Pattern Recognition ■ (■■■■) ■■■–■■■

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

On developing and enhancing plant-level disease rating systems in real fields

Yousef Atoum [a], Muhammad Jamal Afridi [b], Xiaoming Liu [b,*], J. Mitchell McGrath [c], Linda E. Hanson [c]

[a] Department of Electrical and Computer Engineering, Michigan State University, United States
[b] Department of Computer Science and Engineering, Michigan State University, United States
[c] ARS Sugar Beet and Bean Research Unit, U.S. Department of Agriculture, United States

ARTICLE INFO

Article history:
Received 17 March 2015
Received in revised form
22 September 2015
Accepted 28 November 2015

Keywords:
CLS Rater
Histogram of Importance (HoI)
Bag of Words (BoW)
Local Binary Patterns (LBP)
Superpixels

ABSTRACT

Cercospora leaf spot (CLS) is one of the most serious diseases of sugar beet worldwide, and if uncontrolled, causes nearly complete defoliation and loss of revenue for beet growers. The beet sugar industry continuously seeks CLS-resistant sugar beet cultivars as one strategy to combat this disease. Normally human experts manually observe and rate the resistance of a large variety of sugar beet plants over a period of a few months. Unfortunately, this procedure is laborious and the labels vary from one expert to another resulting in disagreements on the level of resistance. Therefore, we propose a novel computer vision system, CLS Rater, to automatically and accurately rate plant images in the real field to the "USDA scale" of 0–10. Given a set of plant images captured by a tractor-mounted camera, CLS Rater extracts multi-scale superpixels, where in each scale a novel Histogram of Importances feature encodes both the within-superpixel local and across-superpixel global appearance variations. These features at different superpixel scales are then fused for learning a regressor that estimates the rating for each plant image. We further address the issue of the noisy labels by experts in the field, and propose a method to enhance the performance of the CLS Rater by automatically calibrating the experts ratings to ensure consistency. We test our system on the field data collected from two years over a two-month period for each year, under different lighting and weather conditions. Experimental results show that both the CLS Rater and the enhanced CLS Rater to be highly consistent with the rating errors of 0.65 and 0.59 respectively, which demonstrates a higher consistency than the rating standard deviation of 1.31 by human experts.

1. Introduction

More than 50% of the total U.S. sugar production is from sugar beets [1]. However, disease affects the productivity of sugar beet, and Cercospora leaf spot (CLS) is one of the more serious diseases in that it infects healthy leaves, causes a toxin-mediated necrosis of leaf tissues that disrupts photosynthesis, and ultimately leads to both real sucrose loss and unrealized income for growers. This disease accounts for a significant reduction in sucrose production from sugar beet roots while increasing impurities concentration, which results in higher operation costs [2]. Given the high cost and environmental effect on applying fungicide methods to overcome sugar beet diseases, planting resistant cultivars using advanced precision farming techniques is the most common and practical method to battle this disease [3]. To identify resistant varieties, once every few days over a course of a few months, the domain experts walk through the field, visually observe the diseased plants, and rate the level of cultivars disease severity using the rating system adopted by U.S. Department of Agriculture (USDA), designated here as the "USDA scale" [4]. However, this manual rating system has three critical drawbacks. It is accompanied with high variations where multiple experts may have different ratings for the same plant, laborious where it requires large amounts of time from experts for frequent and large-scale rating, and relatively insensitive where the human eye is not sensitive enough to rapidly differentiate subtle variation of leaf appearances. Therefore, an improved rating system addressing these drawbacks is highly desired.

Considering the popularity and ever-reducing cost of cameras, a computer vision-based approach can be an excellent choice for a rating system where the images of plants are analyzed and rated in an automated, consistent, and efficient manner. Unfortunately, the agricultural industry appears to lack such types of commercial

* Corresponding author.
E-mail addresses: atoumyou@msu.edu (Y. Atoum), afridimu@msu.edu (M.J. Afridi), liuxm@cse.msu.edu (X. Liu), mitchmcg@msu.edu (J.M. McGrath), hansonl5@msu.edu (L.E. Hanson).

http://dx.doi.org/10.1016/j.patcog.2015.11.021
0031-3203/

systems. In the research community, most of the prior work focuses only on detecting or classifying CLS from a magnified and well-controlled view of leaf images [5–8]. Although such leaf-level approaches simplify the classification problem, they are hard to adopt in practice due to the stringent requirements on image acquisition. In addition, single leaf ratings have been shown to be less reliable for predicting plant damage than whole plant ratings [9].

Alternatively, plant-level images can be more conveniently acquired in real fields via a fly-over UAV or a drive-through tractor (Fig. 1). However, automatic rating on plant-level images is challenging, as illustrated in Fig. 2. The varying light conditions in different weather contribute to a large amount of appearance variations in the images. Dark shadows tend to hide the details making it difficult to analyze the appearance patterns of diseased spots. In the higher ratings of CLS, the dead plants are often difficult to discern from the soil background and hence not confusing them with soil is challenging. Similarly the specular reflection from the sun in healthy leaves displays a yellowish color that is normally present around the diseased leaves, increasing the potential of confusion.

In order to fulfill the application needs and address the technical challenges, we propose a novel system, CLS Rater, for automatic rating of CLS disease in plant-level images captured by a tractor-mounted camera. Notably, this application demands a global rating estimate of a plant image by analyzing diverse appearance patterns of disease in its local regions. We tackle this challenge by our novel technical contribution of superpixel-based *Histogram of Importances (HoI)* features that describe the local patterns of each superpixel aggregated across the global image level. We then utilize these features for learning image-level regression models. Although superpixels are frequently used in image segmentation [10–12], they have not been explicitly used to learn image-level regression models. Furthermore, depending on the rating of a plant, the distinctive regions of diseased leaves can have diverse sizes, from a tiny spot to an extensive area of dead leaves. Hence, the superpixels extraction is conducted at multiple scales, ranging from hundreds to thousands of superpixels, and the proposed HoI feature is extracted at each scale. Finally, the features from multiple scales are fused, from which a regressor is learned based on a set of images and their manual rating (or label) in USDA scale.

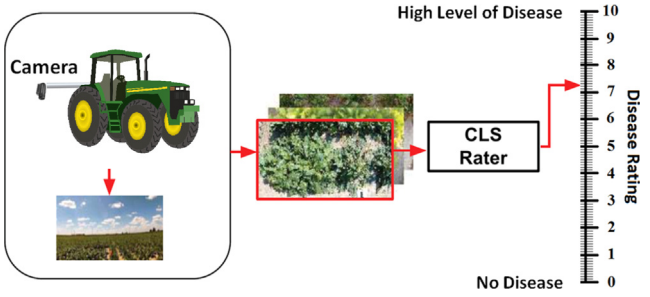

Fig. 1. A camera mounted to a field tractor records the plant videos. CLS Rater performs automated analysis and assigns a rating of "USDA scale" to each video frame.

Using our novel CLS Rater, we have the capability to address some of the existing drawbacks (i.e., laborious and the high variation in labels), simply by driving the tractor through the field and automatically rating every plant with the USDA scale. Unfortunately, the drawback of insensitive has not been well tackled since the manual ratings, on which CLS Rater is trained, are generated using a rating scale designed for low sensitivity. Furthermore, the manual ratings are known to be noisy, as evidenced by the large variations among multiple experts. For example, in the ratings from three experts over a two-month period, the level of disagreement in ratings is considerably high with a standard deviation of 1.31. Hence, it is reasonable to conclude that the CLS Rater learned from the noisy ground truth still desires further improvement. Finally, we hypothesize that enhancing the manual ratings of training samples is able to produce a more consistent and accurate CLS Rater. After applying the label enhancement module (LEM) to the training set, an enhanced CLS Rater can be trained with the new ratings.

Extensive experiments are conducted by using the video data captured in the real field under different outdoor weather conditions, for two consecutive years (2013 and 2014). First, we test the CLS Rater based on the ground truth manual ratings on the 2013 dataset. Experimental results show that our system is more consistent compared to the human rating. CLS Rater can predict ratings with an average rating error of 0.65. Furthermore, when

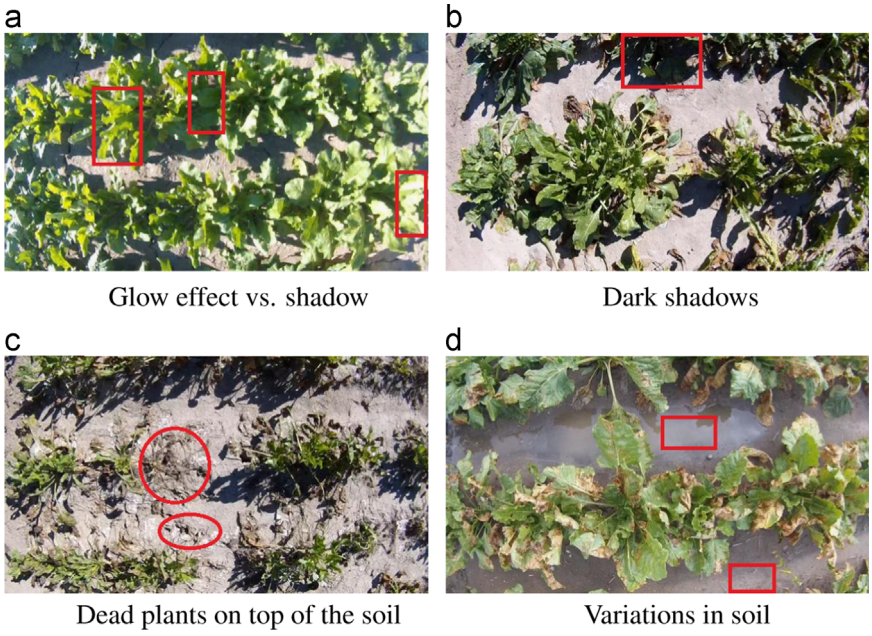

Fig. 2. Appearance variations of real-world plant images in the field: (a) glow effect vs. shadow, (b) dark shadows, (c) dead plants on top of the soil, and (d) variations in soil.

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021

applying the LEM, the enhanced CLS Rater can reduce the error to 0.59. Finally, cross-year experiments are performed by testing the CLS Rater learned in 2013 on the unseen data in 2014.

A preliminary version of this work was published in the International Conference on Pattern Recognition 2014 [13]. We have extended it in a number of ways: (i) developed the LEM to address the issue of noisy labels; (ii) further reduced the rating error of CLS Rater; and (iii) conducted experiments on real-world data of two consecutive years.

