



**HAL**  
open science

## Vineyard leaf roughness characterization by computer vision and cloud computing technics

Thomas Decourselle, Ludovic Journaux, Jean-Claude Simon, Jean-Noël Paoli, Frédéric Cointault, Fan Yang, Johel Miteran

► **To cite this version:**

Thomas Decourselle, Ludovic Journaux, Jean-Claude Simon, Jean-Noël Paoli, Frédéric Cointault, et al.. Vineyard leaf roughness characterization by computer vision and cloud computing technics. International Conference on Agricultural Engineering-AgEng 2010, Sep 2010, Clermont-Ferrand, France. pp.309. hal-00842218

**HAL Id: hal-00842218**

**<https://hal.science/hal-00842218>**

Submitted on 9 Jul 2013

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Vineyard leaf roughness characterization by computer vision and cloud computing technics.

T. Decourselle<sup>1</sup>, L. Journaux<sup>1</sup>, JC. Simon<sup>1</sup>, JN. Paoli<sup>1</sup>, F. Cointault<sup>1</sup>,  
F. Yang<sup>2</sup>, J. Miteran<sup>2</sup>

<sup>1</sup> Agrosup Dijon, 26 Boulevard Docteur Petitjean, BP 87999, 21079 Dijon cedex

<sup>2</sup> Laboratoire LE2I, UFR Sciences et Techniques, BP 47870, 21078 Dijon cedex

t.decourselle@agrosupdijon.fr, l.journaux@agrosupdijon.fr, jc.simon@agrosupdijon.fr,  
jn.paoli@agrosupdijon.fr, f.cointault@agrosupdijon.fr, fanyang@u-bourgogne.fr,  
miteranj@u-bourgogne.fr

---

## Abstract

In the context of vineyard leaf roughness analysis for precision spraying applications, this article deals with its characterization by computer vision and cloud computing techniques. The techniques merge feature extraction, linear or nonlinear dimensionality reduction techniques and several kinds of classification methods. Different combinations are processed and their performances compared in terms of classification error rate, in order to find the best association. However, these combinations are hardly processed because of the lack of computing power and the prohibitive time consumption of the algorithms. To overcome these difficulties, we propose a solution: the use of cloud computing, which considerably improves computing power. We conclude on the well performance of vineyard leaf roughness characterization through the combination of features extraction, dimensionality reduction and classification processed in a cloud computing environment.

**Keywords:** precision viticulture, cloud computing, texture classification

---

## 1. Introduction

Since the development of Precision Agriculture (Robert 1999), many works have been done on the optimization of input in the field so as to reduce the environmental impact and to increase the yield that is the benefit for the farmers. Two specific activities have been especially focused, the fertilization (mineral or organic spreading (Hijazi et al. 2008; Villette et al. 2008) and the spraying (crop/weed discrimination (Yun et al. 2006) for appropriate weeding ...).

Particularly, in the domain of vineyard spraying research, one of the most important objectives is to minimize the volume of phytosanitary products in order to be more environmentally respectful with more effective plant treatments. Thus, the main goal is to be sure that the spray reaches the target in order to reduce losses that occur at the application. Mechanisms of losses by drift are now well known, contrary to runoffs on the leaves. These last ones are related to adhesion mechanisms of liquid on a surface. Specific models have been developed (Forster et al. 2005) and showed that the predominant factor is leaf roughness, for which few robust works and models have been carried out. For example, with hydrophobic surface, "lotus effect" can appear as in the Figure 1.



Figure 1: Representation of lotus effect (photo by William Thielicke)

In a precision viticulture context, our work aims at analysing droplets behaviour on the leaves. Therefore, we have to study vineyard leaf surface and particularly its roughness. Taking the computer vision viewpoint, we propose here to characterize the vineyard leaf roughness through the performance of a combination of spatio-frequential texture feature extraction and classification methods. However, if we consider the methodological aspect of our process, one of most important step corresponds to the dimensionality reduction (DR) of the features in order to reduce redundancies due to significant extracted parameters size. In this context, several kinds of DR approaches have been compared in previous works for other applications, such as multispectral image compression (Journaux et al. 2006). The results coming from these works clearly show that some efficient techniques such as Curvilinear component analysis are very restrictive due to a convergence problem of the cost function minimization. Indeed, such problems are resolved with a second order gradient resolution, such as Broyden-Fletcher-Goldfarb-Shanno (BFGS) or Levenberg-Marquart method (Fletcher 2000). Unfortunately, these techniques are really time consuming and need significant computing power due to the iterative calculation of the derivative function based on the hessian calculations, which ensure to find the global minimum of the cost function. Considering these conditions, we propose here to overcome this lake of power calculation with the use of a separate computing system, well known by the name of cloud computing.

