulting in enhanced label distribution. Finally, the predicted label distribution and the enhanced label distribution are used to compute the Kullback-Leibler divergence loss.

3. Methods

In this section, we present the proposed Prototype-enhanced Learning (PEL) method. The PEL-based classification predictor consists of three modules: a basic feature encoder, a softmax classifier, and a label simulator. An overview of PEL is shown in figure 2.

3.1. Basic Classification Predictor

The PEL is designed to address the problem that conventional one-hot instance labeling makes the network overconfident and limits the model from learning semantic overlap between classes. We introduce PEL starting from the basic classification predictor. A basic classification predictor usually includes a feature encoder and a softmax classifier. Given the i-th input image x_i, the feature encoder extracts the instance feature $f(x_i)$ and feeds it to the classifier. The classifier is a single fully connected (FC) layer followed by a softmax layer

to generate the predicted label distribution $\hat{y}_i$:

$$\hat{y}_i = softmax(FC(f(x_i))), \tag{1}$$

$$\hat{y}_i^j = \frac{exp(FC(f(x_i))_j/t_1)}{\sum_{n=1}^{N} exp(FC(f(x_i))_n/t_1)}, \tag{2}$$

where $\hat{y}_i^j$ is the value in the j-th dimension of $\hat{y}_i$, $FC(f(x_i))_j$ is the j-th dimension of the predicted label logits from the i-th instance, N is the number of categories, and t_1 is the temperature coefficient.

Traditional classification models apply cross-entropy loss between $\hat{y}_i$ and the ground-truth class y_i to supervise the training process:

$$CE(y_i, \hat{y}_i) = -\sum_{j=1}^{N} y_i^j log(\hat{y}_i^j). \tag{3}$$

In the above $y_i^j \in \{0, 1\}$ specifies the one-hot ground-truth class distribution in the j-th dimension.

However, one-hot labels are the same for samples of the same class, regardless of their contents. In fact, samples with the same label may have quite different contents, and naturally their label distributions should also be different. Although a theoretically realistic label distribution is not easy to achieve, we can try to model a distribution that reflects the degree of relationship between instances and labels.

3.2. PEL-based Classification Predictor

Different categories exhibit varying degrees of shared features with each other, implying that labels are not completely independent. With this basic intuition, we propose the PEL method with a label simulator to simulate the correlated label distribution. The pipeline of PEL is illustrated in Figure 3. We first give a network initial prototypes [15] of each class as prior knowledge, which is termed prototypical knowledge base $\mathcal{P} \in \mathbb{R}^{N \times D}$. We use normalized features generated from the final pooling layer for all backbone networks. This prototypical knowledge base is implemented as a matrix, where N is the number of classes and D is the prototype dimension. Each column represents a unique
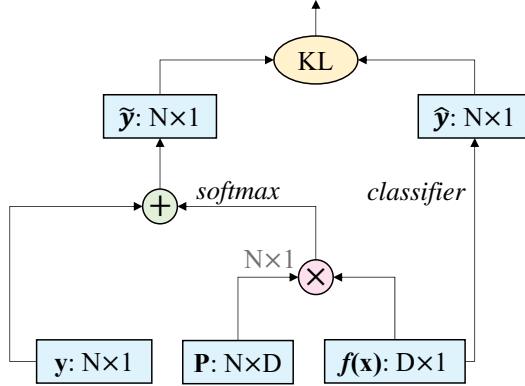
Figure 3: Pipeline of the proposed PEL method. In this picture, $\otimes$ is matrix multiplication, $\oplus$ is element-wise addition, and KL represents the Kullback-Leibler divergence loss.

prototypical feature representation of a class. During training, normalized instance features in one batch will be grouped by their corresponding ground-truth labels. We then compute the mean feature representations in each group. The above process can be formulated as:

$$\mathcal{F}_n = \frac{\sum\limits_{y_i \in n} f(x_i)}{|n|}, \tag{4}$$

where x_i denotes the i-th instance in one batch, y_i represents the ground-truth label of x_i, and $|n|$ represents the number of instances belonging to the n-th category, we have $\mathcal{F}_n$ denoting the mean feature representation of the n-th category.

