

# Leaf Disease Recognition in Vine Plants based on Local Binary Patterns and One Class Support Vector Machines

X.E. Pantazi<sup>1</sup>, D. Moshou<sup>1</sup>, A.A. Tamouridou<sup>1</sup>, S. Kasderidis<sup>2</sup>

<sup>1</sup> Aristotle University, School of Agriculture, Agricultural Engineering Laboratory, Thessaloniki 54124, Greece, Tel: +302310998264, Fax: +302130998729, e-mail: renepantazi@gmail.com

<sup>2</sup> NOVOCAPTIS, i4G Business Incubator, P.O. Box 22461, Antoni Tritsi 21 Thessaloniki 55102, Greece, Tel: +30 2310 804 970, Fax: +30 2310 804 810

**Abstract.** The current application concerns a new approach for disease recognition of vine leaves based on Local Binary Patterns (LBPs). The LBP approach was applied on color digital pictures with a natural complex background that contained infected leaves. The pictures were captured with a smartphone camera from vine plants. A 32-bin histogram was calculated by the LBP characteristic features that resulted from a Hue plane. Moreover, four One Class Support Vector Machines (OCSVMs) were trained with a training set of 8 pictures from each disease including healthy, Powdery Mildew and Black Rot and Downy Mildew. The trained OCSVMs were tested with 100 infected vine leaf pictures corresponding to each disease which were capable of generalizing correctly, when presented with vine leaf which was infected by the same disease. The recognition percentage reached 97%, 95% and 93% for each disease respectively while healthy plants were recognized with an accuracy rate of 100%.

**Keywords:** image processing, novelty detector, classifier conflict, texture descriptors

## 1 Introduction

Plant recognition is regarded as a puzzling computer vision problem which requires dealing with irregular shapes and textures. Interest in methods for visual classification of plants has grown recently [1] as devices equipped with cameras became ubiquitous, making intelligent field guides, education tools and automation in forestry and agriculture practical. Belhumeur et al [1] presented how using such a system in the field enables a botanist to quickly search vast collections of plant species - a process that was time consuming and laborious can now be implemented in seconds. Recognition of leaves is usually applied on broad leaves and needles are treated separately. Several techniques have been introduced for leaf description, utilizing shape features and colour features. The leaf recognition approach which was presented by Fiel and Sablatnig [2] was based on a Bag of Words model with SIFT descriptors and reached

93.6% accuracy on a leaf dataset of 5 Austrian tree species. Kumar et al. [3] proposed Leafsnap, which is a computer vision system capable of identifying plant species. They introduced a pre-filter on input images, numerous speed-ups and additional post-processing within the segmentation algorithm, the use of a simpler and more efficient recognition algorithm based on curvature instead of Inner Distance Shape Context (IDSC); an image dataset with more images, and an interactive system that can be used by non-expert users. The final product was an electronic field guide, available as a free mobile app for iOS devices. Even though the app runs on iPad and iPhone devices, the images of the leaves have to be processed on a server. This requires an internet connection to achieve recognition, so it is problematic for use in natural areas without access to internet or a limited bandwidth connection. Another severe limitation concerns the need to obtain photos of the leaves with a white background. The contribution of the current paper concerns an effective combination of pre-processing stages and One Class Classifiers applied on complex natural scenes with aim to discover early stages of infections in field grown crops enabling faster and more effective prevention of crop epidemics that threaten food supply and quality. The presented approach consists of training one classifier per disease and presenting the feature vector to a committee of one class classifiers which decide autonomously by having been trained only on target data. In the event that there is a multiple activations of one class classifiers the resultant conflict is overcome by an original conflict resolution method that utilizes the proximity of the nearest support vector to make an accurate decision which reached 97% for testing with unforeseen examples. The mentioned technique can generalize in multiple crops for three diseases while it has been trained only with vine leaves.

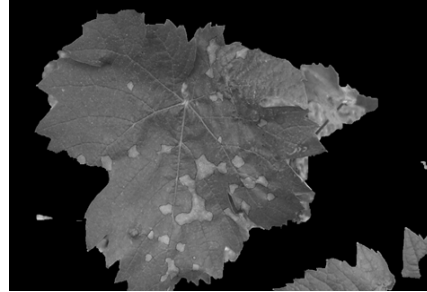
## **2 Materials and Methods**

### **2.1 Image Segmentation**

The segmentation operation needed to extract the leaf that has the symptoms was implemented by making use of the GrabCut algorithm. The basic steps for implementing the GrabCut algorithm in the leaf segmentation [4] are presented as follows: The initial information provided about the foreground and the backgrounds are given by the user is a rectangular selection around the object of interest. Pixels that reside outside this selection are considered as known background and other pixels inside are marked as unknown. From this data we wish to create a model which can be used to decide if the unknown pixels are foreground or background. In the GrabCut algorithm this is achieved by creating  $K$  components of multivariate Gaussian Mixture Models (GMM) corresponding to the two regions.  $K$  components are created from the known background and  $K$  components correspond to the region that could belong to the foreground, giving a total of  $2 \cdot K$  components. The GMM components have the same dimensions as the color space and they are defined based on the color statistics from each cluster. In order to obtain a good segmentation we need to add components with low variance because this makes cluster separation easier.