## 2. Material and methods

Our aim is to estimate vineyard leaf roughness. In this purpose, we use texture analysis. The main goal of texture analysis is to formalize the texture feature by mathematics parameters. In order to test our texture classification protocol, the experiments include images (Figure 2) acquired with a SEM microscope and representing different kinds of leaf surfaces coming from seven plant leaf species.

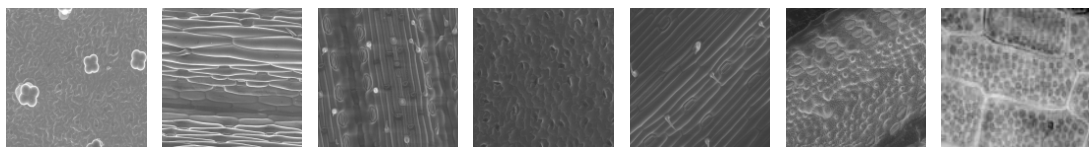


Figure 2: Leaf texture images; from left to right: tomato, ray grass, mature wheat, pea, young wheat, horsetail, vineyard.

For each class of leaf textures, 150 to 200 images have been acquired. Our dataset is made up of 1234 images. Each image consists of a 100  $\mu\text{m}$  scale image, with a resolution of  $512 \times 512$  pixels adapting the scale to our biological application.

### 2.1. Texture feature extraction

In order to extract the texture features, a variety of methods have already been proposed in the literature and tested in practice. In this context, (Jain et al. 1993) distinguish four families of texture feature extraction: statistical, geometrical, model based and signal processing. A successful texture classification or segmentation requires an efficient feature

extraction methodology. However, the major difficulty is that textures in the real world are often not uniform, due to changes in orientation, scale, illumination conditions or others visual appearances. In this context, due to the high variability of orientation, scale and illumination conditions for the different textures, we prefer to test our protocol via the computation of robust invariant such as Generalized Fourier Descriptors (GFD) and Zernike moments (ZM).

### 2.1.1. Generalized Fourier Descriptors

The GFD are defined as follows. Let  $f$  be a square summable function on the plane. The Fourier transform is then:

$$\hat{f}(\xi) = \int_{\mathbb{R}^2} f(x) \exp(-j\xi x) dx. \quad (1)$$

If  $(\lambda, \theta)$  are polar coordinates of point  $\xi$ , we shall denote again  $\hat{f}(\lambda, \theta)$  the Fourier transform of  $f$  at point  $(\lambda, \theta)$ . Gauthier *et al.* (Gauthier et al. 1991) defined the mapping  $D_f$  from  $\mathbb{R}_+$  into  $\mathbb{R}_+$  by

$$D_f(\lambda) = \int_0^{2\pi} |\hat{f}(\lambda, \theta)|^2 d\theta. \quad (2)$$

So,  $D_f$  is the GFD feature vector which describes each texture image as in Figure 3.