In summary, this paper makes four main contributions:

- We design a practical computer vision system that conveniently consumes plant-level images of a real field and automatically rates the CLS resistance in USDA scale.
- We propose a novel HoI feature over the multi-scale superpixels representation, and demonstrate its effectiveness in the regressor learning.
- We address the problem of noisy labels by proposing an LEM, and experimentally show the superior performance of applying LEM over the one using the noisy labels obtained from the experts in this field.
- We collect a Real-World Sugar Beet Database with various degrees of CLS disease and the associated manual ratings in the USDA scale, over a two-month period in both 2013 and 2014. This dataset is publicly available to the research community.[1]

2. Prior work

Considering the contributions of our work, we review relevant prior work in three areas, disease rating, feature representation, and noisy label handling.

In the work of Hanson et al. [14], a wide variety of sugar beet cultivars are grown and manually rated for evaluating their resistance or susceptibility to CLS. There have been a number of prior work focusing on detecting or classifying CLS severity in sugar beets [5–8,15,40]. These approaches utilize magnified leaf-level images to detect the diseased segments and classify a leaf as diseased or healthy. Such approaches address a less challenging problem than ours due to the use of leaf-level images and a two-class classification task, while we perform regression from plant-level images. Furthermore, these approaches are hard to adopt in practice since it is inconvenient to acquire leaf-level detail of each plant in a large field. For instance, in [5], authors classify different diseases in sugar beet leaves, where the plants are grown under controlled laboratory conditions. In [6], the authors use leaf images to differentiate a CLS-symptomatic leaf from a healthy one by an SVM classifier. Similarly, [7,8] also use leaf images and utilize a threshold-based strategy to monitor the diseased part of a leaf. Moreover in [15,40], the authors propose an algorithm to continuously monitor the disease development under real field conditions. This method is applied on a single leaf scale for disease observation, which requires tracking and aligning the same exact leaf across several days. In contrast, we collect plant-level images in a real field under diverse weather conditions, which exposes our system to all kinds of real-world challenges. Further, our system learns a regression model that predicts the continuous severity of CLS disease. To the best of our knowledge, this is the first study to utilize the plant-level real field images and automatically predict the fine-grained severity of a disease.

Since our feature representation builds upon the superpixel, we provide a brief overview of the related work in superpixels. With time, superpixel-based methods are becoming more advanced. For example, authors of [16] discuss how superpixels resulting from different techniques can be combined to improve image segmentation. Similarly, various studies utilize superpixels for classifying local image segments [17]. In [18], authors use a multi-scale superpixel classification approach for tumor segmentation. Furthermore, superpixels have been utilized in various other applications as shown in [10–12]. Note that in our study, CLS rating needs to be conducted globally for an entire image, while superpixels only capture local characteristics of an image. Hence, to fill in the gap, we need to address *how* the local characteristics of superpixels can be summarized as an image-level representation, which unfortunately has not been explicitly studied before and is one novelty of our technical approach.

Label noise is a well-studied problem over the last few decades, due to its negative impact on any pattern recognition problem. Having noisy labels will affect the classification model, increase the complexity, and ultimately reduce the accuracy [19]. Some researchers attempt to learn models that are robust when training data has label noise [20,21]. An alternative approach is to detect noisy labels, correct, or remove them [22,23,41]. A third type of approach is to use classification filtering as a preprocessing step [24–28]. For example, Adaboost is used to filter mislabeled samples in [24], by eliminating a group of the samples with the highest weights. However, most prior work eliminates mislabeled data instead of correcting them, which reduces the number of samples. Also, the majority of them use synthetic data with injected noise [19], rather than real world data as in our case. It is worthy to note that the noisy CLS rating is not caused by mistakenly assigning an incorrect class label, instead it is due to the difficult nature of assigning disease ratings that may vary from one person to another, or one field to another. The limited information provided from the USDA scale of each rating class is one reason for this problem. This task is highly subjective based on how the expert interprets the different ratings from 0 to 10. For example, in our dataset, the standard deviation of CLS ratings among multiple experts can be as high as 1.31. Therefore, given the fact that label noise is presented in almost all samples in our dataset, it is important to be able to correct or enhance the labels, which is the main goal of our LEM.

3. Proposed approach of CLS rater

The input data to the proposed CLS Rater is the plant-level imagery captured by a face-down camera mounted on either a fly-over UAV or a horizontal pole on a regular field tractor. Specifically in this paper we adopt the latter, as illustrated in Fig. 1. Given the captured plant images, we use a superpixel-based approach to extract features at a pre-defined scale, e.g., M superpixels, that best describe the local characteristics. The superpixel is well suited for our given problem, because it concisely and efficiently represents local appearances at a diverse range of scales by grouping pixels with locally uniform color and texture. After superpixel extraction, there are many types of features to represent a local region. We focus on color and texture based feature representation. A D-dim feature vector, e.g., a color and texture histogram, is extracted to represent the local appearance of a superpixel. Given the $M \times D$ feature matrix extracted from all superpixels of an image, we describe the appearance variations across all superpixels via our superpixel-based HoI features, by computing a T-dim histogram for each column of this matrix. This results in a DT-dim vector, where each element describes the distribution of *relative importance* of one feature, e.g., one representative color, among all individual superpixels.

Color features are the most important in this problem, since it is the core indication of CLS severity on the leaves of the plant. A CLS-symptomatic plant exhibits more yellow color in comparison

[1] http://www.cse.msu.edu/liuxm/precisionAgriculture.html

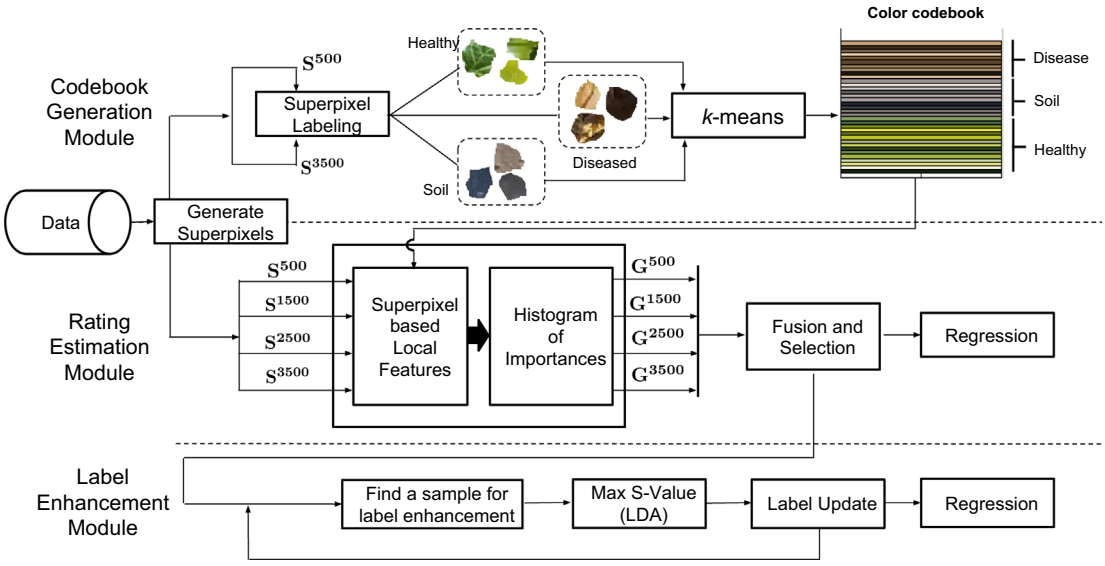

Fig. 3. The high-level architecture of our CLS Rater system.

to a healthy one, where the amount of yellow indicates the disease severity. When a plant is going through different stages of CLS disease development, the color as well as the amount of healthy leaf, diseased leaf tissue, and visible soil regions in plant images are changing accordingly. Therefore, color can be very useful in discriminating these three types of regions and further contributing substantially to the prediction of the rating. Similarly, texture also exhibits distinct patterns on these different regions. Healthy leaves can be described to be smoother, where diseased ones can be characterized to have dried and rough surfaces. Thus, texture is also a good candidate to discriminate between healthy and non-healthy plants.

Similar to any learning-based computer vision system, CLS Rater has a training stage and a testing stage. During the training stage, a regressor is learned from a set of plant images and their ratings in "USDA scale", with the goal that the predicted rating from the regressor is as close to the manually labeled rating as possible. While in the testing stage, the learned regressor is applied to an unseen plant image to automatically predict its disease rating. As shown in Fig. 3, the training stage includes three modules: codebook generation module (CGM), rating estimation module (REM) and label enhancement module (LEM), while the testing stage only includes the REM.

The goal of CGM is to model the representative colors in three different types of regions, i.e., healthy, soil and disease. In CGM, we manually label diverse sets of superpixels into each of the three regions, to which *k*-means clustering is applied independently for generating the codewords of these three regions. In REM, superpixels are extracted from a set of images at four scales, where at each scale a novel feature representation is used to describe both the local and global image characteristics. Features at all scales are then fused and a regressor is learned from the selected features. Processing in the testing stage is the same as REM except that it takes only one image as input. In LEM, we perform label enhancement on the manual ratings obtained from the experts in the field in order to reduce the amount of label noise and better distinguish all possible ratings. This is accomplished by iteratively adjusting the existing rating of each sample, with the goal of achieving the maximum separation among the training samples of different ratings in the feature space. The separation is measured using the multiclass Linear Discriminant Analysis (LDA), which explicitly models the linear separability among the data of multiple classes. We describe the key components of the training stage starting from superpixel extraction, to a detailed explanation for all three modules as follows.

3.1. Superpixel extraction

CLS in its early stages appears as very small spots located on the leaves of the sugar beet plant. As the disease progresses to higher levels, the spots increase in number and coalesce, and the diseased areas change in color. Therefore, the disease segments show large variations of scales ranging from a tiny spot to a large segment depending on the level of CLS severity. Instead of developing an approach to detect spots with varying sizes, as examples in cell [42,43] and fish feed [44], we adopt a middle-level representation, superpixels. A superpixel is a local segment in an image containing a group of neighboring pixels with similar appearance. Normally a scale is specified so that a pre-determined number M of superpixels can be generated for one image. To capture the local characteristics of diseased spots at all rating levels, we generate superpixels $\mathbf{S}^M = \{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_M\}$ of an image at four different scales where $M = \{500, 1500, 2500, 3500\}$. Using the standard implementation of [29], we observe that superpixels at each scale cover local image characteristics in a unique way, as shown in the zoomed-in views of the smallest and largest scales in Fig. 4. For example, small sized superpixels, obtained with a large M, can completely fit to a small diseased spot developed in the early CLS stage. Although a larger sized superpixel cannot restrict its boundary to a small segment present in low rating images, it covers the surrounding of such a small spot and hence provides useful neighborhood contextual information, as indicated by the two parallel arrows in Fig. 4. On the other hand, in high rating images, larger superpixels can cover an entire large area of coalesced spots and provide a more confident indication of the severity of CLS (the leftmost arrow in Fig. 4). Combining all the features obtained from superpixels of various M scales will effectively describe all rating levels of the disease.