The output mean features $\mathcal{F}$ are fed to the prototypical knowledge base $\mathcal{P}$ and the corresponding prototypes are updated accordingly by:

$$\mathcal{P}_n \leftarrow \mathcal{P}_n + \alpha(\mathcal{F}_n - \mathcal{P}_n), \tag{5}$$

where $\mathcal{P}_n$ denotes the prototypical feature representation of the n-th category, $\alpha \in (0, 1)$ is the momentum coefficient that controls the updating rate of each prototype.

After that, the similarity computing block(SCB) will compute the similarity scores w between each instance feature and the prototypical knowledge base.

Intuitively, if an instance is close to a prototype, it may relate to the label of that prototype. Thus even if two instances belong to the same class, they carry different information w.r.t the similarity with other classes. The dependency among labels will be captured instance-specifically, which is superior to a uniform noise distribution as in label smoothing [12]. In practice, we use cosine similarity distance as the similarity metric. The similarity scores are then normalized by a softmax function. Given the input instance feature $f(x_i)$, the corresponding similarity score of the i-th instance w_i is computed as:

$$w_i = softmax(\mathcal{P} \times f(x_i)), \tag{6}$$

$$w_i^j = \frac{exp((\mathcal{P} \times f(x_i))_j / t_2)}{\sum_{n=1}^{N} exp((\mathcal{P} \times f(x_i))_n / t_2)}, \tag{7}$$

where w_i^j is the value in the j-th dimension of w_i, $(\mathcal{P} \times f(x_i))_j$ is the j-th dimension of the predicted label logit from the i-th instance, N is the number of categories, and t_2 is the temperature coefficient.

The label fusing block(LFB) takes the one-hot ground-truth targets $\boldsymbol{y}$ and similarity scores $\boldsymbol{w}$ as inputs, and fuses them with a weight coefficient. We define the enhanced label distribution $\widetilde{y_i}$ as:

$$\widetilde{y_i} = \beta y_i + w_i, \tag{8}$$

where $\beta > 0$ is a weight coefficient to control the effect by the one-hot target y_i of the i-th instance .

As a result, the enhanced label distribution not only contains groud truth information but also has inter-class similarity awareness. Finally, the enhanced label distribution takes place of the one-hot target to supervise the model training. The Kullback-Leibler divergence (KL) loss is applied to measure the distance $\mathcal{L}$ between the predicted and enhanced label distribution, which is formulated by:

$$\mathcal{L} = KL\text{-}divergence(\widetilde{\boldsymbol{y}}, \hat{\boldsymbol{y}})$$
$$= \sum_{i}^{I} \widetilde{y_i} log(\frac{\widetilde{y_i}}{\hat{y_i}}), \tag{9}$$

where I is the number of instances.

In the proposed PEL, the actual labels are dynamically changing along with the updating of prototypes. With the additional supervision from the prototypical knowledge base, models can learn much more information and reduce overfitting when facing small datasets. Moreover, the enhanced labels are more robust to mislabeled samples since the similarity scores can allocate probability to similar labels which usually include the right label. Please note that only the "CNN" and "classifier" are needed during inference, so the computational cost introduced is negligible.

4. Experiments

The performance of our proposed PEL is compared to 22 classification methods for a thorough evaluation. The benchmark methods are broadly split into two groups. One covers hand-crafted feature descriptors including SIT[37], HSC[38], PH[5], RsCoM[39] and MSCM[1]. The second is deep learning methods consisting of 1) Basic CNN-based methods including Alexnet[40], VGG-16[41], MobileNetV2[42] and Xception[43]; 2) Basic transformer-based methods including ViT[44], DeiT[45], TransFG[46] and Hybrid-ViT[44]; 3) state-of-the-art methods for image recognition and FGVC including NTS-NET[47], fast-MPN-COV[48], DCL[49] and Cutmix[50]; 4) state-of-the-art methods for UFGVC including CF[3], MFCIS[5], MaskCOV[4], SPARE[51], Mix-ViT[9].

Dataset	Class	Train	Test
Sweet cherry	88	3788	1623
CottonCultivar	80	240	240
SoyCultivarLocal	200	600	600
SoyCultivarGene	1110	12763	11143
SoyCultivarAge	198	4950	4950
SoyCultivarGlobal	1938	5814	5814
SoyCultivar200	200	3000	3000

Table 1: Statistics of the benchmark datasets.