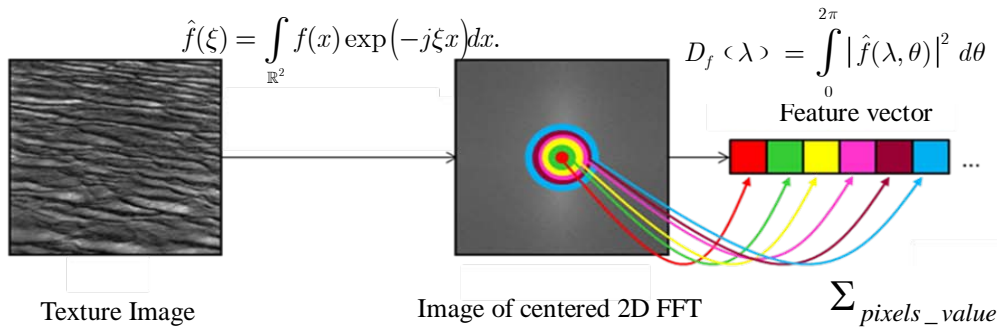


Figure 3: Scheme of GFD texture vectors

This vector will be used as an input of the supervised classification method and be reduced by DR methods. Motion descriptors, calculated according to equation (2), have several properties useful for object recognition: they are translation, rotation and reflection-invariant (Gauthier et al. 1991; Smach et al. 2007).

### 2.1.2. Zernike moments

The kernel of ZM is the set of orthogonal Zernike polynomials defined over the polar coordinate space inside a unit circle. The two dimensional ZM of an image intensity function  $f(r, \theta)$  are defined as:

$$z_{pq} = \frac{p+1}{\pi} \int_0^1 \int_{-\pi}^{\pi} V_{pq}(r, \theta) r dr d\theta, \quad |r| \leq 1 \quad (3)$$

Where the Zernike polynomials are defined as:

$$V_{pq}(r, \theta) = R_{pq}(r) e^{-jq\theta} \quad (4)$$

$$R_{pq}(r) = \sum_{s=0}^{(p-|q|)/2} (-1)^s \frac{(p-s)!}{s! \left(\frac{p-2s+|q|}{2}\right)! \left(\frac{p-2s-|q|}{2}\right)!} r^{p-2s} \quad (5)$$

ZM are rotation-invariant: the image rotation in spatial domain simply implies a phase shift to the ZM. Mukundan et al. (1998), and Khotanzad et al. (1990), have shown that translation-invariance of ZM can be achieved using image normalization method. Chong et al. (2003) present a mathematical framework for the derivation of translation invariants of radial moments defined in polar form.

## 2.2. Dimensionality reduction

The GFD and ZM provide features with potential in pattern recognition. However, it represents very high dimensional datasets which are difficult to handle with redundant information. Unfortunately, in a classification context, high dimensional data are often redundant, strongly correlated and suffer from the problem of the “Hughes Phenomenon” (Hughes 1968). This results in inaccurate classification, as illustrated in the figure 4.

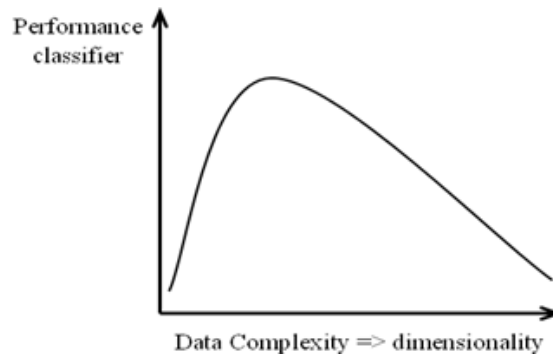


Figure 4: Illustration of Hughes phenomenon

Moreover, data are also large, and the computational cost of elaborates data processing task may be prohibitive. Thus, to improve the classification performance it is well interesting to use DR techniques in order to transform high-dimensional data into a meaningful representation of reduced dimensionality.

### 2.2.1. Estimating intrinsic dimensionality

Let  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)^T$  be an  $n \times m$  data matrix. The number  $n$  represents the number of image examples contained in our texture dataset. Our texture dataset is made up of 1034 agronomical images and 200 vineyard leaf images whose size is 512x512 pixels,  $m$  is the dimension of vector  $\mathbf{x}_i$ , corresponding to the concatenation of the 255 values for GFD and the 25 first ZM. Ideally, the reduced representation has a dimensionality that corresponds to the intrinsic dimensionality of the data. One of our working hypotheses is that, though data points (all texture images) are points in  $\mathbb{R}^m$ , there exists a  $p$ -dimensional manifold  $\mathcal{M} = (\mathbf{y}_1, \dots, \mathbf{y}_n)^T$  that can satisfyingly approximate the space spanned by the data points. The meaning of “satisfyingly” depends on the dimensionality reduction technique used. The so-called intrinsic dimension (ID) of  $\mathbf{X}$  in  $\mathbb{R}^m$  is the lowest possible value of  $p$  ( $p < m$ ) for which the approximation of  $\mathbf{X}$  by  $\mathcal{M}$  is reasonable. In other words, the ID is defined as the number of variables which are sufficient to represent the signal. In order to determine the ID of our dataset, we used a geometric approach that estimates the equivalent notion of fractal dimension (Camastra et al. 2002). Using this method, we estimated and fixed the intrinsic dimensionality of our dataset as being  $p=8$ .