3.2. Codebook generation module

For an arbitrary image, the color of pixels may not have a priori distribution. However for domain-specific images such as sugar beet plant images, it is safe to assume that a distribution of pixel color exists and can be learned for efficient feature representation. Therefore, motivated by the Bag of Words (BoW) approaches [30],

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021

we first learn a color codebook to estimate the representative colors (codewords) in the plant images as illustrated in Fig. 3, so that they can be used later for feature representation. From our dataset we manually select a diverse set of $B=33$ images with various severities of CLS. The images were selected uniformly across several days throughout the sugar beet season, capturing all ratings of CLS disease, and all variations in lighting and weather conditions. For each image, $\mathbf{I}_i$, superpixels at multiple scales $\{\mathbf{S}_i^M\}$ are extracted. To facilitate the labeling for CGM, we develop a GUI where the superpixels $\mathbf{S}_i^M$ of image $\mathbf{I}_i$ are displayed on the screen and a user may select superpixels belonging to healthy, diseased or soil regions via mouse clicks. The selected subsets are denoted as $\mathbf{S}_i^h$, $\mathbf{S}_i^e$, and $\mathbf{S}_i^s$ respectively. We perform this step for all B images to form $\mathbf{S}_H=\{\mathbf{S}_1^h,\mathbf{S}_2^h,\ldots,\mathbf{S}_B^h\}$, $\mathbf{S}_E=\{\mathbf{S}_1^e,\mathbf{S}_2^e,\ldots,\mathbf{S}_B^e\}$ and $\mathbf{S}_S=\{\mathbf{S}_1^s,\mathbf{S}_2^s,\ldots,\mathbf{S}_B^s\}$. We collect 150 superpixels for each of the three categories. This superpixel selection procedure is performed at two scales only: $\{\mathbf{S}_i^{3500}\}$ containing smaller superpixels for selecting diseased spots, and $\{\mathbf{S}_i^{500}\}$ for healthy plants and soil.

The RGB pixel values of all pixels within the superpixels of $\mathbf{S}_H$, $\mathbf{S}_E$, and $\mathbf{S}_S$ are fed to the k-means clustering for extracting codewords of each category. We extract 10 codewords each for the disease and soil categories, and denote them as $\mathbf{C}_E$ and $\mathbf{C}_S$ respectively. Since the healthy part shows larger variations and also responds with lighter green in regions around the diseased part, we select 15 codewords $\mathbf{C}_H$. We combine $\mathbf{C}_H$, $\mathbf{C}_E$, and $\mathbf{C}_S$ to form a codebook with $D=35$ codewords $\mathbf{C}=\{\mathbf{c}_1,\mathbf{c}_2,\ldots,\mathbf{c}_{35}\}$, which will be used in the REM described below. An alternative approach to our codebook learning is to directly learn the color codewords from the images, which is not preferred because the resulting codewords will mainly cover the variations in healthy and soil parts, hence creating a biased codebook. Another possible approach is to use various color invariants [31], for extracting discriminative features that are invariant to illumination and sensor characteristic. The authors in [31] show that some specific color representations are very useful for applications such as face recognition, and experimentally demonstrate that the non-linear effects in the photometric response of the camera are important to derive invariant representations. As one of our future work, it is interesting to study whether similar finding can be observed in the color representation of sugar beet plants.

3.3. Rating estimation module

Given the color codewords from the CGM, as well as the superpixels of an image set, this rating estimation module performs two main tasks: (1) feature representation, and (2) feature selection and regressor learning. We now discuss them as follows.

3.3.1. Feature representation

Feature representation is critical for any computer vision system. Classifying local regions in superpixel segments into diseased or healthy may seem to be a trivial task. However, it is unclear how to generalize this task to consider a global image-level feature that captures both the local pixel statistics, such as the small diseased spots, and the global image regularity, such as a large region of dead leaves. Moreover, a global fine-grained continuous rating needs to be learned from the feature representation of images. These considerations lead to the proposed novel Histogram of Importances feature, computed in two steps.

In the first step, a histogram feature is extracted to represent the color variation of all pixels within each superpixel based on the color codewords. Given that an image $\mathbf{I}$ contains a set of M superpixels $\mathbf{S}^M=\{\mathbf{s}_1,\mathbf{s}_2,\ldots,\mathbf{s}_M\}$, we compute a set of color histograms $\mathbf{H}=[\mathbf{h}_1^\top;\mathbf{h}_2^\top;\ldots;\mathbf{h}_M^\top]$. For each superpixel $\mathbf{s}_m\in\mathbf{S}^M$, we have $\mathbf{h}_m(d)=\frac{h_d}{|\mathbf{h}_m|}$, where h_d indicates the number of pixels $\mathbf{u}$ within $\mathbf{s}_m$ whose color is most similar to $\mathbf{c}_d$ among all D codewords, i.e., $h_d=\sum_{\mathbf{u}\in\mathbf{s}_m}\delta(d=\arg\min_d\|\mathbf{I}(\mathbf{u})-\mathbf{c}_d\|_2)$, and $\delta()$ is the indicator function.

Although $\mathbf{h}_m$ is a good descriptor of local appearance at each superpixel, it cannot be applied to regression learning directly because superpixels between two images may not correspond to each other, and the numbers of superpixels M can be different too. Hence, we aim to extract an image-level feature independent of superpixel locations or M. Specifically, by observing the matrix $\mathbf{H}$ of an image, each element $\mathbf{h}_m(d)$ indicates the relative importance of the color feature $\mathbf{c}_d$ within the superpixel $\mathbf{s}_m$. Such an importance value can vary between 0 and 1. By collecting all the importance values corresponding to the same feature $\mathbf{c}_d$, i.e., one column of $\mathbf{H}$, we can form a T-dim histogram of importance (HoI) $\mathbf{g}_d$, where $\mathbf{g}_d(t)=\sum_m\delta(\frac{t-1}{T}\le\mathbf{h}_m(d)<\frac{t}{T})$, $1\le t\le T$, and both t and T are integers. We show this procedure diagrammatically in Fig. 5. By collecting the HoI of all D color codewords, we have a $D\times T$ feature representation $\mathbf{G}^M=\{\mathbf{g}_d\}$ for one superpixel scale M.

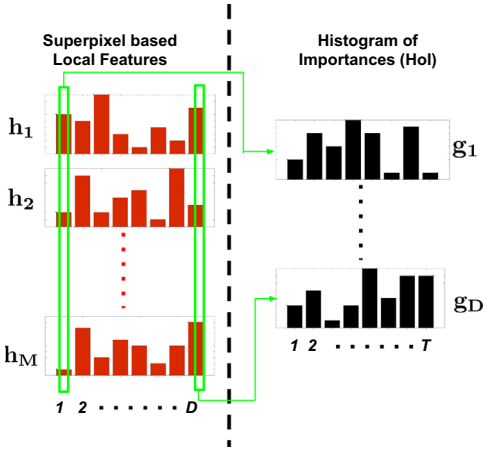

Fig. 5. From the histograms of individual superpixels to the Histogram of Importances (HoI).

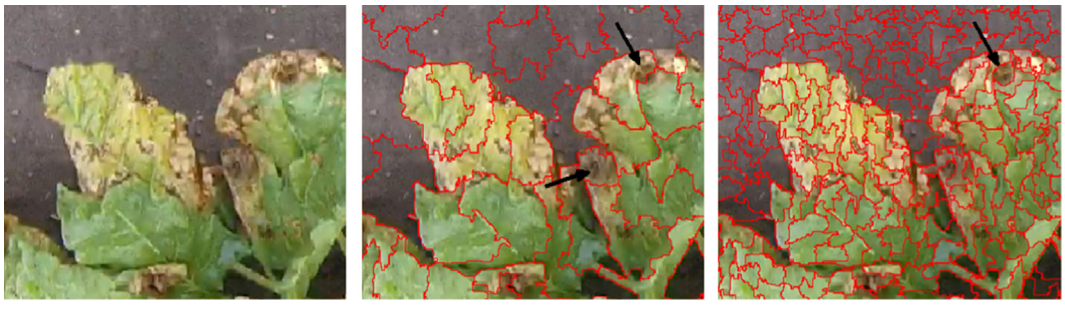

Fig. 4. Superpixels at $M=500$ (center) and 3500 (right) for a local region (left) of a captured image. Note that these images are the zoomed-in views of one local region of the original captured image (e.g., one in Fig. 2).

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021

Similar HoI features are also computed for the LBP-based texture features [32] $\mathbf{L}^M$, where $D=256$. In our study, we use $T=10$ for color features and $T=5$ for LBP features. Thus, for each image at one superpixel scale, we have a total of 1630 features. To visualize the HoI features, Fig. 6 plots $\mathbf{G}^M$ of nine randomly selected images at $M=500$. We can clearly see a decrease of importance in healthy features and a slight increase of importance in soil features, as we move to higher ratings.

3.3.2. Feature fusion, selection and regression

As mentioned before, superpixels at different scales cover local characteristics in different ways and provide different advantages over each other. Therefore, to enjoy the benefits from every scale, we compute the color and LBP based HoI, $\mathbf{G}^M$ and $\mathbf{L}^M$, at all four scales for each image, which results in a feature vector with the length of 1630×4. However, since not all feature elements have a high discriminative power, we perform feature selection by the correlation-based approach [33], which is based on two measures: the high predictive ability and the low correlation with already selected features. We then pass the selected set of 162 discriminative features, $\{\check{\mathbf{G}}^M, \check{\mathbf{L}}^M\}$, to the bagging M5P regressor [34,35]. M5P decision tree learns different regression functions for each leaf node of the tree. Experiments in Section 5 provide a comparative study of different regression schemes on our features. Our results show that bagging M5P to be superior to other well-known regression paradigms.