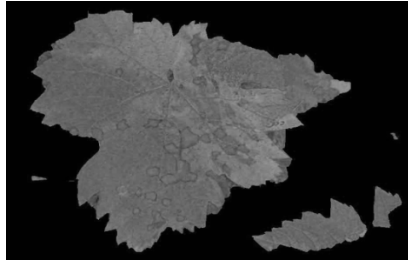
**Fig. 1.** The original image of a vine leaf that has been fed to the model of powderyHSV



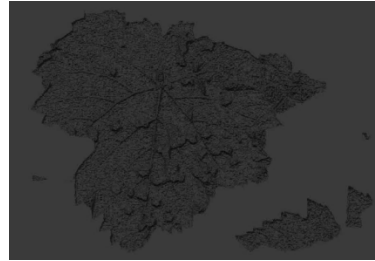
**Fig. 2.** Segmentation of vine leaf by using the GrabCut algorithm on the original image.

## 2.2 The Local Binary Patterns (LBPs)

The local binary pattern is a simple but very effective texture operator which tags the pixels of an image by thresholding the region of each pixel and considers the outcome as a binary number. The LBP scheme can be seen as a fusing approach to the traditionally deviating statistical and structural representations of texture analysis. Perhaps the most significant property of the LBP operator in actual applications is its invariance beside monotonic gray level deviations caused, e.g., by illumination differences. Another likewise important is its computational effortlessness, which makes it possible to analyze images in puzzling real-time situations. The original local binary pattern operator, presented by Li, J., et al.[5] and Llado, et al.[6] was founded on the conjecture that texture has locally two balancing aspects, a pattern and its strength. The operator works in a  $3 \times 3$  locality, using the center value as a threshold. An LBP code is created by multiplying the thresholded values with loads given by the corresponding pixels, and summing up the outcome. As the neighborhood comprises of 8 pixels, a total of  $2^8 = 256$  dissimilar labels can be obtained dependent on the relative gray values of the epicenter and the pixels in the locality. The contrast measure (C) is acquired by subtracting the average of the gray levels under the center pixel after that of the gray levels above (or equal to) the center pixel. If the eight thresholded neighbors of the middle pixel have an identical value (0 or 1), the value of contrast is set to zero. The distributions of LBP codes are used as features in grouping or segmentation. The local binary pattern (LBP) operator is an image operator which converts an image into a group or image of integer labels relating to small-scale appearance of the image. These labels or their statistics, and one of the most commonly used is the histogram, are then applied for further image analysis. The most frequently used versions of the operator are aimed for monochrome still images nevertheless it has been extended likewise for color (multi channel) images and also for videos and volumetric data. An example of an H-plane image and corresponding LBP image are shown in Figures 3 and 4.



**Fig. 3.** H-plane of the vine leaf after the background of the initial image has been removed.



**Fig. 4.** The corresponding LBP image of the vine leaf.

### 2.3 The creation of the LBP histogram

The segmented image acquired from the GrabCut algorithm is utilized to create different monochrome descriptions as the LBP algorithms operates in single channels. Three diverse options were tried, the gray level, Hue channel and saturation channel. The most performing was the Hue channel LBP histogram as the results showed in the recognition of diseased plants against healthy plants. The image of the Hue channel is shown in Fig.4. By applying the LBP transformation on the Hue image the textural features corresponding to contrast and hue change are exploited by labelling neighboring pictures according to the LBP algorithm explained above. The LBP image obtained from this process can be shown in Fig.5. The histogram of the LBP image is constructed according to the frequency of the occurrence of similar values between 0 and 255, a low length histogram containing 32 bins, has been used to partition the range of 255, so as to more rough features instead of local high resolution features which might demonstrate high variability.

### 2.4 One Class Support Vector Machines (OCSVMs)

To support One Class SVM Classification a suitable description of the configuration of SVM as a model to designate target data was presented by Tax and Duin [7] in the form of Support Vector Data Description (SVDD). The objective of SVDD is to learn a decision function in order to predict if an example is actually a target or an outlier. SVDD is created based on the supposition that targets are bounded by a close boundary in the feature space. The OCSVM creates a model from performing adjustment by normal data according to the SVDD description. At the following stage, orders test data based on the deviance from normal training data as being either normal or outlier [9].