DR methods can be classified according to three characteristics:

- **Linearity:** This describes the type of transformation applied to the data matrix, mapping it from  $\mathbb{R}^m$  to  $\mathbb{R}^p$ .
- **Scale analysis (Local or global):** This reflects the kind of properties the transformation does preserve.

- **Metric (Euclidean or geodesic):** This defines the distance function used to estimate whether two data points are close to each other in  $\mathbb{R}^m$ , and should consequently remain close in  $\mathbb{R}^p$ , after the DR transformation.

It already exists a dozen of linear and non-linear DR methods (Lee et al. 2007). Unfortunately, it's impossible at this time of our work to develop all these methods. We only focus on non linear approaches which present a bad trade-off between really good quality of results and high computational cost, that makes these approaches little workable. Among these methods, some require the minimization of a second order cost function so far impossible to achieve due to the lack of computing power and prohibitive time consuming of the second order gradient resolution algorithms. In order to overcome this problem, we propose here one solution based on cloud computing which allows sharing of computing power through internet.

In order to complete the comparison between the different non linear methods, we use as controls the vector with original features and the reduced vector with the well-known linear approach: PCA (Jolliffe 1986).

### 2.2.2. Linear methods

#### Principal Components Analysis (PCA)

It is the best known DR method. PCA finds a linear transformation for keeping the subspace that has largest variance. It can be shown that reconstruction error  $J_{PCA}$  is minimized for the  $u_i$  being the eigenvectors of the covariance matrix of  $\mathbf{X}$ . It is interesting to note that PCA is close to the classical Multidimensional Scaling (MDS) introduced by Shepard (Shepard 1962) and Kruskal (Kruskal 1964) when Euclidean distance is used as it demonstrates in (Fodor 2002). This relation between MDS and PCA is important because MDS corresponds to the basic method of other nonlinear DR methods such as ISOMAP (Tenenbaum et al. 2000). PCA is linear, global and Euclidean technique.

### 2.2.3. Non-linear methods: global approaches

#### Sammon's mapping. (Sammon)

Sammon's mapping is a DR method that tries to preserve the neighbourhood topology of the set of data by preserving distances between points (Sammon 1969). To evaluate the topology preservation, we use the following stress function minimized by a gradient descent:

$$J_{sam} = \frac{1}{\sum_{i,j=1}^n d_{i,j}^m} \left( \sum_{i,j=1}^n \frac{(d_{i,j}^m - d_{i,j}^p)^2}{d_{i,j}^m} \right) \quad (6)$$

Where  $d_{i,j}^m$  and  $d_{i,j}^p$  are the distances between points  $i^{th}$  and  $j^{th}$  points, in  $\mathbb{R}^m$  and  $\mathbb{R}^p$ . This function allows adapting the distances in the projection space compared to the initial space. Sammon's mapping is a nonlinear, global, and Euclidean method.

#### Isometric feature mapping (Isomap)

Isomap (Tenenbaum et al. 2000) estimates the geodesic distance along the manifold using the shortest path in the nearest neighbours' graph. It then looks for a low-dimensional representation that approximates those geodesic distances in the least square sense. The three steps are: (1) Build  $D_m(X)$ , the all-pairs distance matrix. (2) Build a graph from  $X$  ( $k$  nearest neighbours). For a given point  $x_i$  in  $\mathbb{R}^m$ , a neighbour is either one of the  $K$  nearest data points from  $x_i$  or one for which  $d_{ij}^m < \epsilon$ . Build the all-pairs geodesic distance matrix  $\Delta_m(X)$ , using Dijkstra's shortest path algorithm. (3) Use classical MDS to find the

transformation from  $\mathbb{R}^m$  to  $\mathbb{R}^p$  that minimizes

$$J_{ISOMAP}(X, p) = \sum_{i,j}^n (\delta_{ij}^m - \delta_{ij}^p)^2 \quad (7)$$

Isomap is nonlinear, global and geodesic.