3.4. Label enhancement module

So far we have presented a carefully designed learning-based approach to automatically estimate or mimic the disease rating manually labeled by domain experts. However, such manual ratings, either from one expert or the average of multiple experts, are inevitably noisy. For example, Fig. 7 shows that the disagreement among experts is almost everywhere on an entire dataset, with especially large variation for some images (Fig. 7 (a)). As mentioned before, the noisy label is caused by a number of factors, including the level of sensitivity of the human eye, the nonspecific definition of the USDA scale, and the existence of multiple plants within one image. For these reasons, this issue cannot be solved by the experts, and thus an automatic method to enhance the noisy labels of a dataset is desired, which is exactly the objective of LEM.

One potential approach of LEM is to adopt unsupervised learning to learn 11 clusters, each corresponding to one level of CLS disease. However, our preliminary experiment shows that without supervision it is difficult to ensure that the clusters are indeed defined based on the CLS severity. Therefore, we make the following assumption: the noise-free rating of a data sample is in close approximation to its manual rating, and it is thus possible to obtain the former by making a small adjustment to the latter. Based on this assumption, we take the manual ratings as the starting point, and improve them in a systematic manner, with the goal that the enhanced labels will make the different rating levels more discriminative in the feature space. This will in turn result in an enhanced CLS Rater, when trained from the enhanced labels. Specifically, given a dataset and its manual ratings as input, after feature extraction from the REM, the LEM iteratively updates one rating at a time in order to maximize the separation among samples of different ratings. A simple illustration of our proposed LEM is shown in Fig. 3, and a more detailed explanation is in Algorithm 1.

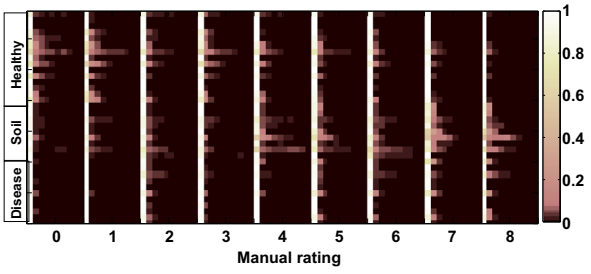

Fig. 6. Color-based HoI of nine images with different ratings.

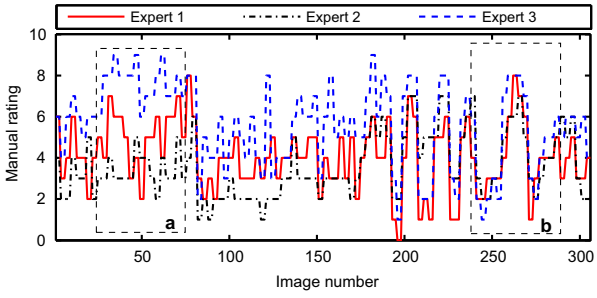

Fig. 7. Assigning disease rates to images from the real-world field is challenging. There are large variations (a) and small variations (b) in manual ratings of three experts.

Algorithm 1. Label enhancement module.

Data: $\mathbf{Y}, \mathbf{X} = \{\check{\mathbf{G}}^M, \check{\mathbf{L}}^M\}$

Result: $\tilde{\mathbf{Y}}$

$\tilde{\mathbf{Y}} = \mathbf{Y}, S = 0$;

do

  $j = F(\tilde{\mathbf{Y}})$;

  $\tilde{\mathbf{Y}}_j^{set} = [\lceil \tilde{\mathbf{Y}}_j + 0.5 \rceil, \lfloor \tilde{\mathbf{Y}}_j \rfloor, \lfloor \tilde{\mathbf{Y}}_j - 0.5 \rfloor]$;

  for $i = 1:3$ **do**

    Partition N samples into c classes based on $\tilde{\mathbf{Y}}$ and i^{th} element of $\tilde{\mathbf{Y}}_j^{set}$;

    $\mathbf{\Sigma} = \sum_{m=1}^{c} \sum_{n=1}^{N_m} (\mathbf{x}_n^m - \mu_m)(\mathbf{x}_n^m - \mu_m)^\intercal$;

    $\mathbf{\Sigma}_b = \sum_{m=1}^{c} (\mu_m - \mu)(\mu_m - \mu)^\intercal$;

    $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \cdots] = eig(\mathbf{\Sigma}^{-1}\mathbf{\Sigma}_b)$;

    $S^{set}(i) = \frac{1}{c-1} \sum_{m=1}^{c-1} \frac{\mathbf{w}_m^\intercal \mathbf{\Sigma}_b \mathbf{w}_m}{\mathbf{w}_m^\intercal \mathbf{\Sigma} \mathbf{w}_m}$;

  end

  $i = \underset{i}{\operatorname{argmax}}\, S^{set}$;

  $\tilde{\mathbf{Y}}_j = \tilde{\mathbf{Y}}_j^{set}(i)$;

  $S_{pre} = S, S = S^{set}(i)$;

while $S > S_{pre}$;

It is obvious that the order of samples being processed within an input dataset of N samples affects the final enhanced labels. Thus, we denote the method for selecting which sample to update its label by $F: \mathbf{Y} \mapsto j, j \in 1, 2, .., N$, where j is the index of the candidate sample. In this work we explore three options for implementing this function: (i) *Random label selection*: This function randomly selects one sample from the input dataset, without considering any prior knowledge about the label. (ii) *Maximum disagreement first*: This function ranks all samples in the descending order of the disagreement among the experts. It first selects samples with the most confusing labels (i.e., the largest disagreement). (iii) *Maximum offset first*: Given a dataset and the current labels, a M5P regression-based CLS Rater is learned and

applied to the training dataset. The sample with the maximum difference between the current label and the rating predicted by CLS Rater is selected as the sample to be processed.

After finding the candidate sample, we assume that its label $\tilde{\mathbf{Y}}_j$ can make a small adjustment to one of the following neighboring ratings: $[\lceil\tilde{\mathbf{Y}}_j+0.5\rceil, \lfloor\tilde{\mathbf{Y}}_j\rceil, \lfloor\tilde{\mathbf{Y}}_j-0.5\rfloor]$. Therefore, we consider the possibility of either modifying this label to one of the neighboring ratings, or maintaining its current label. For each possibility, we compute the S value of the feature set $\mathbf{X}$ given the updated labels $\tilde{\mathbf{Y}}_j$, where S is the class separability computed via a multiclass Linear Discriminant Analysis (LDA). Specifically, we compute $\boldsymbol{\Sigma}_b$, $\boldsymbol{\Sigma}$, $\mathbf{W}$, and S is the average of eigenvalues that are indicative of linear separability among multiple ratings. Finally, we update $\tilde{\mathbf{Y}}$ to the possibility that produces the maximum S value. Note that the label of a sample can be modified more than once, when other samples in the dataset are modified. While making modification on the labels of the samples, it is important to preserve the range of the labels because the S value will shrink if the range is reduced. Therefore, we have a constraint to enforce that no label modification is performed for samples with the maximum rating or minimum rating of a particular dataset. The enhancement process will continue until there is no increase in S values. This means that all data samples are well separated into rating clusters with the minimum overlap among the clusters, and a regressor will then be learned based on the enhanced labels $\tilde{\mathbf{Y}}$.

4. Real-World Sugar Beet Database

Although there are prior works on computer vision-based agriculture applications, there are very few public databases of plant images that are captured in the field. Thus, one contribution of our work is to acquire a sugar beet plant database in two consecutive years with the same imaging setup, and to make this database publicly available.

A conventional RGB camera was attached to a tractor pointing downwards at a height of 1.2 m. The tractor drives through the sugar beet field while maintaining a constant speed of $\sim$1 m s^{-1} (2.2 mph), capturing videos at a frame size of 1080 $\times$ 1920 and 30 frames per second for the entire field. We reduce the frame size of all images to 540 $\times$ 960 for improved computational efficiency. To record the progress of CLS disease, we collect videos periodically during the sugar beet growing season, across a period of two months capturing a wide range of disease severity. Our sugar beet field is of a rectangular shape at 135 $\times$ 168 m. Each section of the field corresponds to a known sugar beet cultivar, with a total of 458 cultivars over the entire field. Hence, the CLS rating study provides many insights to the domain experts regarding the CLS resistances of various cultivars. Along the short edge of this rectangle there are 22 parallel field lines with equal distances between them, where our tractor drives along each of the field lines for data collection.

The first part of the database was captured from July 30, 2013 to September 12, 2013 on 10 different dates. Among these 10 dates, there are 6, 2, and 2 dates with sunny, cloudy and partly cloudy weather respectively. We collect 220 total videos, i.e., 22 videos per day. Each video is about 3 min long and covers one field line. We select a diverse set of 306 images from this dataset ranging through all dates to capture all possible disease ratings. Using the USDA scale, three experts independently provide manual ratings for all these images. The overall distribution of all labeled images across different ratings is tabulated in Table 1. The ratings provided from three experts for all 306 images are also shown in Fig. 7, illustrating the variations in the ratings.

The second part of the database was captured from August 15, 2014 to September 12, 2014 on 7 different dates. Among these 7 dates, there are 3, 1, and 3 dates with sunny, cloudy and partly cloudy weather respectively. This part used the exact same imaging setup as the first part, where the only differences are in the capturing and labeling procedure. Instead of capturing every line in the field separately, the entire field was captured in a total of 2 videos. A GPS system, as an integrated component of the tractor, was utilized to record exact longitude and latitude coordinates while capturing videos. For this part, only one expert provides manual ratings to the plants on 4 out of 7 dates, while she walks through the field, and the manual ratings are recorded w.r.t. the locations of cultivars. Since we aim to have labels for all 2014 datasets, we did not ask the expert to manually label a small subset of images. Instead, using the GPS data, we map all manual ratings in the field to specific video frames, as shown in Fig. 8. However, due to imprecise GPS data, the manual ratings in 2014 dataset are not as ideal as the one in 2013 dataset.

Table 1
Overall distribution of all labeled images across different ratings.

Manual rating	0	1	2	3	4	5	6	7	8	9	10
# of images in 2013	1	11	43	60	49	46	47	24	21	4	0
# of images in 2014	0	34	575	630	878	1121	663	121	2	0	0

5. Experimental results

In this section, based on the Real-World Sugar Beet Database, we design experiments to answer the following questions: (1) How does the CLS Rater perform in comparison to manual expert rating? (2) How do different regression schemes perform at different superpixel scales? (3) How do our discriminative features vary across different CLS ratings? (4) Does maximizing the separation value in the LEM indeed change labels according to disease levels? (5) How do we evaluate the performance of the enhanced labels? We now discuss different aspects of our experiments.

