Fine-Grained Recognition in High-throughput Phenotyping

Beichen Lyu Stuart D. Smith Keith A. Cherkauer
Purdue University, West Lafayette, IN
{lvb, smit1770, cherkaue}@purdue.edu

Abstract

Fine-Grained Recognition aims to classify sub-category objects such as bird species and car models from imagery. In High-throughput Phenotyping, the required task is to classify individual plant cultivars to assist plant breeding, which has posed three challenges: 1) it is easy to overfit complex features and models, 2) visual conditions change during and between image collection opportunities, and 3) analysis of thousands of cultivars require high-throughput data collection and analysis. To tackle these challenges, we propose a simple but intuitive descriptor, Radial Object Descriptor, to represent plant cultivar objects based on contour. This descriptor is invariant under scaling, rotation, and translation, as well as robust under changes to the plant's growth stage and camera's view angle. Furthermore, we complement this mid-level feature by fusing it with the low-level features (Histogram of Oriented Gradients) and deep features (ResNet-18), respectively. We extensively test our fusion approaches using two real world experiments. One experiment is on a novel benchmark dataset (HTP-Soy) in which we collect $\sim$ $2,000$ *high-resolution aerial images of outdoor soybean plots. Another experiment is on three datasets of indoor rosette plants. For both experiments, our fusion approaches achieve superior accuracies while maintaining better generalization as compared with traditional approaches.*

1. Introduction

Fine-Grained Recognition (FGR) is the task of classifying sub-category objects. This task is inherently challenging due to the high intra-class variance but low inter-class variance between objects. Recent research in FGR has empowered applications to classify "finer" categories such as car types for traffic surveillance [28], bird species for ecological observation [1], and retail products for automatic checkout [32]. Continuing research in these FGR applications has reinvented recognition techniques such as part discovery [13] and visual attention [6].

In the agronomy and biology community, High-throughput Phenotyping (HTP) is an emerging topic that also studies sub-category objects but has received little attention from the FGR community. HTP studies the phenotype of biological cultivars using high-throughput data collection and analysis. With the adoption of image-based data, HTP has evolved from the traditional research of genotype-phenotype interactions into a modern interdisciplinary framework, which unifies the research of genotype-environment-management (G×E×M) interactions. Recent applications include genetic selection in plant breeding [3] [33], soybean stress evaluation under flooding [16], wheat yield prediction using multi-spectral imaging [10], and phenotype recognition for zebrafish sorting [25]. In this paper, we focus on plant-based HTP.

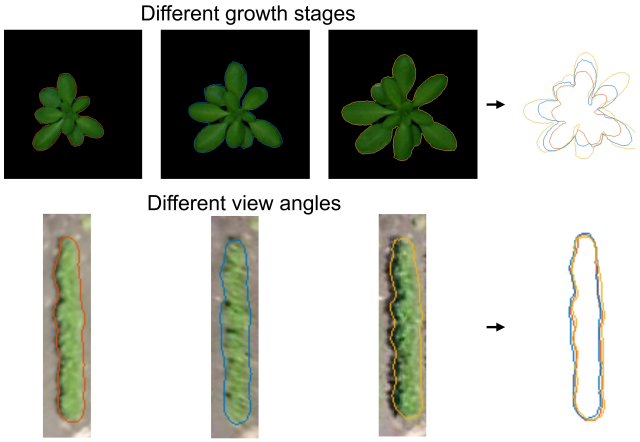

Figure 1: Contour can provide important cues for FGR in HTP. Top and bottom image sets denote top-view variants of an indoor arabidopsis plant and an outdoor soybean plot, respectively.

In the context of HTP, FGR's specific task is to classify massive plant cultivars, which poses three challenges. First, plant cultivars are very similar and usually have a very few number of image replicates collected per class. This can cause easy overfitting for traditional approaches using complex features and models. Second, HTP researchers view

plant as a dynamic system that constantly interacts with the environment. Therefore, plant cultivars are typically observed under changes of growth stage and if planted in the field, changes of camera's view angle using unmanned aerial systems (UAS) images (see Figure 1). Third, HTP image data are collected and analyzed in a high-throughput manner. This requires low cost and complexity when developing FGR approaches, which is particularly critical for HTP researchers with limited computing resources and expertise.

Unfortunately, these challenges have not been well considered in previous related work. To tackle image-based plant classification problems, some works adopt deep features using Convolutional Neural Network (CNN), which may require a large set of annotated images [36] or network architecture redesign to overcome overfitting [18]. In contrast, some works adopt low-level and mid-level feature descriptors such as Border-Interior Pixel Classification for ground object classification [22] and Bag of Visual Words for soybean disease detection [23], which might not be directly applicable for FGR in HTP. As a trade-off, works in other domains such as osteoporosis diagnoses [29] and Presentation Attack Detection [19] adopt the fusion approaches using low-level, med-level, and deep features. Most of the approaches aforementioned need to be supervised by annotated datasets. However, publicly available plant datasets are mostly limited to either category-level recognition using organs (*e.g.* [2] [17]) or phenotypic trait study using indoor plants (*e.g.* [9] [14]), which hinders improvements of FGR in more general HTP settings.

Considering the challenges and previous works presented above, we propose two FGR approaches (`ROD-HOG-Softmax` and `ROD-ResNet-Softmax`) using feature fusion and extensively test them against multiple competitor approaches and HTP datasets, which includes a novel aerial image set of soybean plots in the field (`HTP-Soy`). Both FGR approaches are based on a simple but invariant feature descriptor, Radial Object Descriptor (ROD), which is inspired from the observation of contour to discriminate HTP objects shown in Figure 1. We further fuse it with the low-level features, Histogram of Oriented Gradients (HOG), as well as the deep features extracted from ResNet-18. Both fused features are re-trained in a multi-class classification model, Softmax regression. Experiment results indicate that our fusion approaches outperform the traditional approaches of low-level features or single deep features, even under the changes of growth stage and camera's view angle.