#### 2.2.4. Non-linear methods: local approaches

##### Curvilinear Components Analysis (CCA) and Curvilinear Distance Analysis (CDA)

CCA is an evolution of the Multidimensional Scaling (MDS) and Sammon's mapping algorithms (Demartines et al. 1997). Instead of the optimization of a reconstruction error, CCA aims at preserving the distance matrix while projecting data onto a lower dimensional manifold. Let  $D_m(X)$  be the  $n^2 \times n^2$  matrix of distances between pairs of points in  $\mathbf{X}$ :

$$D_m(X) = (d_{ij}^m), \text{ where } d_{ij}^m = \|\mathbf{x}_i - \mathbf{x}_j\| \quad (8)$$

After DR transformation to  $\mathbb{R}^p$ , we also have:

$$D_p(X) = (d_{ij}^p), \text{ where } d_{ij}^p = \|\mathbf{y}_i - \mathbf{y}_j\| \quad (9)$$

CCA tries to find the best suitable transformation, minimizing:

$$J_{CCA}(X, p) = \sum_{i,j=1}^n (d_{ij}^m - d_{ij}^p)^2 F(d_{ij}^p), \quad (10)$$

Where  $F$  is a decreasing, positive weighting function, giving more importance to the preservation of small distances. CCA is nonlinear, local and Euclidean.

CDA is a refinement of CCA (Lee et al. 2007), minimizing:

$$J_{CDA}(X, p) = \sum_{i,j=1}^n (\delta_{ij}^m - d_{ij}^p)^2 F(d_{ij}^p), \quad (11)$$

Where  $\delta_{ij}^m$  measures the geodesic distance between  $\mathbf{x}_i$  and  $\mathbf{x}_j$ , as in Isomap. CDA is nonlinear, local and geodesic.

### 2.3. Classification

Classification is a central problem in pattern recognition (Duda et al. 2001) and many approaches to solve it have been proposed such as the connectionist approach (Bishop 1995) or metrics based methods, k-nearest neighbours (k-nn) and kernel-based methods such as support vector machines (SVM) (Vapnik 1998). In our experiments, the average performances of the dimensionality reduction methods applied to the features have to be evaluated. In this context, we have chosen and evaluated six efficient classification approaches from four families of classification: the boosting (adaboost) family (Schapire 1990) using three weak classifiers, (hyperplan, hyperinterval and hyperrectangle), the hyperrectangle (polytope) method (Miteran et al. 1994), the SVM method (Vapnik 1998; Abe 2005) and the connectionist family with a multilayer perceptron (MLP) (Rumelhart et al. 1986). To validate the classification performance and estimate the average error for each method, we performed 20 iterative experiments with a 10-fold cross validation procedure (Witten et al. 2005).

### 2.4. Cloud computing

Cloud computing is a concept referring to the use of memory and computing power of servers around the world interconnected via Internet (Vaquero et al. 2009). We see here the principle of grid computing. Users access many services online without having to manage infrastructure. Similarly, applications and data are no longer on their local computers, but in the cloud, Internet. The access service is via a standard application most of the time a Web

browser, located on any PC, or Web services. Applications are being transformed into service, users can consume on demand. This is the concept "Software as a Service (SaaS). This software is being deployed on each machine, freeing customers from software administration. In addition, software evolves seamlessly to customers, without impacting their information systems. Finally, service is billed according to use and the amount of computing resources consumed.

This software, a service, is encapsulated on a platform called "Platform as a Service" (PaaS) (Vaquero et al. 2009). This platform facilitates application deployment with lower cost, lower complexity, without having to purchase and manage hardware, software and hosting capabilities. These platforms are based on infrastructure that has also become a service, "Infrastructure as a Service" (IaaS) (Vaquero et al. 2009). These facilities are generally virtualization environments. Clients no longer have to purchase and manage servers and softwares to ensure the data and software storage and to buy and manage the network equipment. We find a stack with layers formed by the software (SaaS) platforms (PaaS) and infrastructure (IaaS). The client uses one or more of these layers to meet these needs.