4.1. Leaf material and Benchmarks

In this research, we adopted 7 benchmark datasets in our experimental evaluation. Table 1 summarizes the statistics for each dataset, i.e., the number of classes, training set, and testing set. For ultra-fine-grained visual classification (UFGVC), there are 7 image datasets including sweet cherry [5], SoyCultivarLocal [36], CottonCultivar, SoyCultivarGlobal, SoyCultivarGene, SoyCultivarAge, and SoyCultivar200[1].

Methods	Backbone	Accuracy(%)				
		Cotton	S.Loc	S.Gene	S.Age	S.Glo
Alexnet[40]	Alexnet	22.92	19.50	13.12	44.93	13.21
VGG-16[41]	VGG-16	50.83	39.33	63.54	70.44	45.17
MobileNetV2[42]	ResNet-50	49.58	34.67	63.17	-	31.66
NTS-NET[47]	ResNet-50	51.67	42.67	-	-	-
fast-MPN-COV[48]	ResNet-50	50.00	38.17	45.26	-	11.39
Cutmix[50]	ResNet-50	45.00	26.33	66.39	62.68	30.31
DCL[49]	ResNet-50	53.75	45.33	71.41	73.19	42.21
MaskCOV[4]	ResNet-50	58.75	46.17	73.57	75.86	50.28
SPARE[51]	ResNet-50	60.42	44.67	79.41	75.72	56.45
ViT[44]	Transformer	52.50	38.83	53.63	66.95	40.57
DeiT[45]	Transformer	54.17	38.67	66.80	69.54	45.34
TransFG[46]	Transformer	54.58	40.67	22.38	72.16	21.24
Hybrid-ViT[44]	Transformer&ResNet	50.83	37.00	71.74	73.56	18.82
Mix-ViT[9]	Transformer&ResNet	60.42	56.17	<u>79.94</u>	76.30	51.00
PEL	ResNet-50	<u>62.92</u>	<u>58.67</u>	79.50	<u>81.45</u>	<u>57.58</u>
PEL	DenseNet-121	**63.33**	**59.33**	**81.49**	**82.30**	**60.56**

Table 2: The top 1 classification accuracy on the CottonCultivar (Cotton), SoyCultivarLocal (S.Loc), SoyCultivarGene (S.Gene), SoyCultivarAge (S.Age) and SoyCultivarGlobal (S.Glo) datasets. Results style: **best** and <u>second-best</u> among each method.

4.2. Implementation Details

We implement our proposed PEL in Pytorch. All networks are trained and tested on a single Tesla A-100 GPU. We evaluate PEL on two widely used backbones: ResNet-50 and DenseNet-121. They are initialized by the ImageNet pre-trained model. The input images are resized to 512×512 and center cropped into 448×448. Random rotation with a degree of 15 and random horizontal flips are adopted for data augmentation. The above are standard setups in the literature. We adopt an SGD optimizer with a momentum of 0.9 and weight decay 1e-4. The base learning rate is 0.001 and the batch size is set to 8 for all datasets, except for SoyCultivarGlobal with a learning rate of 0.01 and a

Methods	Backbone	Acc(%)					
		R1	R3	R4	R5	R6	Avg
Alexnet[40]	Alexnet	49.90	44.65	45.15	47.47	37.47	44.93
VGG-16[41]	VGG-16	72.32	72.53	74.95	71.11	61.31	70.44
NTS-NET[47]	ResNet-50	63.94	67.68	51.52	61.41	55.76	60.06
fast-MPN-COV[48]	ResNet-50	67.68	64.55	66.87	68.49	50.71	63.66
Cutmix[50]	ResNet-50	65.56	59.19	64.24	68.79	53.64	62.28
DCL[49]	ResNet-50	76.87	73.84	76.16	76.16	62.93	73.19
MaskCOV[4]	ResNet-50	79.80	74.65	79.60	78.28	66.97	75.86
SPARE[51]	ResNet-50	78.28	79.90	78.69	77.27	64.44	75.72
ViT[44]	Transformer	69.29	64.55	70.40	71.01	59.49	66.95
DeiT[45]	Transformer	73.03	70.40	69.09	74.65	60.51	69.54
TransFG[46]	Transformer	74.95	74.55	74.24	76.26	60.81	72.16
Hybrid-ViT[44]	Transformer&ResNet	77.17	76.97	74.75	76.36	63.53	73.56
Mix-ViT[9]	Transformer&ResNet	79.29	77.17	77.98	79.19	67.88	76.30
PEL	ResNet-50	<u>84.24</u>	<u>82.53</u>	<u>84.14</u>	**84.64**	**71.71**	<u>81.45</u>
PEL	DenseNet-121	**85.85**	**83.53**	**86.26**	<u>84.44</u>	<u>71.41</u>	**82.30**