Experimental setup: Most of our experiments are based on the 2013 dataset, where we randomly split the 306-image set into two equal parts and use one for regressor training and the other for testing. This is also repeated to generate multiple partitions of training and testing sets. For each image $\mathbf{I}_i$ in our dataset, the manual ratings from three experts are averaged to generate the ground truth rating $\bar{r}_i$. Given $\bar{r}_i$ and the estimated rating of $\hat{r}_i$ from CLS Rater, we compute the rating error of our system on a K-image testing set as $e=(\frac{1}{K}\sum_i \|\bar{r}_i-\hat{r}_i\|^2)^{1/2}$.

*Feature analysis:*We start by analyzing the performance of the proposed HoI features and the selected features by one of the best performing classifiers, M5P regressor as indicated in the regression result section, during the training stage. Specifically, we evaluate the effectiveness of the selected features and compute their feature value across different unseen testing images with varying CLS disease ratings. Note that the M5P is a tree-based regressor, where each node is associated with a selected feature. From the M5P hierarchy, we select the top four nodes (features) that represent different types of features, i.e., the color features from the disease, soil and healthy region and one LBP-based texture feature. In order to see how effective these four selected features are on the testing images, we allocate the testing images with the same ground truth rating into one group. For each of the four selected features, we compute its average feature values from images within the same group. This leads to a vector for each selected feature, which is further normalized by dividing with the maximal element in the vector. We plot the resulting four vectors in Fig. 9, which illustrate a clear trend of the four features. We notice a proportional

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021

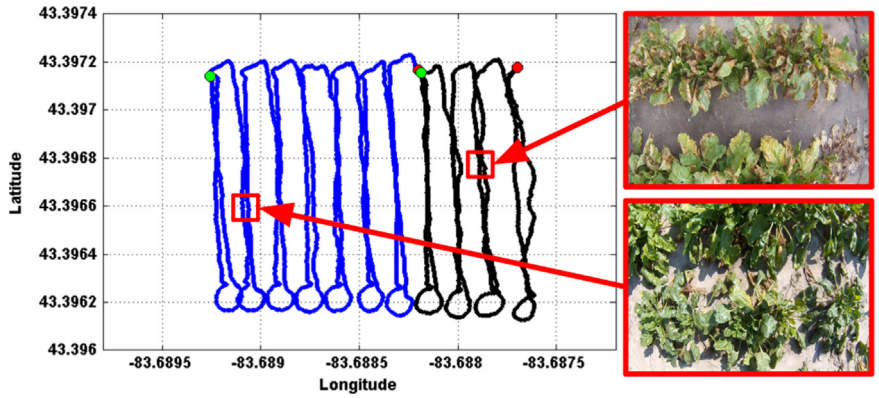

Fig. 8. Mapping GPS coordinates to specific video frames. The blue and black lines represent two video sequences captured on August 21, 2014. The green and red circles represent the start and end of each video sequence respectively. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

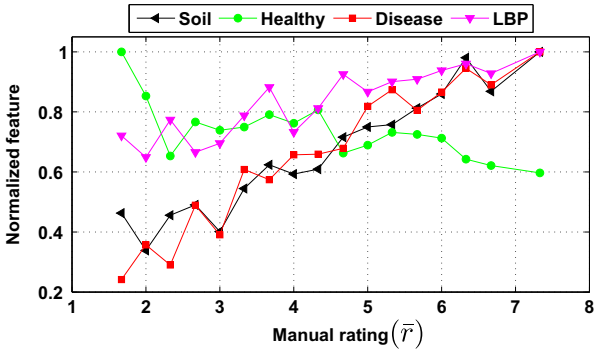

Fig. 9. Top hierarchy features of bagging M5P regressor.

relationship among soil, disease and LBP features with a high correlation in the behavior across ratings. Whereas, the healthy leaves tend to have an inverse relationship with all other features. This is highly expected, since at higher ratings, the amount of green leaves in the frame decreases, which are typically replaced with diseased leaves and soil. This study also provides an insight on how the HoI feature element extracted from various regions contributes to CLS rating.

*CLS Rater prediction analysis:*While Fig. 9 indicates the strong correlation between the novel HoI features and the rating, the ability of CLS Rater to predict rating is more important. Our CLS Rater is designed to predict ratings based on the USDA scale with 11 different levels of disease ratings. To analyze the predictions of our rater based on the novel HoI features, we attempt to test the discriminative ability of the rater across a large variety of ratings.

Using the experimental setup on the 2013 dataset, the predictions on one testing set are illustrated in Fig. 10. The narrow line-like plot shows that the rating error is evenly distributed across the entire rating range, and also our CLS Rater is able to predict labels very similar to the human labels on the unseen data, which is desired for practical applications.

Fig. 11 illustrates the strength of the CLS rater at a global scale, as well as locally at more challenging cases with high disagreement. We use the labeled data collected from 2013 that consists of 306 samples, where every sample is labeled by three experts. At a local scale, the data samples with high disagreement tend to have similar rating errors as the samples with lower disagreement. At a global scale, the absolute difference between the experts mean rating and the CLS estimated rating is all less than 1.3 (i.e., worst case) on the USDA scale, with an average error of 0.65, which is far less than most of the standard deviations of expert ratings.

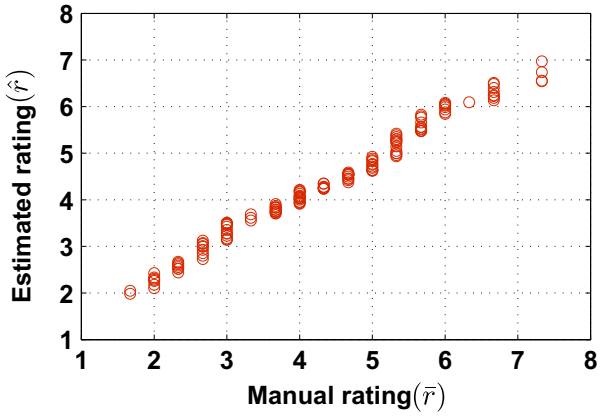

Fig. 10. Ground truth manual rating vs. the estimated rating of CLS Rater.

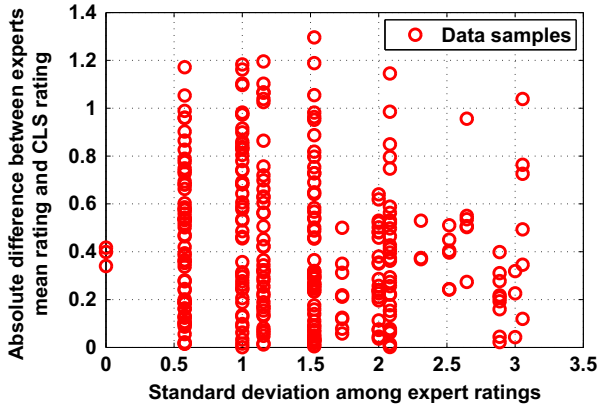

Fig. 11. Experts' disagreement vs. CLS rating error. The *x*-axis indicates the amount of disagreement across three experts represented with the standard deviation. The *y*-axis is the absolute difference between the experts mean rating and the CLS estimated rating.

*Label enhancement results:*We now study the LEM and its contribution to CLS Rater. First, we explore the various methods for selecting which sample to update the label, which is the function F with three options: random label selection, maximum disagreement first or maximum offset first. We attempt to enhance the ground truth label of the training set of 2013 dataset with a total of 153 images. Fig. 12 shows a comparison of all three functions during the iterative process of selecting the candidate sample. It is worthy to note that all three methods start at low S values

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021

meaning that the ground truth ratings are not well separated among different ratings. The best resulting S value is produced using the maximum disagreement first method, which converges at $S=172$ after a total number of 1982 iterations. This method selects the sample with ratings that has the highest inconsistency among multiple experts.

After the label enhancement converges, we can compare the original ground truth ratings (average of three manual ratings) with the enhanced ratings generated from the maximum disagreement first method, as shown in Fig. 13. On one hand, although on average each samples rating has been examined nearly 13($\approx\frac{1982}{153}$) times, the differences between the original and final ratings are very minimal, where the absolute difference has a distribution of $\mathcal{N}(0.56, 0.62)$. This is a good indication of our assumption that noisy-free label of a sample is in close approximation to its manual label. On the other hand, even with a small modification on the ratings, a much larger S value is achieved which indicates improved separability among different ratings.

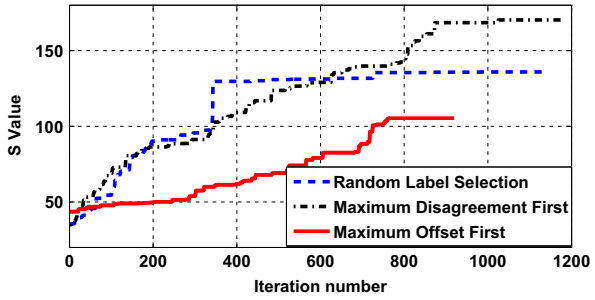

Fig. 12. S value comparison of the three sample selection functions.

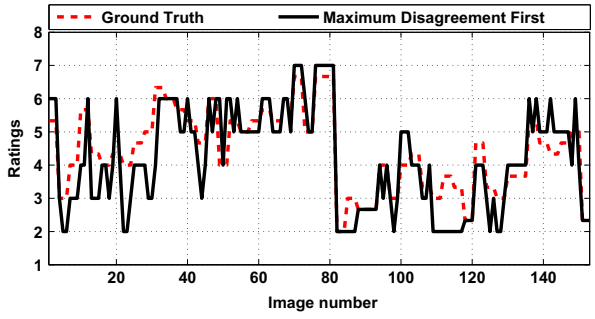

Fig. 13. Comparing the enhanced ratings generated from the maximum disagreement first method with the ground truth ratings.

Since the LEM operates on a particular dataset, it is possible that one sample might converge to different enhanced ratings when it is a member of a different dataset. Obviously this is not desired, and therefore we design experiments to explore this potential issue. On the 2013 dataset, we generate five random subsets of data with a different number of images, and apply LEM based on maximum disagreement first to each subset. Fig. 14 shows the label enhancement results for all five subsets, and the bottom row shows the standard deviation of the enhanced ratings of common samples across five sets. An average standard deviation of 0.41 is obtained over all common samples. Therefore, we can observe that the dependency of enhanced ratings to a particular dataset composition is relatively low, and it seems that the enhanced ratings are moving toward the noisy-free labels of the samples.