### 2.4.1 One-Class Support-Vector-Learning for Multi Class Problems

The described one-class approach can be applicable to multiclass problems in the following way. It is anticipated that data samples from  $k$  different classes are given. Let  $\Omega = \{1, \dots, k\}$  denotes the set of classes. For each distinct class  $t \in \Omega$  a one-class classifier  $f_t$  is created. Specifically,  $k$  centres and corresponding radii  $R_t$  are considered. To determine the class relationship of a new undetected pattern  $z$  all decision functions are calculated in parallel leading to class memberships. In the classification stage of an undetected input pattern  $z$  three probable resulting circumstances have to be distinguished:

1. If just one classifier shows its bond for the new pattern, the resultant class label is the class membership of this classifier.
2. If more classifiers raise a claim for this pattern, a conflict situation has arrived.
3. On the other hand it is possible that no classifier matches, so in this case we can characterize this outcome as an outlier situation.

The problematic with such uncertain (2 and 3) classification assignments can be disentangled in two ways: Discarding the classification of the novel pattern or finding the matching class label through an outlier-/conflict processing approach. In the subsequent section we will present a strategy to crack such problems [9].

### 2.5 Nearest-Support-Vector Strategy

The class label of a novel pattern  $z$  is assessed by comparing the remoteness to the support vectors of the involved classifiers. The classifier that has the nearest support vector defines the class membership of  $z$  assuming that are the support vectors, and  $i_t$  is the amount of support vectors of classifier  $t$ .

$$SV(t) = \left\{ \hat{x}_1^t, \dots, \hat{x}_{i_t}^t \right\} \quad (1)$$

$$f(z) = \operatorname{argmin}_{t \in C} \left\| \hat{x}_i^t - z \right\| \quad (2)$$

The decision function is then demarcated by where the minimum will be taken over all the support vectors of classes after the candidate set [8].

## 3 Results and Discussion

### 3.1 Application of LBP in disease recognition of infected vine leaves

Initially, smartphone photo is obtained from infected plants. The standard requirement is that a leaf demonstrating symptoms of powdery mildew is situated in the center of the image. To obtain only the leaf area and subtract the background, a segmentation

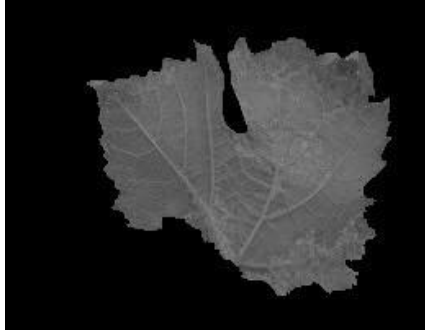
operation is performed. Then, the HSV image is obtained and the Hue plane is shown in Fig.5. The corresponding LBP image obtained from the Hue plane of Fig. 5 is shown in Fig.6. Then the LBP image is used in order to calculate the histogram with 32 bins. The global LBP histogram equation is given as follows:

$$H(i) = \sum_{x,y \in \text{image}} I(f(x,y) = i) \quad (3)$$

where  $i=1,\dots,32$ ,  $I$  represents the indicator function used for thresholding and  $f(x,y)$  represents the LBP label. The histograms have been used as input in One Class Classifiers in the form of One Class SVMs in order to recognize specific leaf samples as a target class and treat healthy samples and other samples from different diseases as outliers. The Kernel function that has been used was RBF. The default width of the tube was  $\epsilon=0.1$ . Eight samples were enough for building powdery mildew model in vines that can recognize this disease. A similar procedure was followed for healthy leaves of vines, by building an one Class SVM model that could recognize only healthy leaves. Cross testing of each One Class SVM with leaves from the other category showed perfect classification for the calibration sets and additional samples that were picked randomly from the internet and smartphone. In the current work OpenCV Computer Vision Library Version 2.4.11 has been used for image processing and classifier development.

The One Class SVM that has been trained on vines to recognize powdery mildew is called ‘powderyHSV’ and it has been stored in XML format. It has been tested in various infected vine leaves successfully. Here, it is tested on vine leaves.. Initially, the powderyHSV One Class SVM recognizes the disease successfully (Fig.7). However, in order to test the selectivity of the model, a second model has been utilized, that is called ‘blackrotHSV’ and recognizes black rot (Fig.8). As it is shown below (Fig.7), both models are activated from the observed symptoms as belonging to the sphere of confidence of one Class SVM of each model. Conflict resolution takes place by calculating the distances from the support vectors of the test image feature vector (Hue histogram) (Fig.7). The result of the conflict resolution is that the actual model, it is identified to be powderyHSV, which is correct (Fig.7).