Table 3: The top 1 classification accuracy on the five subsets R1, R3, R4, R5 and R6 of the SoyCultivarAge dataset. "Avg" denotes the average classification accuracy of the five subsets. Results style: **best** and <u>second-best</u> among each method.

batch size of 64. The number of epochs is selected according to the data size. All datasets use epoch 120 except for the larger dataset SoyCultivarGene with epoch 200. The temperature coefficients $t1$ (Eq. 2) and $t2$ (Eq. 7) are both set as 1. The momentum coefficient α (Eq. 5) is empirically set to 0.9. Besides, we set the weight coefficient β (Eq. 9) as 6 in all datasets except for SoyCultivarGene and SoyCultivarGlobal with β as 8. Ablation studies on the effect of the weight coefficient are reported in section 4.4. Given an ImageNet-pretrained backbone network, the prototypical knowledge base is initialized as the mean feature of each category on the training set. During test time, the architectures of SCB and LFB are removed.

4.3. Performance Comparison

To assess the capability of PEL in the UFGVC tasks, we conduct evaluations on 7 ultra-fine-grained datasets. Most of the datasets have been split into train and test sets, except in the sweet cherry dataset, we adopt ten random train and test splits as in MFCIS[5] with a split ratio of 7:3.

Comparison on CottonCultivar Dataset. As reported in Table 2, the proposed PEL surpasses all the competitive methods w.r.t the classification ac-

Methods	Backbone	Accuracy(%)			
		Low	Mid	Up	Avg
SIT[37]	-	18.30	12.15	13.20	14.55
HSC[38]	-	23.00	18.80	16.15	19.32
RsCoM[39]	-	30.15	28.04	31.15	29.78
MSCM[1]	-	34.70	33.55	31.03	33.09
CF[3]	ResNet-50	37.40	40.10	39.01	38.84
CF[3]	DenseNet-121	43.50	47.15	44.90	45.18
MFCIS[5]	Xception	76.00	76.02	79.67	77.23
DCL[49]	ResNet-50	79.52	<u>85.40</u>	<u>87.20</u>	<u>84.04</u>
MaskCOV[4]	ResNet-50	79.70	81.20	83.50	81.47
Mix-ViT[9]	Transformer&ResNet	78.82	81.74	84.13	81.56
PEL	ResNet-50	<u>79.90</u>	84.50	85.41	83.27
PEL	DenseNet-121	**80.55**	**86.83**	**88.84**	**85.34**

Table 4: The top 1 classification accuracy on the three subsets Low, Mid and Up of the SoyCultivar200 dataset. "Avg" denotes the average classification accuracy of the three subsets. Results style: **best** and <u>second-best</u> among each method.

curacy. Specifically, PEL based on DenseNet-121 achieves the best performance, which is 2.91% higher than the second-best methods Mix-Vit and SPARE.

Comparison on SoyCultivarLocal Dataset. PEL presents favorable performance results as listed in Table 2. DenseNet-121 based PEL achieves the best performance at 59.33% among 15 outstanding methods, followed by the recently proposed state-of-the-art method Mix-Vit with a classification accuracy of 56.17%.

Comparison on SoyCultivarGene Dataset. In Table 2, DenseNet-121 based PEL obtains the highest classification accuracy of 81.49%, outperforming all the other 14 methods. We observe that Mix-ViT shows its superiority while evaluated on a relatively larger dataset (SoyCultivarGene) with more than 20 thousand image samples.

Comparison on SoyCultivarGlobal Dataset. As shown in Table 2, PEL are competitive compared to the state-of-the-art methods. For example, DenseNet-121 based PEL gains 4.11% improvement over the second-best approach SPARE.