In summary, this paper illustrates three contributions to FGR in HTP:

1. Applies FGR into general HTP settings by introducing a novel benchmark dataset (`HTP-Soy`) for soybean plot recognition in UAS images.
2. Proposes a simple but intuitive feature descriptor, ROD, that is robust under scaling, rotation, and translation, as well as temporal and viewpoint changes.
3. Demonstrates the effectiveness of fusing ROD with HOG and deep features (`ROD-HOG-Softmax` and `ROD-ResNet-Softmax`) by testing them against multiple competitor approaches and HTP datasets.

The rest of paper is organized as follows: Section 2 details procedures to extract ROD along with invariance proof; section 3 describes the feature fusions in Softmax regression; section 4 tests the proposed approaches via two experiments; and section 5 concludes with outlook.

2. Extracting Radial Object Descriptor

In this section, we introduce the whole workflow to extract ROD. Along with the description of pre-processing and post-processing steps, we highlight the step of contour unfolding. ROD's invariance is also proved at the end.

2.1. Pre-processing

The pre-processing step consists of three substeps: object localization, object segmentation, and contour extraction, which are illustrated in Figure 2. For the first substep of object localization, we locate HTP objects in UAS images directly based on prior knowledge such as sensor fusion and experiment set-up. Here we use the method developed by [12] and [16], which basically averages the orthomosaic image along the horizontal and vertical axes and then locates soybean plots based on the presence of brightness peaks. Further details may be found in [16]. In the second substep, we segment the object using either brightness thresholds based on greenness or Otsu's thresholding [21], or parametric methods such as GrabCut [24] and CNN [27]. Noisy canopies can be further filtered out based on the sizes of Connected Components. The third substep is to extract the object's contour as an array of contour pixel coordinates, E, where we can use the Moore-Neighbor Tracing algorithm [8].

2.2. Contour Unfolding

We further unfold the 2-dimensional contour into 1-dimensional ROD, as shown in Figure 3. ROD, *i.e.* the blue curve in drawing b of Figure 3, is an array of values and the array's length is the same as the number of pixels on the contour. Each value in ROD indicates the shortest distance from each pixel to the seed. The seed, *i.e.* the gray center line in drawing a of Figure 3, is the approximate center of biological objects of radial shape. This is because such biological objects typically grow radially outwards from the seed. For example, a cell or rosette plant grows with respect to their center point while a leaf or crop plot grows with respect to their center line of symmetry.

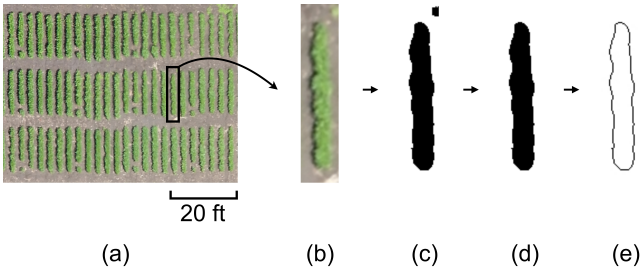

Figure 2: (a) An orthomosaic image of soybean plots with different genetics, (b) a soybean plot, (c) soybean plot with green canopy segmented, (d) soybean plot without outlier canopy, and (e) contour of soybean plot.

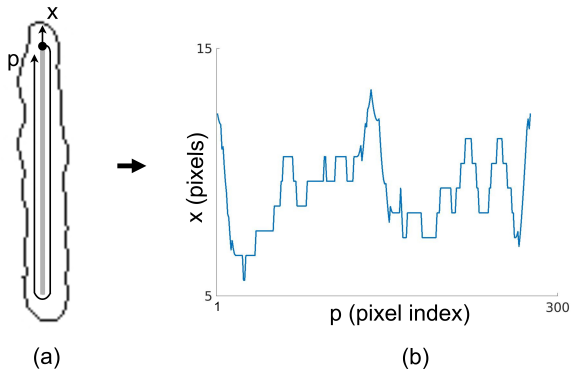

Figure 3: A demonstration of unfolding a contour (black outline in (a)) into ROD (blue curve in (b)) based on the seed (gray center line in (a)).

Details of the contour unfolding step are formulated in Algorithm 1. Having the contour E extracted from the last step, we first compute the center point c by averaging pixels on the contour. If the seed is a single dot, then we directly compute each pixel's Euclidean distance to the seed as ROD x. Otherwise, we assume the seed is a line. For example, soybean seeds are initially sowed along a line so we compute the seed as a center line denoted by S. Then, we approximate the length of the seed, l, based on the four locations of the topmost, bottommost, leftmost, and rightmost pixel. This approximation of length ensures a roughly even distribution of distances between the pixels and seed. However, this approximation of seed length should be properly adjusted based on the context and the object's geometry. Now we can compute ROD x, and each of its value denotes the shortest Euclidean distance between the seed S and a pixel e on contour E. For a fast approximation of ROD when processing a large number of contours, we can assume that the contour consists of a rectangle in the middle and two half circles at two ends. Then for each pixel e, we can directly find the corresponding seed s that has the shortest distance to e. Note that x and E will have the same pixel indexing p.