Cloud comes in three types: private cloud public, cloud and hybrid cloud. Public cloud (or external) describes cloud in traditional way. That is, Internet resources providing a service request via Web applications or Web services. In private cloud (or internal), companies are transforming their data center infrastructures through virtualization technologies and automation. Infrastructure can then provide resources to application as needed. This results in better resource utilization and reduced costs. However, companies must manage and buy servers, network hardware and software. This type of cloud is only a structure transformation using the tools of public clouds. With private cloud, the users are only purchasing a service rather than purchasing and management to do. Finally, a hybrid cloud is composed of both private and clouds both audiences.

We have built a private cloud using 12 identical machines for hardware level. In terms of software, we used GNU Linux PelicanHPC<sup>1</sup> which is based on GNU Debian<sup>2</sup>. The processing on the images was performed using GNU Octave<sup>3</sup>, a freeware, open source and compatible with MatLab<sup>4</sup>.

### 3. Results and discussion

Here, we present the results in terms of classification error, with several combinations of DR and classification methods and the results in terms of time computing, with several configurations of clouds.

#### 3.1. Classification results

The results are exposed in terms of classification error rate (Table 1). For our dataset, regarding to the classification error using the original feature space, the best results are obtained using SVM ( $e=3.82\%$ ). All the other methods give poorer results (from 5.4 % to 35.7%). Their performances are often improved by DR and the optimum of error rate is obtained by combining CDA and SVM ( $e=1.92\%$ ). The Isomap combination with SVM gives similar results.

It appears that the combination of GFD and ZM for texture extraction, CDA for DR and SVM for classification provides sufficient information to characterize vineyard leaf roughness. Moreover, in the group of fast methods of decision, the best result is obtained using Hyperrectangle combined with Isomap ( $e=4.25\%$ ).

---

<sup>1</sup> <http://idea.uab.es/mcreel/ParallelKnoppix/>

<sup>2</sup> <http://www.debian.org/>

<sup>3</sup> <http://www.gnu.org/software/octave/>

<sup>4</sup> <http://www.mathworks.com/>



		Methods	Boosting			Hyperrectangle	SVM	MLP
			Hyperplan	Hyperinterval	Hyperrectangle			
<b>Original features</b>			6.52	5.4	16.87	27.61	3.82	35.7
	<b>linear</b>	PCA	7.64	3.94	8.62	9.66	<b>2.35</b>	11.9
<b>Dimensionality reduction</b>	<b>nonlinear</b>	Sammon	26.9	25.84	10.89	10.1	5.48	13
		Isomap	7.2	5.12	4.25	7.8	<b>2.28</b>	11.2
		CCA	31.07	17.47	5.23	9.98	<b>2.89</b>	16.2
		CDA	34.13	7.6	4.83	9.75	<b>1.92</b>	15.4

Table 1: Classification results on the plants leaf dataset (% error rate)

It appears that the combination between GFD, ZM and CDA provides sufficient information to characterize plant leaf roughness. Particularly, this solution of texture classification allows to separate our seven kinds of plants leaf surface images like vineyard leaves in seven different clusters.

For our specific application of vineyard spraying, these results have to be combined with contact angles measured using a high-speed imaging system based on PTVS technique under development with the FUSAGx University. In our future work, we want to apply our protocol on two different problems: first, in precision viticulture in order to modelling the evolution of vineyard leaves roughness in a context of precision spraying research by analyzing different growth stage of leaves. Second, we will apply this comparison review to multispectral texture images for which the original dimensional space is higher and for which the correlation between spectral bands are often very important.

### 3.1. Cloud computing results

The results are exposed in terms of time computing:

Methods/Number of machines	No cloud	2	6	12
	(1)	(1+1)	(1+5)	(1+11)
PCA	0,46	0,27	0,09	0,05
Isomap	24,55	14,19	4,84	2,51
CCA	361,78	209,12	71,36	36,99
CDA	378,59	218,84	74,67	38,71
<b>Sammon's mapping</b>	894,21	516,88	176,37	91,43

Table 2: Time computing (in s) with several cloud configurations

For our experiment regarding to the cloud experiment results, the best computing time for iterative approach is obtained with Isomap in 2.51s (12 machines) against 24.55s (1 machine). It is important to note that the rating is not changed between the methods with and without cloud. However, even if the gain seems to be short for Isomap methods, we can see that it is the opposite for Sammon's mapping in 91.43s (12 machines) against 894.21s (1 machine). At this time of our experiment, it is interesting to note that it is the slowest methods for which the gain is the more important.