Fig. 12 shows that the larger separability can be achieved using the enhanced ratings on the dataset where LEM is applied. The next step is to validate that if we learn an enhanced CLS Rater from the enhanced ratings and apply it to an unseen dataset, whether a larger separability can still be observed. To test this generalization capability, using the training set we learn four CLS Raters based on four labels, the ground truth ratings and the enhanced labels with each of three sample selection functions. Each CLS Rater is applied to the testing set, and based on the estimated ratings all testing samples can be grouped into multiple classes. Then we calculate the eigenvalues of the matrix $\mathbf{\Sigma}^{-1}\mathbf{\Sigma}_b$, where $\mathbf{\Sigma}$ and $\mathbf{\Sigma}_b$ are computed as in Algorithm 1. By repeating this experiment on ten random partitions of training and testing sets, we show the distribution of top eigenvalues in Fig. 15. Since larger eigenvalues indicate high linear separability among the classes, the result demonstrates that the enhanced CLS Rater is able to make the unseen testing set more separable and less confusing between consecutive rating levels. Also, among the three sample selection functions, the maximum disagreement first method seems to have a minor advantage over the others.

*Regression results:*Using the 2013 dataset, we evaluate a diverse set of regression methods belonging to three categories: (1) functional regression (SVM [36], Least Median Squared Linear (LMS)

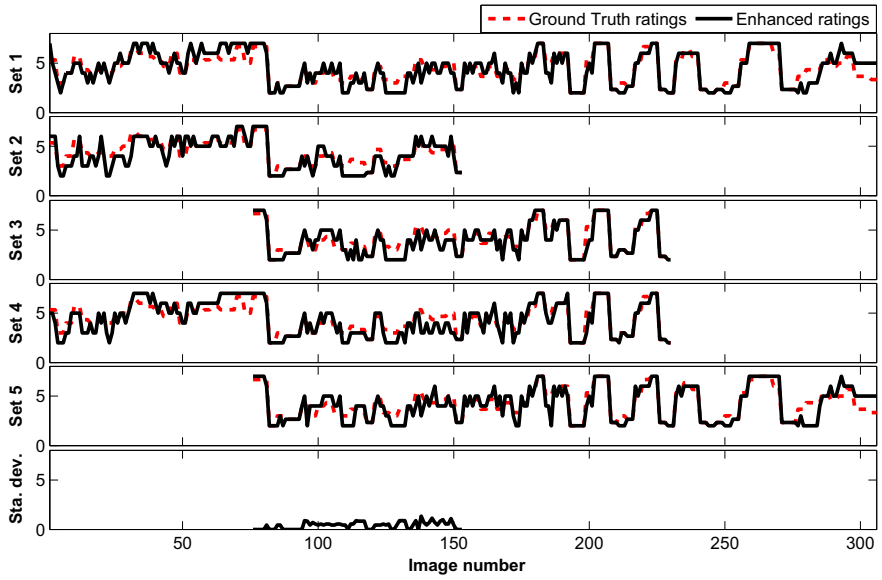

Fig. 14. Convergence analysis of LEM. Rows 1–5 represent the results of applying LEM to different subsets of 2013 dataset. Row 6 is the standard deviation of the enhanced ratings of common samples in five subsets.

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021

[37], and Linear), (2) decision tree learning-based regression (M5P) [35], and (3) rule learning-based regression (M5Rules) [38]. We use bagging with each of these methods to enhance their predictive abilities. To remove the bias in coding, we utilize the standard regression implementations in [33]. Table 2 shows the results where the mean and standard deviation of rating errors are computed from five random partitions of the 2013 dataset. When no "LEM" is used, both the training and testing are based on the ground truth ratings, i.e., the average of three ratings.

We observe that while features at different superpixel scales are preferred by different regression methods, the fused feature ($\mathbf{S}^{all}$) achieves the best performance regardless of the method. Also in general M5P performs the best among all regression methods. Therefore, our CLS Rater utilizes the fused feature with an M5P regressor. The baseline method to compare with our HoI feature is the well-known BoW features [30] based on the 35 color codewords and 256 LBP codewords of each image. As shown in the BoW column of Table 2, none of the regression methods based on BoW are superior to CLS Rater.

By using the LEM, we evaluate the performance of enhanced CLS Rater. Since the enhanced CLS Rater is trained on the enhanced labels, its rating error is also computed w.r.t. the enhanced labels on the testing set, which is obtained by applying the LEM to the testing set. It can be seen that the fused feature with a M5P regressor is still superior to other regressors or other features. Also, in almost all cases, the enhanced CLS Rater has smaller rating errors than the original CLS Rater, with the minimum error reduced from 0.65 to 0.59. On one hand, this superiority indicates that the enhanced CLS Rater can predict ratings more consistently and with less confusion. On the other hand, the reduced error is especially encouraging when the labels in the training set and testing set are *independently* enhanced via LEM. Furthermore, note that the improvement margin of the enhanced CLS Rater is larger for linear regressors (LMS or Linear). A group of test images along with the three ratings are illustrated in Fig. 16, where each column shows three samples with manual ratings being very similar. We observe that the plants in the same column may show different resistances to CLS, thus assign them with the same manual rating indicating the noisy labels from the experts. For example, the image in the first row and third column has an assigned manual rating of 5.3, yet its resistance is more similar to plants in the second column. Therefore, it is desired that the CLS Rater and enhanced CLS Rater predict the ratings of 4.0 and 4.1 respectively. While this is the case of an overrating from the experts, there are also cases of underratings, such as the example at the second row and second column.

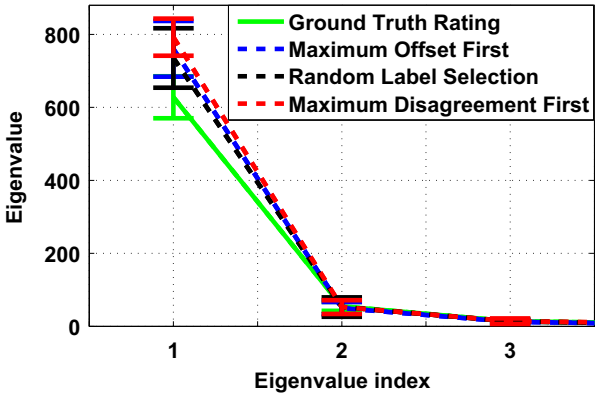

Fig. 15. Eigenvalues of the LDA on the testing set when the CLS Rater is trained with different labels.

Finally, we also explore the scenario of evaluating the enhanced CLS Rater w.r.t. the ground truth ratings. Trained with M5P regressor on the enhanced labels from the maximum disagreement first function has an average rating error of 0.83. Clearly this is an unfair scenario since training and testing are based on different types of labels, i.e., train on the enhanced labels while evaluating w.r.t. the ground truth ratings. However, this relatively small rating error is a good indication that the LEM is indeed updating labels according to disease levels.

We further explore how the regression methods perform w.r.t. different types of appearance features, i.e., color and LBP. As shown in Fig. 17, when learning the regressor with ground truth ratings, fusing color and LBP features improve the system performance for various regression methods. Note that the enhanced CLS Rater also uses the combined color and LBP features. However, M5P and M5Rules perform well using color alone, and fusing with LBP has no noticeable improvement in the rating error. Moreover, when combining color and LPB using the enhanced CLS Rater, all regressors have substantially improved to almost the same high performance, i.e., around 0.6 error rate. In other words, when using enhanced CLS Rater, the choice of regression methods is less important, which allows us to use a more efficient and simple regressor, yet still achieving the high performance.

CLS Rater vs. expert rating: In general, it takes about five seasons to train an unskilled individual for rating CLS disease and at least one season to train a pathologist. However, it is well known that human experts tend to provide inconsistent rating for CLS as discussed earlier. Hence, it is interesting to compare the rating error of CLS Rater to the error observed in human expert rating. The minimum rating error is 0.65 for CLS Rater, and 0.59 for enhanced CLS Rater, as shown in Table 2. For comparison, we calculate the standard deviation of expert rating using the same equation as our system error e, i.e., $e^h = (\frac{1}{3K}\sum_i\sum_j \|\bar{r}_i - r_i^j\|^2)^{1/2}$. Based on the same five partitions in computing e, the standard deviation of expert rating e^h is 1.31 ± 0.08. The superior consistency of our system, i.e., with or without the LEM, over the human experts indicates the great potential of applying CLS Rater in practices.

CLS Rater across the years: Ideally the CLS Rater learned from data samples of one year can be repeatedly utilized in the real field

Table 2
Rating error (e) at different superpixel scales.

Regression	LEM	$\mathbf{S}^{500}$	$\mathbf{S}^{1500}$	$\mathbf{S}^{2500}$	$\mathbf{S}^{3500}$	*BoW*	$\mathbf{S}^{all}$
M5P	No	0.90 ± 0.03	0.91 ± 0.04	0.88 ± 0.03	0.69 ± 0.04	0.73 ± 0.02	$\mathbf{0.65 \pm 0.03}$
	Yes	0.72 ± 0.03	0.73 ± 0.06	0.75 ± 0.02	0.62 ± 0.02	0.72 ± 0.02	$\mathbf{0.59 \pm 0.04}$
SVM	No	1.10 ± 0.09	1.12 ± 0.05	1.05 ± 0.09	0.81 ± 0.08	0.83 ± 0.03	0.75 ± 0.04
	Yes	0.69 ± 0.03	0.79 ± 0.05	0.79 ± 0.07	0.63 ± 0.02	0.79 ± 0.08	0.60 ± 0.02
Linear	No	1.46 ± 0.17	1.40 ± 0.11	1.06 ± 0.13	0.91 ± 0.03	0.83 ± 0.04	0.82 ± 0.06
	Yes	0.67 ± 0.02	0.78 ± 0.02	0.75 ± 0.02	0.64 ± 0.01	0.79 ± 0.03	0.62 ± 0.01
M5Rules	No	0.92 ± 0.04	0.92 ± 0.05	0.89 ± 0.03	0.70 ± 0.03	0.74 ± 0.03	0.66 ± 0.05
	Yes	0.66 ± 0.03	0.73 ± 0.03	0.76 ± 0.01	0.65 ± 0.01	0.78 ± 0.04	0.61 ± 0.02
LMS	No	1.35 ± 0.42	1.41 ± 0.17	0.95 ± 0.04	0.94 ± 0.12	0.85 ± 0.03	0.70 ± 0.04
	Yes	0.66 ± 0.01	0.81 ± 0.08	0.76 ± 0.02	0.64 ± 0.01	0.88 ± 0.03	0.62 ± 0.05

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021

in subsequent years. Therefore, it is important to evaluate the generalization capability of CLS Rater on a testing set that is collected from a different year as the training set. For this purpose, we use the 2013 dataset as the training set and the 2014 dataset as the testing set. The labels for the training set are either from one expert (who also labeled the 2014 dataset), or the enhanced labels by LEM based on the maximum offset first function, which result in the CLS Rater and the enhanced CLS Rater respectively. Similarly, two types of labels exist for the testing set. As shown in Fig. 18, each box represents the manual ratings of the field at a