Algorithm 1: ContourUnfolding $(E, isDot)$

Input : contour pixel coordinates $E \in \mathbb{R}^{2\times \#\text{ of pixels}}$
seed shape indicator $isDot \in \{\text{true}, \text{false}\}$
Output: ROD $x \in \mathbb{R}^{\#\text{ of pixels}}$

1 $c := \frac{\sum_{p=1}^{\#\text{ of pixels}} E_{:,p}}{\#\text{ of pixels}}$
2 **if** $isDot$ **then**
3 $\quad x := ||E - c||_2$
4 **else**
5 $\quad l := (E_{1,\max} - E_{1,\min}) - (E_{2,\max} - E_{2,\min})$
6 $\quad S := \{(i, c_2)\}, i := c_1 - \frac{l}{2}...c_1 + \frac{l}{2}$
7 $\quad x := \{||E_{:,p} - s||_2 \mid s := \text{argmin}_{s\in S} ||E_{:,p} - s||_2\},$
8 $\quad\quad p := 1...\#\text{ of pixels}$
9 **end**
10 $x := \frac{x - x_{\min}}{x_{\max} - x_{\min}}$

Unfolding the contour into ROD as the final feature is important as it brings two benefits. First, the dimensionality of the feature representation has been significantly reduced from 2-dimension to 1-dimension. Besides, we can still restore the 2-dimensional contour from the 1-dimensional ROD provided that we keep the seed and angular information. This ability to compress and restore the feature may greatly enhance the portability of sensing functions on small mobile devices such as smart glasses and UAS [26].

2.3. Post-processing

Before fitting ROD into classifiers, we can further post-process it to enhance consistency. For example, we can normalize values in ROD into the range of $[0, 1]$ as shown in line 9 of Algorithm 1. Also, we can scale RODs of different lengths into ones of a fixed length by sampling or interpolation. For plants in non-convex shape such as these in section 4.4, we will reorder ROD using polar coordinates to counteract the non-uniform unfolding.

2.4. Invariance Proof

Below we prove that the newly proposed feature descriptor, ROD, is invariant under scaling, rotation, and translation.

Theorem 1. *Radial Object Descriptor is invariant under uniform scaling, rotation, and translation.*

Proof. Based on Algorithm 1, this theorem can be formulated as

$$\frac{x_i - x_{\min}}{x_{\max} - x_{\min}} = \frac{x'_i - x'_{\min}}{x'_{\max} - x'_{\min}}, \forall i = 1, 2...|x|$$

where x and x' respectively denote the ROD before and after transformation.

The rest of proof is to show the normalized x's invariance, which can be intuitively perceived since the normalized x indeed denotes the relative Euclidean distance between any two points on an object. The full proof is available in the Supplemental Materials.

□

Furthermore, ROD is robust under changes of plant growth stage and camera view angle, which will be experimentally demonstrated in section 4. The change of the camera's view angle will be reflected as the stretch of value magnitude and density in ROD. This stretch can be mostly counteracted as we include the Softmax regression with parametric learning. Also, if each contour pixel of a plant grows perfectly outwards from the seed following a straight trajectory at a constant rate, then change of growth stage is equivalent to uniform scaling, which has been proved invariant.

3. Classification with Feature Fusion

In this section, we start with an introduction of the HOG and ResNet-18, and then describe their fusion with ROD in the Softmax regression (`ROD-HOG-Softmax` and `ROD-ResNet-Softmax`).

3.1. Histogram of Oriented Gradients

To complement the mid-level feature ROD, we fuse it with another low-level local feature descriptor, HOG, which was initially used by [5] for pedestrian detection. As its name implies, HOG contains histogram of local gradients at different orientations, which is powerful for differentiating objects with significant texture patterns. Specifically, an image is gridded into cells containing a specific number of pixels and in each cell, we compute gradients of each pixel and summarize these gradients based on their magnitude and orientation in a histogram. Histograms of neighbor cells are further grouped into groups of blocks with a ratio of overlapping to improve local contrast. Recent applications of HOG in FGR [13] [31] [35] have also shown the effectiveness of HOG, particularly in its ability to detect discriminative parts between similar objects as in our FGR case.

3.2. ResNet-18

Another way to complement ROD is to add deep features from ResNet-18, *i.e*. activations of the last pooling layer. ResNet-18 is a 18-layer Residual Neural Network (ResNet) proposed by [11], which is a type of CNN with residual parts to skip layers so that higher accuracy can be efficiently achieved using deeper layers. We use ResNet as the representative of the deep feature extraction methods because its performance has been demonstrated in plant related research such as disease recognition [7], fruit detection [36] and weed classification [20]. The ResNet of 18 layers will suffice for our dataset as each subclass has a very limited number of replicates, *i.e*. $6-14$.

3.3. Softmax Regression

Softmax regression is a classification model that accepts a matrix of real values as input in the Softmax function and after some learning with respect to a loss function and a set of weight parameters, predicts a set of categorical values as output. Specifically, our input will be the feature matrix $X \in \mathbb{R}^{m\times n}$, label vector $y \in \{y_i\}^m, y_i \in V$, and a set of unique labels $V \in \{v_i\}^k, i = 1...k$. Note that m is the number of examples, n is the number of features (or equivalently, the total number of values in $x^{(\text{ROD})}$ and $x^{(\text{HOG})}$ or $x^{(\text{ResNet})}$), and k is the number of alternative labels. Each row in X is a concatenation of corresponding ROD and HOG features, *i.e*. $X_{i,:} = [x^{(\text{ROD})}, x^{(\text{HOG})}]$ or $[x^{(\text{ROD})}, x^{(\text{ResNet})}]$.