For the faster methods as PCA, we can see that the gain is not profitable compared to the lowest approach. In the case of the fastest approach, it seems it will appear a strong limitation due to pass band width with and within the cloud.

So, the best methods combination of DR/classification coupled with cloud computing seems to be the CDA/SVM combination for iterative approach even if the standard PCA coupled with SVM still offers a good trade-off between computation time and performances.

## 4. Conclusion and perspectives

In this paper, we proposed a comparison of iterative DR methods combined, with several classification methods coupled, with cloud computing process in the context of texture classification of natural images using GFD and ZM. We used the powerful cloud computing which has interesting properties in the context of hard computing algorithms.

The use of cloud computing for image processing is interesting in terms of computation time. Processing times are reduced significantly. However, our cloud is made up of 12 machines is not enough power for the methods we used. Even if this time has greatly decreased, it is still too high, for example for the Sammon's mapping method. It will significantly increase the number of machines in the cloud and/or have recourse to external clouds, such as Cyclone<sup>5</sup>.

The results are acceptable and the proposed method can be used as robust tool for roughness analysis. Nevertheless, the experiments were done on small samples for the agronomic dataset, although the results obtained on our dataset are pertinent. To improve these results, we will increase the number of leaf texture surfaces with different leaf species (vineyard leaves and other crops) at different growth stages to follow the evolution.

From the image processing viewpoint, comparisons are currently done with other texture features such as spatio-frequency and statistical parameters (Gabor filters, co-occurrence matrices ...) and combination of colour-texture analysis that provide high dimensional data by the concatenation of the textural features and the spectral information.

Finally, the combination of CDA and SVM could be used for other agronomic parameter detection such as hydrophobic or hydrophilic surfaces, or monocotyledon/dicotyledon recognition. The modeling of artificial leaf texture for experiments in laboratory to control the hydrophobicity of surface and then adapt the sprays appears also as an interesting and innovative research. For this last work, comparisons with spectral tools will be necessary.

For a same crop, we want to distinguish different growth stages (like vineyard leaves), in order to optimize the input but also to detect earlier the diseases. Indeed, a disease will modify the structure of the leaf so the texture.

Finally, we are currently developing some research around precision viticulture, and the results and methods presented in this paper will help us to model the ratio between the leaf roughness and the droplets behaviour. This will be combined to the use of optical approaches, such as PTVS (Particle Tracking Velocimetry Sizing) to study the behaviour of the droplets on the leaves. This research interests the sprayer manufacturers but also the phyto pharmaceutical firms.

## 5. References

- Abe, S. (2005). *Support Vector Machines for Pattern Classification*. London, Springer-Verlag.
- Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*, Oxford University Press.
- Camastra, F. and A. Vinciarelli (2002). "Estimating the Intrinsic Dimension of Data with a Fractal-Based Method." *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(10): 1404-1407.
- Chong, C.-W., P. Raveendran and R. Mukundan (2003). "Translation invariants of Zernike moments." *Pattern Recognition* **36**: 1765-1773.
- Demartines, P. and J. Héroult (1997). "Curvilinear Component Analysis : A self-organizing neural network for nonlinear mapping of data sets." *IEEE Transactions on neural networks* **8**(1): 148-154.
- Duda, R. O., P. E. Hart and D. G. Stork (2001). *Pattern Classification (2nd Edition)*. New York, Wiley Interscience Publication.
- Fletcher, R. (2000). *Practical Methods of Optimization*, John Wiley & Sons.