One hundred samples from each class were used (actual infection from powdery mildew, black rot and downy mildew). Without applying nearest vector based conflict resolution, it was impossible to achieve correct classification in 50% of the cases. Through conflict resolution, 100% percentage was reached in classifying both infected and healthy state. The numbers reached 97% for powdery mildew identification, compared to support vector machine model for black rot and downy mildew. The recognition percentage reached 95% and 93% for black rot and downy mildew respectively. In all disease cases, the test samples came from vine leaves that were not included to the training set. Support Vector Machine training was accomplished, as mentioned above, with 7 samples that all came from vine cultivations’ leaves.



**Fig. 5.** The segmented image has been used to obtain the HSV planes. Here, the Hue plane is shown.



**Fig. 6.** The corresponding LBP image of the vine leaf.



**Fig. 7.** Conflict resolution takes place and the image is labelled correctly as infected with powdery mildew.



**Fig. 8.** Vine leaf image without conflict resolution is classified correctly as Black Rot infected.

This fact alone indicates the high generalization capability of the model, demonstrated and validated through the efficient performance on different vine leaf images that were gathered under varying conditions. The most significant advantage of this method does not only regard the application of One Class Classifiers, but also the highly effective way of discerning between infected leaves and image background. This property derives from the application of the Gaussian mixtures based segmentation method, remarkably effective for isolating image areas of sharp variation in color. The effectiveness of the presented application is a result of the sharp color variation that occurs at infected leaves, in comparison with healthy ones, thus an immediate differentiation is possible. In cases with a complex background, GrabCut algorithm can be trained to recognize a certain foreground not corresponding to the central area of the image. This ability was not required for use in this application, but can be optionally activated according to the criterion of right choice of each infected leaf in examination.

## 4 Conclusions

In the current paper an effective combination of pre-processing stages and One Class Classifiers applied on complex natural scenes is presented. The main aim of the presented research was to discover early stages of infections in vine leaves in order to enable faster and more effective prevention of vine epidemics that threaten food supply and quality. Four One Class Support Vector Machines (OCSVMs) were trained with a training set of 8 pictures from each disease including healthy, Powdery Mildew and Black Rot and Downy Mildew. The trained OCSVMs were tested with 100 infected vine leaf pictures corresponding to each disease. The recognition percentage reached 97%, 95% and 93% for each disease respectively while healthy plants were recognized with an accuracy rate of 100%.

## Acknowledgements

The presented study has been funded by PPP-FI /FRACTALS project.

## References

1. Belhumeur, P.N., Chen, D., Feiner, S., Jacobs, D.W., Kress, W.J., Ling, H., Lopez, I., Ramamoorthi, R., Sheorey, S., White, S., Zhang, L.: Searching the world's herbaria: 16, A system for visual identification of plant species. In: Computer Vision{ECCV 2008. Springer (2008) pp.116-129.
2. Fiel, S., Sablatnig, R. (2011). Automated identification of tree species from images of the bark, leaves and needles. In: Proc. of 16th Computer Vision Winter Workshop, Mitterberg, Austria, pp.1-6.
3. Kumar, N., Belhumeur, P.N., Biswas, A., Jacobs, D.W., Kress, W.J., Lopez, I.C., Soares, J.V.: Leafsnap (2012). A computer vision system for automatic plant species identification. In: Computer Vision, ECCV 2012. Springer, pp.502-516.
4. Rother, C., Kolmogorov, V., Blake, A. (2004). GrabCut: Interactive foreground extraction using iterated graph cuts, ACM Trans. Graph., vol. 23, pp. 309–314.
5. Li, J., Wu, W., Wang, T., Zhang, Y. (2008). One step beyond histograms: Image representation using Markov stationary features. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8.
6. Llado, X., Oliver, A., Freixenet, J., Marti, R., Marti, J.(2009). A textural approach for mass false positive reduction in mammography. Comput. Med. Imaging Graph. 33, 415–422 .
7. Tax, D. and Duin, R. (2004). Support vector data description, Mach. Learn. 54(1), pp.45–66.
8. Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C (2001). Estimating the support of a high-dimensional distribution, Neural computation 13 (7), pp.1443-1471.
9. Sachs, A., Thiel, C., Schwenker, F.(2006). One-class support-vector machines for the classification of bioacoustic time series. ICGST International Journal on Artificial Intelligence and Machine Learning (AIML) 6(4), pp.29–34.