To train a Softmax regression, we need to learn the set of weight parameters $W \in \mathbb{R}^{n\times k}$ while minimizing a loss function. Here we use the cross entropy as the loss function and thus, the regression model can be formulated as

$$W = \underset{W\in\mathbb{R}^{n\times k}}{\operatorname{argmin}} -\sum_{i=1}^{m}\sum_{j=1}^{k} \mathbb{1}[y_i = v_j] \log \frac{\exp(X_{i,:}W_{:,j})}{\sum_{l=1}^{k}\exp(X_{i,:}W_{:,l})} \tag{1}$$

A sample set of features and filter heatmaps of ROD and HOG is available in the Supplemental Materials.

4. Experiments

In this section, we evaluate the proposed approach against other approaches from the literature via two experiments, which aim to classify plant cultivars (*i.e*. subclasses as in table 1) in different HTP settings. The first experiment looks at soybean plots grown in the field using UAS images where we highlight our approach's robustness under changing camera view angle. The second experiment looks at rosette plants in the greenhouse using a stationary camera where we highlight our approach's robustness versus change of growth stage.

We select soybean, arabidopsis, bean, and komatsuna as our experimental targets since they are important plant types for plant phenotypic research. For example, arabidopsis is a classic model plant with well-sequenced genome for plant biology research, komatsuna is a popular leafy vegetable in Asia since it is insect-resistant and grows very fast [30], and soybean is a critical crop type for global food production.

	Set-up			Image specifications				Evaluation per subclass		
	Setting	Collection method	View angle	image #	class #	subclass # per class	replicate # per subclass	rep. # (train)	rep. # (val.)	rep. # (test)
HTP-Soy	outdoor field	UAS with RGB camera	top-view with changing angles	1728	96	3	6	3	1	2
Arabidopsis [18]	indoor green-house	stationary RGB camera	top-view with fixed angle	2134	97	5	6 (overlapped)	3	1	2
Bean [4]				350	5	5	14	8	1	5
Komatsuna [30]				210	5	5	9 (overlapped)	5	1	3

Table 1: Dataset specifications. Each class is split into subclasses by growth stage and each subclass consists of replicates that differ by view angle or collection time.

4.1. Datasets

The dataset (HTP-Soy) used in experiment 1 was collected using a UAS over an outdoor soybean field. The soybean field was approximately 5 ha in size with plots planted in straight lines with consistent spacing by tractor and each neighborhood of soybean plots shared the same genetics. The UAS was an eBee from senseFly which flew over the field at a height of 120 m with a senseFly S.O.D.A. RGB camera on board. This resulted in a leaf-level image resolution of 2.5 cm. We flew the UAS with a forward and side overlap ratio of over 85% and 75% so that each soybean plot could get multiple replicates from different view angles. As mentioned in section 2.1, we located each soybean plot using a mix of sensors including the GPS mounted on the UAS and ground control points (GCPs) on the ground. The GCPs were control points that had their positions accurately measured by Real-time Kinematic (RTK) and had reflectors installed to be visible from UAS images.

Details of our dataset are tabulated in Table 1 and sample images of soybean plots are also presented in Figure 4. There are 1728 images used in experiment 1 which cover 96 soybean plots (or classes) in 3 growth stages (or subclasses), respectively on July 2, July 6, and July 12 in 2018. In each growth stage, we view each soybean plot from 6 view angles (or replicates), which can be indirectly seen from their changing shadows.

The three datasets (Arabidopsis, Bean, Komatsuna) used in experiment 2 were from stationary cameras over rosette plants for indoor phenotypic study. They were publicly released in [18], [4], and [30], respectively. Plants with different genetics (or classes) were planted in a substrate in which environmental conditions such as soil and lighting were strictly controlled. Stationary cameras were installed above plants to take top-view images at a regular rate, which resulted in images from multiple collection times (or replicates). Since only one image is taken per time, we group neighbor image replicates taken during a period into different growth stages (or subclasses).

Note that canopy are segmented in all images used in the two experiments. It is important to filter out noisy backgrounds so that only features related to plant phenotypes themselves are used for recognition evaluation. We roughly segment the canopy in each RGB image I by using the inequality relationship, $I_{:,:,2} > (I_{:,:,1} + k) \cap I_{:,:,2} > (I_{:,:,3} + k)$ [12]. For each of the three growth stages in HTP-Soy, we use $k \approx 0, 10, 20$ respectively. For Arabidopsis, Bean, and Komatsuna, we use $k \approx 0, 20, 25$, respectively.

4.2. Comparison with Other Approaches

We evaluate the robustness of ROD as well as its fusion with HOG and ResNet in Softmax using seven different approaches, whose configurations are tabulated in Table 2. All approaches use cross entropy as their loss function.

	Feature	Model	Optimizer
CNN	image	CNN	Adam
ResNet-18		ResNet-18	
Fourier-Softmax	Fourier	Softmax	Scaled Conjugate Gradient
ROD-Softmax	ROD		
HOG-Softmax	HOG		
ROD-HOG-Softmax	ROD & HOG		
ROD-ResNet-Softmax	ROD & ResNet-18		

Table 2: Configurations of competitor approaches.