---

<sup>5</sup> [http://www.sgi.com/products/hpc\\_cloud/cyclone/](http://www.sgi.com/products/hpc_cloud/cyclone/)

- Fodor, I. K. (2002). A Survey of Dimension Reduction Techniques, Lawrence Livermore National Laboratory technical report.
- Forster, W. A., J. A. Zabkiewicz and M. O. Kimberley (2005). "A Universal Spray Droplet Adhesion Model." *Transactions of the ASAE* **48**(4): 1321-1330.
- Gauthier, J.-P., G. Bornard and M. Silbermann (1991). "Harmonic analysis on motion groups and their homogeneous spaces." *IEEE Transactions on Systems, Man and Cybernetics* **21**(1): 159-172
- Hijazi, B., F. Cointault, F. Yang and M. Paindavoine (2008). High-speed motion estimation of fertilizer granules with Gabor filters, SPIE.
- Hughes, G. F. (1968). "On the mean accuracy of statistical pattern recognizers." *IEEE Transactions on Information Theory* **14**: 55-63.
- Jain, A. K. and M. Tuceryan (1993). Texture analysis. *Handbook of pattern recognition and computer vision by C.H.Chen*: 235 - 276.
- Jolliffe, I. T. (1986). *Principal Component Analysis*. New York, Springer Verlag.
- Journaux, L., I. Foucherot and P. Gouton (2006). "Reduction of the number of spectral bands in Landsat images : a comparison of linear and nonlinear methods." *Optical Engineering* **45**(6): 067002
- Khotanzad, A. and H. H. Yaw (1990). "Invariant image recognition by Zernike moments." *IEEE Transactions on Pattern Analysis and Machine Intelligence* **12**(5): 489-497.
- Kruskal, J. B. (1964). "Non-metric multidimensional scaling: a numerical method." *Psychometrika* **29**: 115-129.
- Lee, J. A. and M. Verleysen (2007). *Nonlinear Dimensionality Reduction*. London, Springer Verlag.
- Miteran, J., P. Gorria and M. Robert (1994). "Geometric classification by stress polytopes. Performances and integrations." *Traitement du signal* **11**: 393-407.
- Mukundan, R. and K. R. Ramakarishnan (1998). Moment Functions. *Image Analysis-Theory and Applications*. World and B. Scientific. Singapore.
- Robert, P. C. (1999). Precision Agriculture: Status and Research Needs. *Proceedings of the 2nd European Conference on Precision Agriculture*, Odense, Denmark, SCI, London, UK.
- Rumelhart, D. E., J. L. McClelland(1986). *Parallel Distributed Processing*. Cambridge, Mass, MIT Press.
- Sammon, J. W. (1969). "A nonlinear mapping for data analysis." *IEEE Transactions on Computers* **C-18**: 401-409.
- Schapire, R. E. (1990). "The strenght of weak learnability." *Machine Learning* **5**: 197-227.
- Shepard, R. N. (1962). "The analysis of proximities: Multidimensional scaling with an unknown distance function. Part 1." *Psychometrika* **27**: 125-140.
- Smach, F., C. Lemaître, J.-P. Gauthier, J. Miteran and M. Atri (2007). "Generalized Fourier Descriptors with Applications to Objects Recognition in SVM Context " *Journal of Mathematical Imaging and Vision* **30**: 43-71.
- Tenenbaum, J. B., Vin de Silva and J. C. Langford (2000). "A Global Geometric Framework for Nonlinear Dimensionality Reduction." *Science* **290**: 2319-2323.
- Vapnik, V. (1998). *Statistical learning theory*. New York, Wiley Interscience Publication.
- Vaquero, L. M., L. Rodero-Merino, J. Caceres and M. Lindner (2009). "A break in the clouds: towards a cloud definition." *SIGCOMM Comput. Commun. Rev.* **39**(1): 50-55.
- Villette, S., F. Cointault, E. Piron, B. Chopinet and M. Paindavoine (2008). "Simple imaging system to measure velocity and improve the quality of fertilizer spreading in agriculture." *Journal of Electronic imaging* **17**(3): 1109-1119.
- Witten, I. H. and F. Eibe (2005). *Data Mining: Practical Machine Learning Tools and Techniques (Second Edition)*, Morgan Kaufmann.
- Yun, Z. and H. Yong (2006). Crop/weed discrimination using near-infrared reflectance spectroscopy (NIRS), SPIE.