Comparison on SoyCultivarAge Dataset. SoyCultivarAge dataset cov-

Methods	Backbone	Accuracy(%)	
		Cherry	SoyCultivar200
HSC[38]	-	16.47	45.45
PH[5]	-	42.08	-
DCNN[34]	-	32.55	-
RsCoM[39]	-	-	64.93
MSCM[1]	-	-	72.40
CF[3]	DenseNet-121	-	83.55
Xception[43]	Xception	66.52	-
MFCIS[5]	Xception	83.52	78.00
DCL[49]	ResNet-50	<u>95.50</u>	87.50
MaskCOV[4]	ResNet-50	91.75	83.92
Mix-ViT[9]	Transformer&ResNet	93.10	84.50
PEL	ResNet-50	95.34	**89.33**
PEL	DenseNet-121	**95.62**	<u>89.20</u>

Table 5: The top 1 classification accuracy on the sweet cherry and SoyCultivar200 dataset. Results style: **best** and <u>second-best</u> among each method.

ers five subsets where each subset contains leaves collected from a specific cultivar period. The comparative performance of 14 competing methods is summarized in Table 3. For each subset, PEL shows dominant capability compared to other approaches. Particularly, DenseNet-121 based PEL achieves the best performance with a significant margin of 6.05% over the second-best method MaskCOV in the R1 period subset. Moreover, we observe that R6 exhibits a substantially more than 10% lower accuracy than other periods.

Comparison on SoyCultivar200 Dataset. Table 4 presents the performance of existing hand-crafted feature extraction methods as well as competitive deep learning methods on the subsets of the SoyCultivar200 dataset. According to the results, deep learning methods are much superior to hand-crafted feature extraction methods. Furthermore, results demonstrate that soy leaves collected from the low part of one plant are less discriminative than those from the mid and up parts. The finding suggests that the newly emerging leaves in the upper part of a plant carry richer information than the mature leaves in the lower part. In addition, we mix up all subsets in SoyCultivar200, train and test each approach on the whole dataset. As Table 5 shows, PEL achieves the best
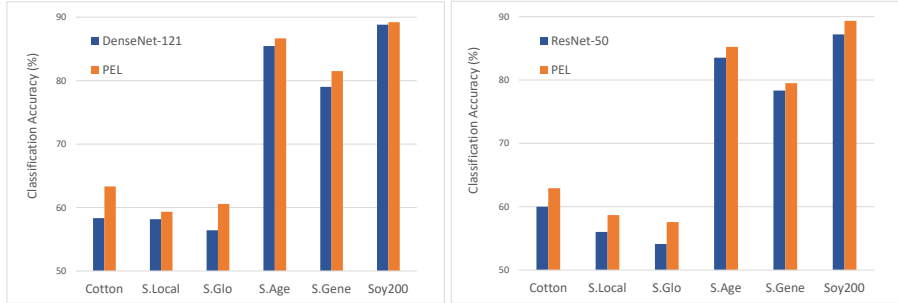
Figure 4: Performance comparison between baselines and PEL.

performance 89.33%.

Comparison on sweet cherry Dataset. In Table 5, we compare PEL with the existing methods on the sweet cherry dataset. We observe that DenseNet-121 based PEL achieves encouraging classification accuracy which is 95.62%. Besides, DCL also shows its advantage with only 0.12% lower than our method.

The overall performance is consistent with the number of samples in each category. There are only around 3 samples per cultivar in CottonCultivar, SoyCultivarLocal, and SoyCultivarGlobal, which involves a few-shot problem in the UFGVC tasks. Therefore, the performances on these three benchmark datasets are relatively low which are around 60%. As we know that few shot datasets are prone to overfitting, prototypical learning is a conventional way for few shot tasks. Thus, our prototype-enhanced learning approach helps alleviate the few-shot problem in UFGVC tasks. Concluded from Table 4 and Table 5, deep learning methods have a very high performance ceiling that dominates comparison with hand-crafted feature extraction methods. In short, the results demonstrate the potential of our proposed method toward narrowing the performance gap between the FGVC and UFGVC tasks.

4.4. Ablation Studies

We conduct a comprehensive ablation study to further verify the contribution of the proposed PEL method. The baseline methods includes ResNet-50[52] and DenseNet-121[53]. As shown in Figure 4, PEL consistently improves the

Backbone	Dataset	weight	Accuracy(%)
		4	86.46
ResNet-50[52]	SoyCultivar200	6	**89.33**
		8	88.20
		6	44.94
DenseNet-121[53]	SoyCultivarGlobal	8	**60.56**
		10	59.49

Table 6: Ablation on the effect of weight coefficeint regarding classification accuracy on two benchmark datasets.