The first two approaches, CNN and ResNet-18, are based on CNN. CNN is a vanilla version of CNN which has an architecture of two convolutional layers and one fully connected layer. Each convolution layer is followed by a batch normalization layer, a ReLu layer, and a max

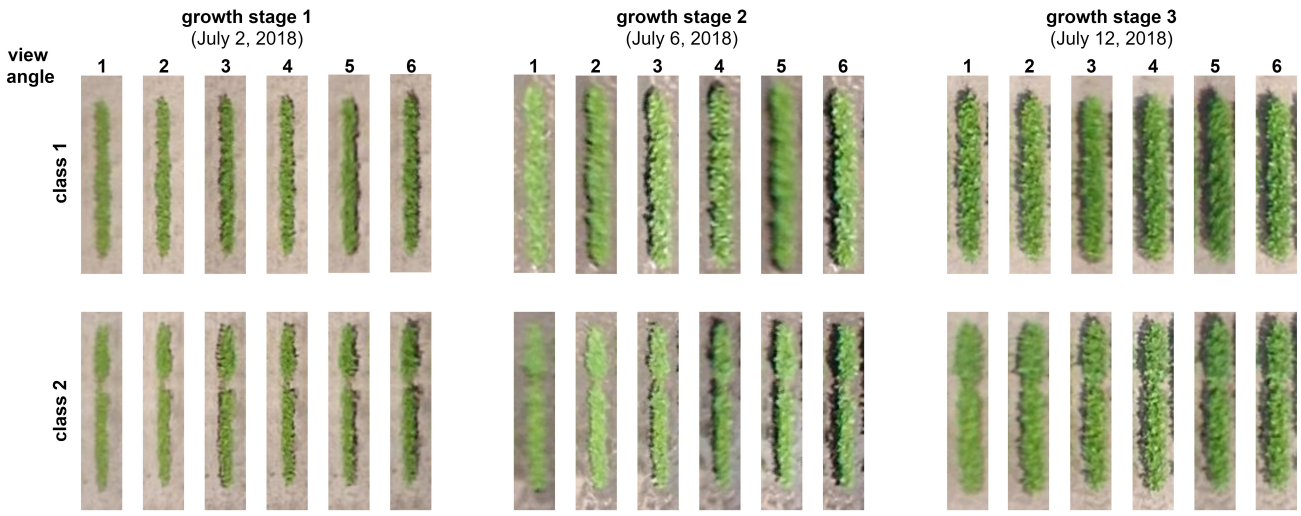

Figure 4: Sample data in `HTP-Soy`. For the change of view angle, observe shadows on the soybean plot outlines.

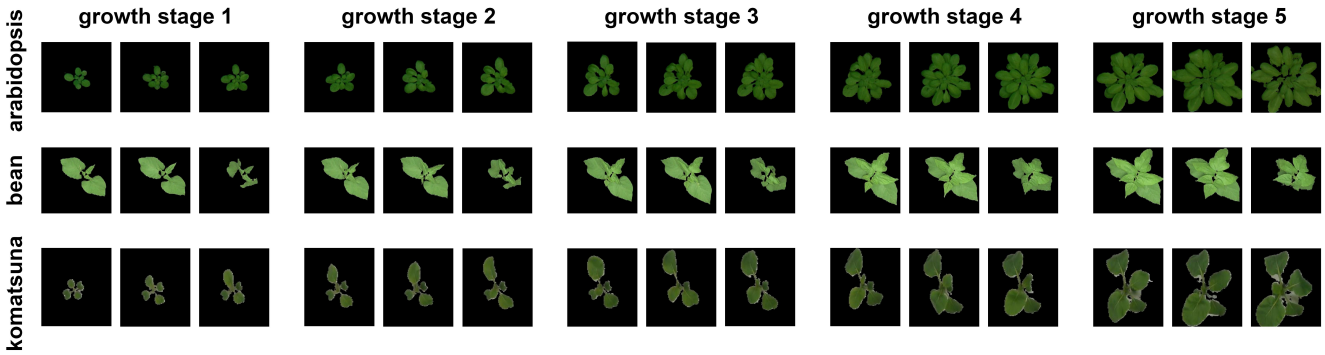

Figure 5: Sample data in `Arabidopsis` [18], `Bean` [4], and `Komatsuna` [30].

pooling layer. The first and second convolution layers respectively have 3 and 48 filters of size $(3, 3)$, and both are padded to ensure size consistency. The max pooling layers have a size $(2, 2)$ for both pooling and stride. For details of `ResNet-18`, see section 3.2. For both `CNN` and `ResNet-18`, we manually tune the mini-batch size and learning rate based on each validation dataset before finalizing its classification result. `CNN` is trained using a GeForce RTX 2080 Ti GPU with 16 GB RAM while `ResNet-18` is trained using a NVIDIA T4 Tensor Core GPU with 16 GB RAM. Training is stopped when validation accuracy starts to decay. `ResNet-18` is pretrained on ImageNet.

The remaining four approaches (`Fourier-Softmax`, `ROD-Softmax`, `HOG-Softmax`, `ROD-ResNet-Softmax`) are used to evaluate our methods (`ROD-HOG-Softmax` and `ROD-ResNet-Softmax`) against algorithms with different selections of features. `Fourier-Softmax` and `ROD-Softmax` will be compared to evaluate ROD's robustness since Fourier feature descriptor [34] and ROD are very similar as contour-based mid-level descriptors with proved invariance. To be fair, the size of Fourier vectors will be set equal to that of corresponding ROD in each dataset. In addition, `ROD-Softmax`, `HOG-Softmax`, `ROD-HOG-Softmax`, and `ROD-ResNet-Softmax` will be compared to show how the fusion of ROD with HOG and ResNet will improve classification results. All HOGs will be computed with cell size of $(5, 5)$, block size of $(2, 2)$, number of overlapping cells $(1, 1)$, and 9 orientation bins in histogram. All five models will be trained using the Scaled Conjugate Gradient method and will be terminated when either the loss reaches zero or gradient reaches 10^{-6}. The four Softmax-based approaches are run using an Intel Xeon 2.80 GHz CPU and 16 GB RAM.