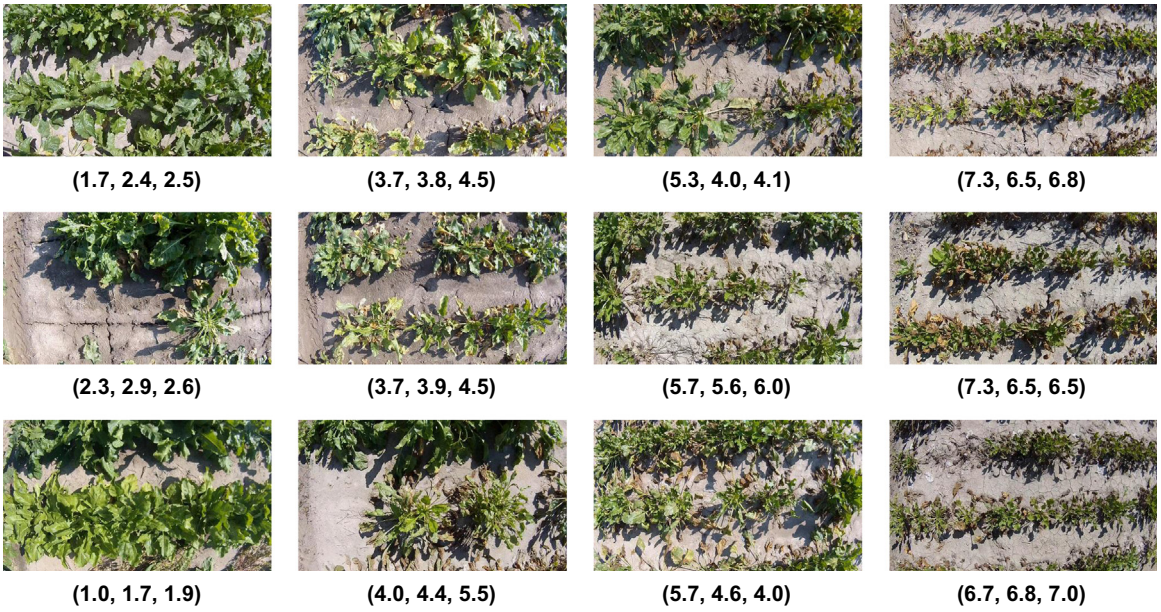

Fig. 16. Examples of testing images with the CLS ratings in the form (a–c), where (a) is the manual rating assigned from the experts, (b) is the CLS Rater prediction and (c) is the enhanced CLS Rater prediction.

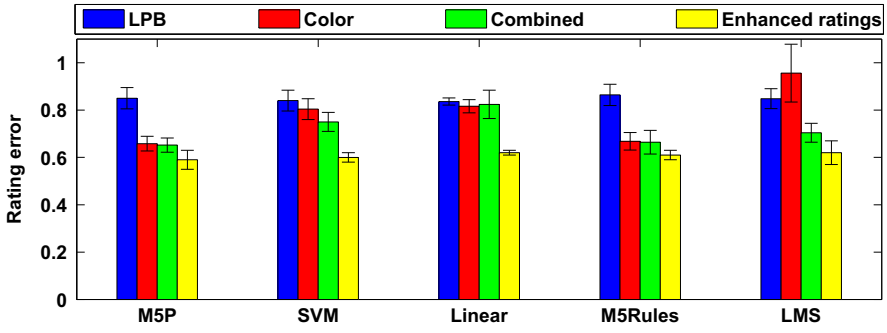

Fig. 17. Regression performance with different feature types and labels.

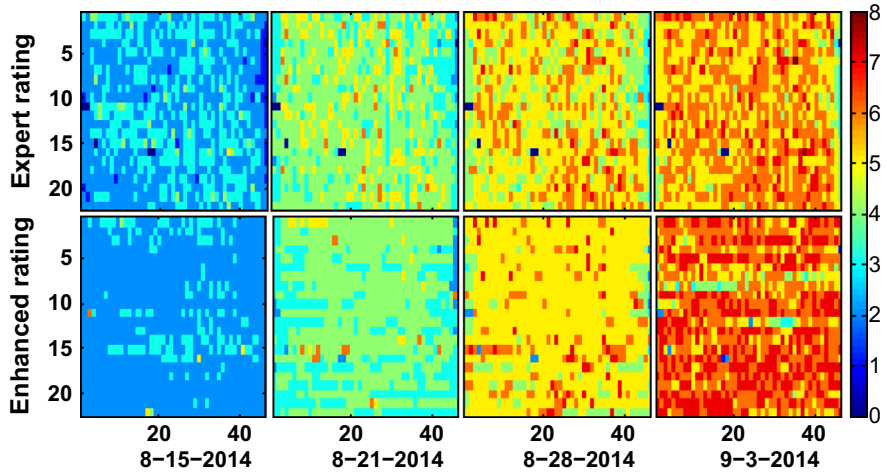

Fig. 18. Ratings of a plant field in four days of 2014. The first row shows manual ratings from one expert. The second row shows the enhanced ratings by LEM.

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021

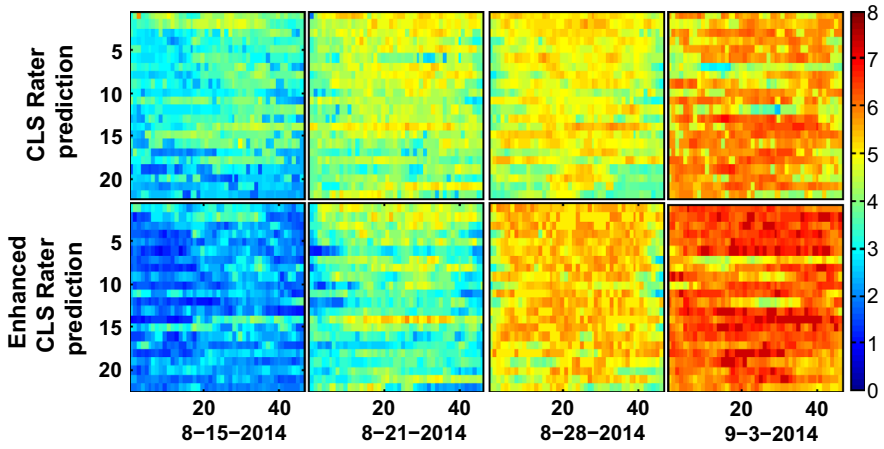

Fig. 19. Automatically predicted ratings of the plant field in four days of 2014. The first row is the predictions of the CLS Rater trained on the manual ratings. The second row is the predictions of the enhanced CLS Rater trained on the enhanced ratings.

specific day, which is made of 22×46 subunits, where 22 is the number of field lines and 46 is the number of evenly sampled images along each field line. Note that only one expert provides ratings for the four chosen days to record various disease ratings. The second row shows the enhanced ratings after applying LEM using the maximum offset first function.

By applying CLS Rater and the enhanced CLS Rater on the testing set, we obtain the rating results in Fig. 19. We can see that the CLS Rater was not very successful at predicting very high or low rating in "Sept 3" and "Aug 15". Moreover, it appears that cultivars located on lines 7 and 8 have relatively higher resistance to the CLS disease in comparison to other lines. An average rating error of the CLS Rater is 1.26 w.r.t. the manual ratings, while an average rating error of 1.05 is achieved for the enhanced CLS Rater w.r.t. the enhanced ratings. Therefore, similar to Table 2, we see again that the enhanced CLS Rater provides more consistent rating, even for across-year experiments. The reason for observing higher rating errors than Table 2 is twofold: (i) the appearance variation between the years; (ii) the imprecise mapping of GPS data to video frames, and hence assigning manual rating to frames. Nevertheless, the rating error in this challenging across-year experiment is still smaller than the standard deviation of the combined expert ratings in 2013.

6. Conclusions

This paper introduced a novel computer vision system, CLS Rater, which uses real field plant images for the automatic rating of the CLS disease in sugar beet plants. Our CLS Rater utilizes a novel HoI feature to represent the local characteristics of superpixels at the image level and predicts the rating with an error of 0.59, which is substantially more consistent in comparison to manual ratings performed by human experts. We tested our system on a real field of sugar beet plants under different lighting and weather conditions for two consecutive years. We also addressed the issue of the noisy expert labels by developing an LEM to enhance the labels. One future direction is to learn CLS Rater from a set of image pairs each ranked by their disease severity, using approaches such as boosted rank learning [39]. Furthermore, since the technical approach of CLS Rater is very general, it is potentially applicable to disease monitoring of other plants and a variety of precision agriculture applications in the real field.

Conflict of interest

None declared.

Acknowledgements

This project was sponsored in part by Michigan Sugar Company Competitive Grant (#MSC-13-08) and Project Greeen (#GR14-034). The authors thank Tom Goodwill and Bill Niehaus for labeling images with CLS.

Appendix A. Supplementary data

Supplementary data associated with this paper can be found in the online version at http://dx.doi.org/10.1016/j.patcog.2015.11.021.

References

[1] ⟨http://www.ers.usda.gov/topics/crops/sugar-sweeteners/background.aspx#.UpfNwuIwBZ8⟩.
[2] J. Khan, L.d. Rio, R. Nelson, V. Rivera-Varas, G. Secor, M. Khan, Survival, dispersal, and primary infection site for *Cercospora beticola* in sugar beet, Plant Dis. 92 (5) (2008) 741–745.
[3] A.K. Mahlein, U. Steiner, H.W. Dehne, E.C. Oerke, Spectral signatures of sugar beet leaves for the detection and differentiation of diseases, in: European Conference on Precision Agriculture, vol. 11, 2010, pp. 413–431.
[4] E. Ruppel, J. Gaskill, Techniques for evaluating sugarbeet for resistance to *Cercospora beticola* in the field, Am. Soc. Sugar Beet Technol. J. (1971).
[5] S. Bauer, F. Korc, W. Förstner, Investigation into the classification of diseases of sugar beet leaves using multispectral images, in: European Conference on Precision Agriculture, vol. 9, 2009, pp. 229–238.
[6] T. Rumpf, A.-K. Mahlein, U. Steiner, E.-C. Oerke, H.-W. Dehne, L. Plümer, Early detection and classification of plant diseases with support vector machines based on hyperspectral reflectance, Comput. Electron. Agric. 74 (1) (2010) 91–99.
[7] H. Al-Hiary, S. Bani-Ahmad, M. Reyalat, M. Braik, Z. ALRahamneh, Fast and accurate detection and classification of plant diseases, Int. J. Comput. Appl. 17 (1) (2011) 31–38.
[8] W. Shen, Y. Wu, Z. Chen, H. Wei, Grading method of leaf spot disease based on image processing, in: International Conference on Computer Science and Software Engineering, vol. 6, IEEE, Hubei, China, 2008, pp. 491–494.
[9] J. Vereijssen, J. Schneider, A. Termorshuizen, M. Jeger, Comparison of two disease assessment methods for assessing Cercospora leaf spot in sugar beet, Crop Prot. 22 (1) (2003) 201–209.
[10] B. Fulkerson, A. Vedaldi, S. Soatto, Class segmentation and object localization with superpixel neighborhoods, in: Proceedings of International Conference on Computer Vision (ICCV), IEEE, Kyoto, Japan, 2009, pp. 670–677.
[11] J. Tighe, S. Lazebnik, Superparsing: scalable nonparametric image parsing with superpixels, in: Proceedings of European Conference on Computer Vision (ECCV), Springer, Crete, Greece, 2010, pp. 352–365.