Method	Backbone	Dimension	Parameters	GFLOPs
ViT[44]	Transformer	**0.7K**	86.24M	-
DeiT[45]	Transformer	**0.7K**	86.24M	-
TransFG[46]	Transformer	**0.7K**	86.80M	-
Hybrid ViT[44]	Trans.&ResNet	**0.7K**	99.20M	-
Mix-ViT[9]	Trans.&ResNet	**0.7K**	**35.42M**	-
Alexnet[40]	Alexnet	4K	61.10M	**2.84**
VGG-16[41]	VGG-16	4K	138.36M	61.64
ResNet-50[52]	ResNet-50	2K	25.56M	16.47
DenseNet-121[53]	DenseNet-121	**1K**	**7.0M**	11.53
DCL[49]	ResNet-50	2K	23.68M	16.47
MaskCOV[4]	ResNet-50	2K	23.75M	16.47
PEL	ResNet-50	2K	25.56M	16.47
PEL	DenseNet-121	**1K**	**7.0M**	11.53

Table 7: Performance comparison with respect to feature dimension, parameters and GFLOPs with input size of 448×448. The results in bold represent the most efficient in each group of methods.

performance over baselines, indicating the effectiveness of the PEL method.

Effect of weight coefficient on PEL. As we mentioned before, the similarity scores conducted in PEL may introduce noises in the early training stage when prototypes are unstable and imprecise. Therefore, we implement ablation studies of the effect of the weight coefficient β (Eq. 9) on SoyCultivar200 and SoyCltivarGlobal respectively. In Table 6, we compare ResNet-50 based PEL evaluated on SoyCultivar200 with different weight coefficients from 4 to 8. We set β to 6 for experiments on all datasets except SoyCultivarGene and SoyCultivarGlobal. As presented in Table 6, the weight coefficient of 8 achieves the best performance by the DenseNet-121 based PEL on SoyCltivarGlobal. We set

β as 8 for all experiments on SoyCultivarGene and SoyCultivarGlobal.

Computational cost. In Table 7, we list the comparison results in three aspects: feature dimension (a key measurement in classification tasks), computational complexity (GFLOPs), and memory consumption (model parameters). For a fair comparison, the input size is set to 448×448 and the category number is set to 80 for all the competing methods. The DenseNet-121 based PEL is relatively effective in each aspect with only 11.53 GFLOPs, 1K feature dimension, and 7M model parameters.

5. Conclusions

In this paper, we propose a novel PEL method for the challenging ultra-fine-grained image recognition. As an open topic, cultivar identification covers three key issues: high similarity among classes, label mislabeling, and lack of data. On top of that, PEL generates enhanced label distributions that not only contain the target category but also have inter-class similarity awareness, thus being more sensitive to the similarity degree among categories. Moreover, the enhanced labels are more robust to mislabeled samples since the similarity scores can allocate probability to similar labels which usually include the right label, thus the model can still learn useful information from mislabeled samples. With the additional supervision from the prototypical knowledge base, models can reduce overfitting when facing small datasets. Our method has achieved superior performance on 7 UFGVC benchmark datasets compared to 22 competitive methods.

The beneficial performance and excellent efficiency confirm the superiority of PEL in the UFGVC tasks. However, results demonstrate that on datasets lacking training samples, such as CottonCultivar, SoyCultivarLocal, and SoyCultivarAge, the overall performance remains around 60%, and there is still much room for improvement. Therefore, the lack of data is still a crucial challenge in UFGVC tasks. Since UFGVC tasks are often accompanied by the few-shot problem, effective methods to avoid overfitting in UGFVC can be a

key research point in future work.

6. Acknowledgement

References

[1] B. Wang, Y. Gao, X. Yuan, S. Xiong, X. Feng, From species to cultivar: Soybean cultivar recognition using joint leaf image patterns by multiscale sliding chord matching, biosystems engineering 194 (2020) 99–111.

[2] W. Zheng, C. Gou, L. Yan, Forest representation learning with multiscale contour feature learning for leaf cultivar classification, in: 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE, 2019, pp. 774–777.

[3] B. Wang, H. Li, J. You, X. Chen, X. Yuan, X. Feng, Fusing deep learning features of triplet leaf image patterns to boost soybean cultivar identification, Computers and Electronics in Agriculture 197 (2022) 106914.

[4] X. Yu, Y. Zhao, Y. Gao, S. Xiong, Maskcov: A random mask covariance network for ultra-fine-grained visual categorization, Pattern Recognition 119 (2021) 108067.