We take three measurements to minimize bias when evaluating each approach. First, we evaluate all accuracy re-

sults using the mean accuracy (mA). For all validation mAs (except CNN and ResNet-18), we use k-fold cross validation where k is the number of replicates in the training set. Second, instead of retraining model using both training and validation sets to report testing accuracy, we directly use the model trained with the best validation accuracy to report the testing accuracy. This is important as each subclass has a very limited number of replicates and using more training data to report testing accuracy can favor non-CNN approaches. Third, we randomize the order of classes as well as the order of replicates within.

4.3. Experiment 1: Classifying Soybean Plots

Using the HTP-Soy dataset, we first evaluate competitor approaches by classifying soybean plots using images of the same growth stage. That is, we train and test approaches using each of the three datasets collected on July 2, July 6, and July 12 and combine classification results from these three datasets in Table 3. Validation mA and test mA are averaged over all three datasets.

As shown in Table 3, ROD-ResNet-Softmax achieves the highest validation mA of 0.947 while ROD-HOG-Softmax achieves the highest test mA of 0.866, which slightly outperforms ROD-ResNet-Softmax. It is not easy to get such high mAs as we use a very limited number of replicates per subclass for training and each replicate differs by camera's view angle. Also, ROD-Softmax significantly outperforms Fourier-Softmax, which indicates the advantage of ROD over Fourier as a contour-based mid-level descriptor. It is even surprising to see that ROD-Softmax achieves mA comparable to HOG-Softmax even though HOG contains much richer information.

	Validation mA	Test mA
CNN	0.667	0.594
ResNet-18	0.830	0.816
Fourier-Softmax	0.218	0.295
ROD-Softmax	0.747	0.769
HOG-Softmax	0.779	0.785
ROD-HOG-Softmax	0.865	**0.866**
ROD-ResNet-Softmax	**0.947**	0.857

Table 3: Classification results of experiment 1 using HTP-Soy data from the same growth stage.

We further compare the generalization ability of ResNet-18 and ROD-HOG-Softmax to classify soybean plots using the model trained from a different growth stage. This is an area for HTP engineers to improve object localization accuracy using older (*i.e.* later growth stage) image datasets. Also, this can help plant breeders to discover plant phenotypic traits shared between growth stages.

As Figure 6 shows, ROD-HOG-Softmax outperforms ResNet-18 in inferring soybean plots for most pairs of growth stages. ROD-HOG-Softmax's advantage becomes more obvious during latter growth stages (*i.e.* 2 and 3) where soybean canopy grows faster and its contour becomes more discriminative. This may also imply that ResNet-18 tend to overfit features that are not robust under change of growth stage.

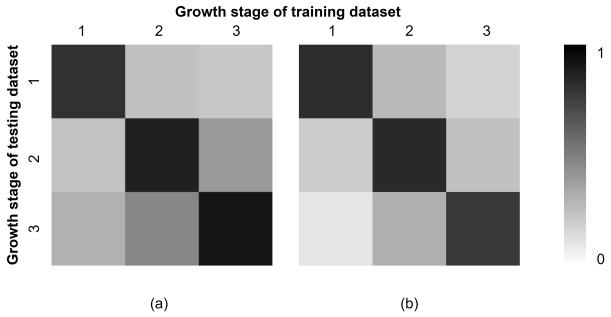

Figure 6: Test mA of experiment 1 by using combinations of training and testing datasets from different growth stages via (a) ROD-HOG-Softmax and (b) ResNet-18.

4.4. Experiment 2: Classifying Rosette Plants

In this experiment, we switch recognition targets from soybean plots with multiple plants to individual rosette plants using the datasets of Arabidopsis, Bean, and Komatsuna. Unless specified, all other set-up processes are the same as those in experiment 1.

As shown in Table 4, ROD-ResNet-Softmax outperforms almost all other approaches in terms of both validation mA and test mA (*i.e.* 0.92-1.00) for almost all plants. Similar to the classification results in experiment 1, ROD-Softmax significantly outperforms Fourier-Softmax as well as achieves classification accuracy comparable to HOG-Softmax. The exception is that for arabidopsis, ROD-Softmax's mAs (0.499 and 0.517) are far below those of HOG-Softmax (0.919 and 0.924). This may be caused by the dramatic altering of arabidopsis' leaf positions as well as self-occlusions so that its contour becomes less discriminative [2].

The paper [18] that proposed the original Arabidopsis dataset described a novel approach of CNN-LSTM which achieved an accuracy of 0.93. However, images used in their dataset did not have canopy segmented so that plant's background such as substrate box and soil texture could implicitly boost learning. In contrast, by using the same original dataset, our ROD-HOG-Softmax approach can achieve a validation mA of 0.990 and test mA of 0.992.

	Arabidopsis [18]		Bean [4]		Komatsuna [30]	
	Validation mA	Test mA	Validation mA	Test mA	Validation mA	Test mA
CNN	0.858	0.844	**1.000**	0.992	**1.000**	**1.000**
ResNet-18	0.922	0.912	**1.000**	0.984	**1.000**	**1.000**
Fourier-Softmax	0.152	0.153	0.382	0.392	0.313	0.213
ROD-Softmax	0.499	0.517	0.911	0.880	0.953	**1.000**
HOG-Softmax	0.919	0.924	**1.000**	**1.000**	**1.000**	**1.000**
ROD-HOG-Softmax	0.921	0.917	0.996	0.992	**1.000**	**1.000**
ROD-ResNet-Softmax	**0.974**	**0.927**	**1.000**	0.968	**1.000**	**1.000**

Table 4: Classification results of experiment 2 using data from the same growth stage.