[12] H. Liu, Y. Qu, Y. Wu, H. Wang, Class-specified segmentation with multi-scale superpixels, in: Proceedings of Asian Conference on Computer Vision Workshops (ACCV), Springer, Singapore, 2013, pp. 158–169.
[13] M.J. Afridi, X. Liu, J.M. McGrath, An automated system for plant-level disease rating in real fields, in: Proceedings of International Conference on Pattern Recognition (ICPR), IEEE, Stockholm, Sweden, 2014, pp. 148–153.
[14] L. Hanson, T. Goodwill, J. McGrath, Beta PIs from the USDA-ARS NPGS evaluated for resistance to *Cercospora beticola*, 2013, Plant Dis. Manag. Rep. 8 (2014) FC170.
[15] R. Zhou, S. Kaneko, F. Tanaka, M. Kayamori, M. Shimizu, Image-based field monitoring of Cercospora leaf spot in sugar beet by robust template matching and pattern recognition, Comput. Electron. Agric. 116 (2015) 65–79.
[16] Z. Li, X.M. Wu, S.F. Chang, Segmentation using superpixels: a bipartite graph partitioning approach, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Providence, RI, 2012, pp. 789–796.
[17] A. Kae, K. Sohn, H. Lee, E. Learned-Miller, Augmenting CRFs with Boltzmann machine shape priors for image labeling, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Columbus, OH, 2013, pp. 2019–2026.
[18] Z. Hao, Q. Wang, H. Ren, K. Xu, Y.K. Seong, J. Kim, Multiscale superpixel classification for tumor segmentation in breast ultrasound images, in: Proceedings of International Conference on Image Processing (ICIP), IEEE, Orlando, FL, 2012, pp. 2817–2820.
[19] B. Frénay, M. Verleysen, Classification in the presence of label noise: a survey, IEEE Trans. Neural Netw. Learn. Syst 25 (5) (2014) 845–869.
[20] N. Manwani, P. Sastry, Noise tolerance under risk minimization, IEEE Trans. Cybern. 43 (3) (2013) 1146–1151.
[21] F.A. Breve, L. Zhao, M.G. Quiles, Semi-supervised learning from imperfect data through particle cooperation and competition, in: International Joint Conference on Neural Networks, IEEE, 2010, pp. 1–8.
[22] J.-w. Sun, F.-y. Zhao, C.-j. Wang, S.-f. Chen, Identifying and correcting mislabeled training instances, in: Future Generation Communication and Networking, vol. 1, IEEE, Jeju, South Korea, 2007, pp. 244–250.
[23] D. Gamberger, N. Lavrac, S. Dzeroski, Noise detection and elimination in data preprocessing: experiments in medical domains, Appl. Artif. Intell. 14 (2) (2000) 205–223.
[24] S. Verbaeten, A. Van Assche, Ensemble methods for noise elimination in classification problems, in: Multiple Classifier Systems, Springer, 2003, pp. 317–325.
[25] A. Malossini, E. Blanzieri, R.T. Ng, Detecting potential labeling errors in microarrays by data perturbation, Bioinformatics 22 (17) (2006) 2114–2121.
[26] J. Thongkam, G. Xu, Y. Zhang, F. Huang, Support vector machine for outlier detection in breast cancer survivability prediction, in: Advanced Web and Network Technologies, and Applications, Springer, Berlin Heidelberg, 2008, pp. 99–109.
[27] X. Zeng, T. Martinez, A noise filtering method using neural networks, in: IEEE International Workshop on Soft Computing Techniques in Instrumentation, Measurement and Related Applications, IEEE, 2003, pp. 26–31.
[28] D. Wang, X. Tan, Robust distance metric learning in the presence of label noise, in: The 28th AAAI Conference on Artificial Intelligence, Québec, Canada, 2014, pp. 1321–1327.
[29] M.Y. Liu, O. Tuzel, S. Ramalingam, R. Chellappa, Entropy rate superpixel segmentation, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Colorado Springs, CO, 2011, pp. 2097–2104.
[30] L. Fei-Fei, P. Perona, A Bayesian hierarchical model for learning natural scene categories, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, IEEE, San Diego, CA, 2005, pp. 524–531.
[31] O. Arandjelović, Colour invariants under a non-linear photometric camera model and their application to face recognition from video, Pattern Recognit. 45 (7) (2012) 2499–2509.
[32] T. Ojala, M. Pietikäinen, D. Harwood, A comparative study of texture measures with classification based on featured distributions, Pattern Recognit. 29 (1) (1996) 51–59.
[33] I.H. Witten, E. Frank, Data Mining: Practical Machine Learning Tools and Techniques, The Morgan Kaufmann Series in Data Management Systems, 3rd edition, 2011.
[34] R.J. Quinlan, Learning with continuous classes, in: Proceedings of the Fifth Australian Joint Conference on Artificial Intelligence, vol. 92, World Scientific, 1992, pp. 343–348.
[35] Y. Wang, I.H. Witten, Induction model trees for continuous classes, in: European Conference on Machine Learning, Prague, Czech Republic, 1997, pp. 128–137.
[36] S.K. Shevade, S.S. Keerthi, C. Bhattacharyya, K.R.K. Murthy, Improvements to the SMO algorithm for SVM regression, IEEE Trans. Neural Netw. 11 (5) (2000) 1188–1193.
[37] P.J. Rousseeuw, A.M. Leroy, Robust Regression and Outlier Detection, Wiley, 1987.
[38] G. Holmes, M. Hall, E. Prank, Generating Rule Sets from Model Trees, Springer, 1999.
[39] H. Wu, X. Liu, G. Doretto, Face alignment via boosted ranking models, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Anchorage, AL, 2008, pp. 1–8.
[40] R. Zhou, S. Kaneko, F. Tanaka, M. Kayamori, M. Shimizu, Disease detection of cercospora leaf spot in sugar beet by robust template matching, Computers and Electronics in Agriculture 108 (2014) 58–70.
[41] X. Liu, P. Tu, F. Wheeler, Face model fitting on low resolution images, in: Proc. British Machine Vision Conf. (BMVC), Edinburgh, UK, 2006, pp. 1079–1088.
[42] M.J. Afridi, C. Liu, C. Chan, S. Baek, X. Liu, Image segmentation of mesenchymal stem cells in diverse culturing conditions, IEEE Winter Conf. on Applications of Computer Vision (WACV), Steamboat Springs CO, 2014.
[43] M.J. Afridi, X. Liu, E. Shapiro, A. Ross, Automatic in vivo cell detection in MRI, in: Proc. Int. Conf. Medical Image Computing and Computer Assisted Intervention (MICCAI), Springer, 2015, pp. 391–399.
[44] Y. Atoum, S. Srivastava, X. Liu, Automatic feeding control for dense aquaculture fish tanks, IEEE Signal Processing Letters 22 (8) (2015) 1089–1093.

Yousef Atoum received a B.S. degree in Computer Engineering from Yarmouk University, Jordan, in 2009, and a M.S. degree in Electrical and Computer Engineering from Western Michigan University, in 2012. He is currently a Ph.D. student in the Electrical and Computer Engineering department at Michigan State University, with interests in object tracking, computer vision and pattern recognition.

Muhammad Jamal Afridi is a Ph.D. student in the Department of Computer Science and Engineering at Michigan State University. He received his B.E. degree in Mechatronics from National University of Sciences and Technology, Pakistan, in 2009. He worked as a Research Engineer at nexGIN RC before joining the Ph.D. program at MSU, in Fall 2011. His research areas include pattern recognition, medical image analysis, computer vision and robotics.

Xiaoming Liu is an Assistant Professor in the Department of Computer Science and Engineering at Michigan State University (MSU). He received the B.E. degree from Beijing Information Technology Institute, China and the M.E. degree from Zhejiang University, China, in 1997 and 2000 respectively, both in Computer Science, and the Ph.D. degree in Electrical and Computer Engineering from Carnegie Mellon University, in 2004. Before joining MSU, in Fall 2012, he was a Research Scientist at General Electric Global Research Center. His research areas are face recognition, biometrics, image alignment, video surveillance, computer vision and pattern recognition. He has authored more than 90 scientific publications, and has filed 22 U.S. patents. He is a member of the IEEE.

J. Mitchell McGrath is a Research Geneticist with the USDA Agricultural Research and an Adjunct Professor in the Department of Plant, Soil, and Microbial Sciences at Michigan State University. He received his B.S. degree from the University of Massachusetts-Amherst and Ph.D. from the University of California-Davis in 1989. He held postdoctoral fellowship positions at the University of Michigan-Ann Arbor and the University of Wisconsin-Madison prior to joining his current position with responsibilities in sugar beet genetics, genomics, and germplasm enhancement. He has published over 200 scientific articles and technical reports.

Linda E. Hanson is a research plant pathologist with the USDA, Agricultural Research Service and an adjunct Associate Professor in the Department of Plant, Soil and Microbial Science at Michigan State University (MSU). She received her B.S. in Botany from University of Washington, her M.S. in Plant Pathology from MSU, and her Ph.D. in Plant Pathology from Cornell University. She has been with USDA-ARS since 1997 and has worked at three different locations. Her research focus is on fungi that interact with plants, particularly soil-borne fungi and plant pathogens.

Please cite this article as: Y. Atoum, et al., On developing and enhancing plant-level disease rating systems in real fields, Pattern Recognition (2016), http://dx.doi.org/10.1016/j.patcog.2015.11.021