[5] Y. Zhang, J. Peng, X. Yuan, L. Zhang, D. Zhu, P. Hong, J. Wang, Q. Liu, W. Liu, Mfcis: an automatic leaf-based identification pipeline for plant cultivars using deep learning and persistent homology, Horticulture research 8.

[6] S. H. Lee, C. S. Chan, S. J. Mayo, P. Remagnino, How deep learning extracts and learns leaf features for plant classification, Pattern Recognition 71 (2017) 1–13.

[7] A. Beikmohammadi, K. Faez, A. Motallebi, Swp-leafnet: A novel multistage approach for plant leaf identification based on deep cnn, Expert Systems with Applications 202 (2022) 117470.

[8] W. Zheng, L. Yan, C. Gou, F.-Y. Wang, Fuzzy deep forest with deep contours feature for leaf cultivar classification, IEEE Transactions on Fuzzy Systems.

[9] X. Yu, J. Wang, Y. Zhao, Y. Gao, Mix-vit: Mixing attentive vision transformer for ultra-fine-grained visual categorization, Pattern Recognition 135 (2023) 109131.

[10] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531.

[11] J. Gou, B. Yu, S. J. Maybank, D. Tao, Knowledge distillation: A survey, International Journal of Computer Vision 129 (2021) 1789–1819.

[12] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, G. Hinton, Regularizing neural networks by penalizing confident output distributions, arXiv preprint arXiv:1701.06548.

[13] R. Müller, S. Kornblith, G. E. Hinton, When does label smoothing help?, Advances in neural information processing systems 32.

[14] P. Yang, X. Sun, W. Li, S. Ma, W. Wu, H. Wang, Sgm: sequence generation model for multi-label classification, arXiv preprint arXiv:1806.04822.

[15] J. Snell, K. Swersky, R. Zemel, Prototypical networks for few-shot learning, Advances in neural information processing systems 30.

[16] Z. Ji, X. Chai, Y. Yu, Y. Pang, Z. Zhang, Improved prototypical networks for few-shot learning, Pattern Recognition Letters 140 (2020) 81–87.

[17] J. S. Cope, D. Corney, J. Y. Clark, P. Remagnino, P. Wilkin, Plant species identification using digital morphometrics: A review, Expert Systems with Applications 39 (8) (2012) 7562–7573.

[18] R. Hu, W. Jia, H. Ling, D. Huang, Multiscale distance matrix for fast plant leaf recognition, IEEE transactions on image processing 21 (11) (2012) 4667–4672.

[19] X. Wang, B. Feng, X. Bai, W. Liu, L. J. Latecki, Bag of contour fragments for robust shape classification, Pattern recognition 47 (6) (2014) 2116–2125.

[20] C. Zhao, S. S. Chan, W.-K. Cham, L. Chu, Plant identification using leaf shapes—a pattern counting approach, Pattern Recognition 48 (10) (2015) 3203–3215.

[21] J. S. Cope, P. Remagnino, S. Barman, P. Wilkin, Plant texture classification using gabor co-occurrences, in: International Symposium on Visual Computing, Springer, 2010, pp. 669–677.

[22] Z. Tang, Y. Su, M. J. Er, F. Qi, L. Zhang, J. Zhou, A local binary pattern based texture descriptors for classification of tea leaves, Neurocomputing 168 (2015) 1011–1023.

[23] M. G. Larese, R. Namías, R. M. Craviotto, M. R. Arango, C. Gallo, P. M. Granitto, Automatic classification of legumes using leaf vein image features, Pattern Recognition 47 (1) (2014) 158–168.

[24] J. Charters, Z. Wang, Z. Chi, A. C. Tsoi, D. D. Feng, Eagle: A novel descriptor for identifying plant species using leaf lamina vascular features, in: 2014 IEEE international conference on multimedia and expo workshops (ICMEW), IEEE, 2014, pp. 1–6.

[25] C. Yang, Plant leaf recognition by integrating shape and texture features, Pattern Recognition 112 (2021) 107809.

[26] N. Kumar, P. N. Belhumeur, A. Biswas, D. W. Jacobs, W. J. Kress, I. C. Lopez, J. V. Soares, Leafsnap: A computer vision system for automatic plant species identification, in: European conference on computer vision, Springer, 2012, pp. 502–516.