Similar to experiment 1, ROD-HOG-Softmax outperforms ResNet-18 in inferring all three types of plants. As Table 7 shows, ROD-HOG-Softmax achieves very good testing mA particularly for adjacent pairs of growth stages. When inferring bean plants, ROD-HOG-Softmax even achieves nearly perfect test mAs across all pairs of growth stages.

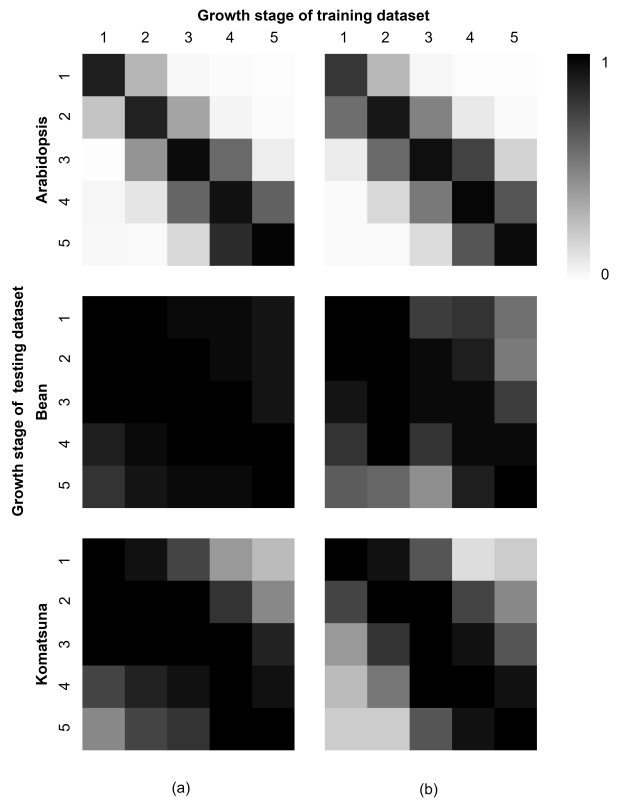

Figure 7: Test mA of experiment 2 by using combinations of training and testing datasets from different growth stages via (a) ROD-HOG-Softmax and (b) ResNet-18.

5. Conclusion

In this paper, we apply FGR in the novel domain of HTP and advance its future study by introducing a benchmark dataset (HTP-Soy) for soybean plot recognition using UAS images. We further propose a simple and robust feature descriptor (ROD) based on contour and fuse it with HOG and deep features in Softmax, which achieve superior accuracies for plant cultivar classification when trained at the different and same growth stages, respectively. For future work, we would like to apply fusion approaches to guide the discovery of novel phenotypic traits in plant breeding, as well as improve the geo-spatial accuracy of crop plot extraction in high-resolution UAS images for different architectural crops.

All datasets are available in the Supplemental Materials submission. HTP-Soy is also archived at the Purdue University Research Repository [15]: https://doi.org/10.4231/ZAD3-MG98.

Acknowledgement

We thank Dr. Katy M. Rainey's group at Purdue University for providing the soybean experiment site. We also thank Department of Computer Science at Purdue University for providing the computing resources. Part of this research is supported by USDA Agriculture and Food Research Initiative Award 2016-07982.

References

[1] Thomas Berg, Jiongxin Liu, Seung Woo Lee, Michelle L. Alexander, David W. Jacobs, and Peter N. Belhumeur. Birdsnap: Large-scale fine-grained visual categorization of birds. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2019–2026, 2014.

[2] Sruti Das Choudhury, Ashok Samal, and Tala Awada. Leveraging image analysis for high-throughput plant phenotyping. *Frontiers in Plant Science*, 10, 2019.

[3] Jared Crain, Suchismita Mondal, Jessica Rutkoski, Ravi P. Singh, and Jesse Poland. Combining high-throughput phenotyping and genomic information to increase prediction and

selection accuracy in wheat breeding. *Plant Genome*, 1(1), 2018.

[4] Jeffrey A. Cruz, Xi Yin, Xiaoming Liu, Saif M. Imran, Daniel D. Morris, David M. Kramer, and Jin Chen. Multi-modality imagery database for plant phenotyping. *Machine Vision and Applications*, pages 735–749, 2016.

[5] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:886–893, 2005.

[6] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4476–4484, 2017.

[7] Alvaro Fuentes, Sook Yoon, Sang Cheol Kim, and Dong Sun Park. A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. *Sensors*, 2017.

[8] Abeer George Ghuneim. Contour tracing. http://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_George_Ghuneim/index.html, 2000. Accessed on July 24, 2019.

[9] Hervé Goëau, Pierre Bonnet, and Alexis Joly. Plant identification in an open-world (lifeclef 2016). *LifeCLEF*, 2016.

[10] Muhammad Adeel Hassana, Mengjiao Yanga, Awais Rasheeda, Guijun Yangc, Matthew Reynoldsd, Xianchun Xiaa, Yonggui Xiaoa, and Zhonghu He. A rapid monitoring of ndvi across the wheat growth cycle for grain yield prediction using a multi-spectral uav platform. *Plant Science*, 282:95–103, 2019.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[12] Anthony A. Hearst, Keith A. Cherkauer, and Katy M. Rainey. Multilayer uas image ortho-mosaics for field-based high-throughput phenotyping. *Plant Phenome Journal*, In Review.

[13] Jonathan Krause, Timnit Gebru, Jia Deng, Li-Jia Li, and Fei-Fei Li. Learning features and parts for fine-grained recognition. *International Conference on Pattern Recognition*, pages 26–33, 2014.