[27] G. L. Grinblat, L. C. Uzal, M. G. Larese, P. M. Granitto, Deep learning for plant identification using vein morphological patterns, Computers and Electronics in Agriculture 127 (2016) 418–424.

[28] P. Barré, B. C. Stöver, K. F. Müller, V. Steinhage, Leafnet: A computer vision system for automatic plant species identification, Ecological Informatics 40 (2017) 50–56.

[29] M. P. Shah, S. Singha, S. P. Awate, Leaf classification using marginalized shape context and shape+ texture dual-path deep convolutional neural network, in: 2017 IEEE International conference on image processing (ICIP), IEEE, 2017, pp. 860–864.

[30] S. H. Lee, C. S. Chan, P. Remagnino, Multi-organ plant classification based on convolutional and recurrent neural networks, IEEE Transactions on Image Processing 27 (9) (2018) 4287–4301.

[31] X. Yu, Y. Zhao, Y. Gao, S. Xiong, X. Yuan, Patchy image structure classification using multi-orientation region transform, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 34, 2020, pp. 12741–12748.

[32] X. Chen, B. Wang, Y. Gao, Pairwise rotational-difference lbp for fine-grained leaf image retrieval, in: 2022 IEEE International Conference on Image Processing (ICIP), IEEE, 2022, pp. 3346–3350.

[33] X. Chen, B. Wang, Invariant leaf image recognition with histogram of gaussian convolution vectors, Computers and Electronics in Agriculture 178 (2020) 105714.

[34] C. Liu, J. Han, B. Chen, J. Mao, Z. Xue, S. Li, A novel identification method for apple (malus domestica borkh.) cultivars based on a deep convolutional neural network with leaf image input, Symmetry 12 (2) (2020) 217.

[35] H. Tavakoli, P. Alirezazadeh, A. Hedayatipour, A. B. Nasib, N. Landwehr, Leaf image-based classification of some common bean cultivars using dis-

criminative convolutional neural networks, Computers and electronics in agriculture 181 (2021) 105935.

[36] X. Yu, Y. Zhao, Y. Gao, X. Yuan, S. Xiong, Benchmark platform for ultra-fine-grained visual categorization beyond human performance, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 10285–10295.

[37] B. Wang, Y. Gao, Structure integral transform versus radon transform: A 2d mathematical tool for invariant shape recognition, IEEE Transactions on Image Processing 25 (12) (2016) 5635–5648.

[38] B. Wang, Y. Gao, Hierarchical string cuts: a translation, rotation, scale, and mirror invariant descriptor for fast shape retrieval, IEEE Transactions on Image Processing 23 (9) (2014) 4101–4111.

[39] B. Wang, Y. Gao, X. Yuan, S. Xiong, Local r-symmetry co-occurrence: characterising leaf image patterns for identifying cultivars, IEEE/ACM Transactions on Computational Biology and Bioinformatics.

[40] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, Communications of the ACM 60 (6) (2017) 84–90.

[41] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556.

[42] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L.-C. Chen, Mobilenetv2: Inverted residuals and linear bottlenecks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 4510–4520.

[43] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1251–1258.

[44] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, arXiv preprint arXiv:2010.11929.

[45] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, H. Jégou, Training data-efficient image transformers & distillation through attention, in: International conference on machine learning, PMLR, 2021, pp. 10347–10357.

[46] J. He, J.-N. Chen, S. Liu, A. Kortylewski, C. Yang, Y. Bai, C. Wang, Transfg: A transformer architecture for fine-grained recognition, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 36, 2022, pp. 852–860.

[47] Z. Yang, T. Luo, D. Wang, Z. Hu, J. Gao, L. Wang, Learning to navigate for fine-grained classification, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 420–435.

[48] P. Li, J. Xie, Q. Wang, Z. Gao, Towards faster training of global covariance pooling networks by iterative matrix square root normalization, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 947–955.

[49] Y. Chen, Y. Bai, W. Zhang, T. Mei, Destruction and construction learning for fine-grained image recognition, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 5157–5166.

[50] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, Y. Yoo, Cutmix: Regularization strategy to train strong classifiers with localizable features, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 6023–6032.

[51] X. Yu, Y. Zhao, Y. Gao, Spare: Self-supervised part erasing for ultra-fine-grained visual categorization, Pattern Recognition 128 (2022) 108691.

[52] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.

[53] G. Huang, Z. Liu, L. Van Der Maaten, K. Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.