[14] Neeraj Kumar, Peter N. Belhumeur, Arijit Biswas, David W. Jacobs, W. John Kress, Ida Lopez, and Joao V. B. Soares. Leafsnap: A computer vision system for automatic plant species identification. *European Conference on Computer Vision*, pages 502–516, 2012.

[15] Beichen Lyu, Stuart D. Smith, Keith A. Cherkauer, and Katy M. Rainey. Htp-soy: An aerial image set of multi-category soybean for high-throughput phenotyping (htp). *Purdue University Research Repository*.

[16] Beichen Lyu, Stuart D. Smith, Yexiang Xue, and Keith A. Cherkauer. Deriving vegetation indices from high-throughput images by using unmanned aerial systems in soybean breeding. *ASABE Annual International Meeting*, 2019.

[17] Massimo Minervini, Andreas Fischbachb, Hanno Scharr, and Sotirios A. Tsaftaris. Finely-grained annotated datasets for image-based plant phenotyping. *Pattern Recognition Letters*, 81:80–89, 2016.

[18] Sarah Taghavi Namin, Mohammad Esmaeilzadeh, Mohammad Najafi, Tim B. Brown, and Justin O. Borevitz. Deep phenotyping: deep learning for temporal phenotype/genotype classification. *Plant Methods*, 14(1):66, 2018.

[19] Dat T. Nguyen, Tuyen D. Pham, Na R. Baek, and Kang R. Park. Combining deep and handcrafted image features for presentation attack detection in face recognition systems using visible-light camera sensors. *Sensors*, 18:699, 2018.

[20] Alex Olsen, Dmitry A. Konovalov, Bronson Philippa, Peter Ridd, Jake C. Wood, Jamie Johns, Wesley Banks, Benjamin Girgenti, Owen Kenny, James Whinney, Brendan Calvert, Mostafa Rahimi Azghadi, and Ronald D. White. Deepweeds: A multiclass weed species image dataset for deep learning. *Scientific Reports*, 9(1):2058, 2019.

[21] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

[22] Otavio A. B. Penatti, Keiller Nogueira, and Jefersson A. dos Santos. Do deep features generalize from everyday objects to remote sensing and aerial scenes domains? *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 44–51, 2015.

[23] Rillian Diello Lucas Pires, Diogo Nunes Gonçalves, Jonatan Patrick Margarido Orue, Wesley Eiji Sanches Kanashiro, Jose F. Rodrigues Jr., Bruno Brandoli Machado, and Wesley Nunes Gonçalves. Local descriptors for soybean disease recognition. *Computers and Electronics in Agriculture*, 125:48–55, 2016.

[24] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics*, 23(3):309–314, 2004.

[25] Mark Schutera, Thomas Dickmeis, Marina Mione, Ravindra Peravali, Daniel Marcatoa, Markus Reischl, Ralf Mikut, and Christian Pylatiuk. Automated phenotype pattern recognition of zebrafish for high-throughput screening. *Bioengineered*, 7(4):261–265, 2016.

[26] Jonghoon Seo, Seungho Chae, Jinwook Shim, Dongchul Kim, Cheolho Cheong, and Tack-Don Han. Fast contour-tracing algorithm based on a pixel-following method for image sensors. *Sensors*, 16(3):353, 2016.

[27] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4):640–651, 2017.

[28] Jakub Sochor, Jakub Spanhel, and Adam Herou. Boxcars: Improving fine-grained recognition of vehicles using 3-d bounding boxes in traffic surveillance. *IEEE Transactions on Intelligent Transportation Systems*, 20(1):97–108, 2019.

[29] Ran Su, Tianling Liu, Changming Sun, Qiangguo Jin, Rachid Jennane, and Leyi Wei. Fusing convolutional neural network features with hand-crafted features for osteoporosis diagnoses. *Neurocomputing*, 385:300–309, 2020.

[30] Hideaki Uchiyama, Shunsuke Sakurai, Masashi Mishima, Daisaku Arita, Takashi Okayasu, Atsushi Shimada, and Rin

ichiro Taniguchi. An easy-to-setup 3d phenotyping platform for komatsuna dataset. *IEEE International Conference on Computer Vision*, pages 2038–2045, 2017.

[31] Yaming Wang, Jonghyun Choi, Vlad I. Morariu, and Larry S. Davi. Mining discriminative triplets of patches for fine-grained classification. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1163–1172, 2016.

[32] Xiu-Shen Wei, Quan Cui, Lei Yang, Peng Wang, and Lingqiao Liu. Rpc: A large-scale retail product checkout dataset. *arXiv:1901.07249*, 2019.

[33] Alencar Xavier, Benjamin Hall, Anthony A. Hearst, Keith A. Cherkauer, and Katy M. Rainey. Genetic architecture of phenomic-enabled canopy coverage in glycine max. *Genetics*, 206:1081–1089, 2017.

[34] Charles Zahn and Ralph Z. Roskies. Fourier descriptors for plane closed curves. *IEEE Transactions on Computers*, (3):269–281, 1972.

[35] Han Zhang, Tao Xu, Mohamed Elhoseiny, Xiaolei Huang, Shaoting Zhang, Ahmed Elgammal, and Dimitris Metaxas. Spda-cnn: Unifying semantic part detection and abstraction for fine-grained recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1143–1152, 2016.

[36] Yang-Yang Zheng, Jian-Lei Kong, Xue-Bo Jin, Xiao-Yi Wang, Ting-Li Su, and Min Zuo. Cropdeep: the crop vision dataset for deep-learning-based classification and detection in pecision agriculture. *Sensors*, 19(5):1058, 2019.