# Tree Species Classification Based on 3D Bark Texture Analysis

Ahlem Othmani[1,2], Alexandre Piboule[2], Oscar Dalmau[3], Nicolas Lomenie[4], Said Mokrani[1], and Lew Fock Chong Lew Yan Voon[1]

[1] Laboratory LE2I - UMR CNRS 6306
12 rue de la fonderie, 71200 Le Creusot, France
{ahlem.othmani,lew.lew-yan-voon}@u-bourgogne.fr, mokranicampus@yahoo.fr
[2] Office National des Forêts, Pôle R&D de Nancy
11 rue de l'Ile de Corse, 54000 Nancy, France
alexandre.piboule@onf.fr
[3] Centro de Investigacion en Matematicas A.C
Guanajuato GTO 36000, Mexico
dalmau@cimat.mx
[4] Laboratory LIPADE - EA 2517, Université Paris Descartes
45 rue des Saints-Pères, 75006 Paris, France
nicolas.lomenie@mi.parisdescartes.fr

**Abstract.** Terrestrial Laser Scanning (TLS) technique is today widely used in ground plots to acquire 3D point clouds from which forest inventory attributes are calculated. In the case of mixed plantings where the 3D point clouds contain data from several different tree species, it is important to be able to automatically recognize the tree species in order to analyze the data of each of the species separately. Although automatic tree species recognition from TLS data is an important problem, it has received very little attention from the scientific community. In this paper we propose a method for classifying five different tree species using TLS data. Our method is based on the analysis of the 3D geometric texture of the bark in order to compute roughness measures and shape characteristics that are fed as input to a Random Forest classifier to classify the tree species. The method has been evaluated on a test set composed of 265 samples (53 samples of each of the 5 species) and the results obtained are very encouraging.

**Keywords:** Tree species classification, 3D pattern recognition, 3D bark texture analysis, forest inventory.

## 1 Introduction

TLS is today a well established technique for the acquisition of precise and reliable 3D point clouds from which forest inventory attributes can be calculated at the single tree level [1] [2]. Numerous work on the calculation of the Diameter at Breast Height (DBH), the height of the tree, the volume of wood and so on can be found in the literature. However, to the best of our knowledge not much

R. Klette, M. Rivera, and S. Satoh (Eds.): PSIVT 2013, LNCS 8333, pp. 279–289, 2014.

has been done in the field of tree species recognition at the single tree level although it is a very important issue if one would like to analyze the 3D data of each of the species of a mixed planting separately. One can only find some work concerning tree species recognition using a combination of TLS data and hyperspectral [3] or panoramic images [4][5]. The aim of our work is to recognize tree species based on TLS data only for two major reasons. The first one is to save acquisition time and/or to avoid the use of multiple acquisition systems for capturing different types of datasets. The second reason is to avoid the need for data coregistration and/or fusion techniques in order to simplify data processing.

Using TLS data only, we can only analyze 3D geometrical shape features such as the shape of the leaves, the general shape of the crown and the variations in geometry across the surface of the bark known as the 3D geometric texture of the bark in order to recognize the tree species. In our case, we have to exclude leaf shape analysis because our forest inventory data are mostly acquired during winter when the trees are leafless. The reason is to reduce occlusions due to leaves for more accurate wood volume calculation and also to do the measurement outside the growing period of the trees. The general shape of the canopy is a good characteristic feature of the species of a tree for isolated trees. However, in a forest planting, the management type and the density have a big impact on the canopy. They render it polymorphic so that it is difficult to use the shape of the canopy as a discriminating criterion for tree species recognition. Finally, the bark is probably the most discriminating feature of the species even if it is subject to changes during the tree's life because of age, injuries and modified growth pattern due to environmental disturbances.

Fig. 1 illustrates the 3D geometric texture characteristics of the bark of the five most important tree species that we have to recognize. One can notice that each of the five tree species has a distinguishable 3D geometric bark texture characteristic. The beech has a relatively smooth surface, the spruce is less smooth compared to the beech and it has circular scars, the pine and the oak are rough with vertical strips but the growth pattern is different, and the hornbeam is smooth with an undulating texture.


**Fig. 1.** 3D point clouds of the five tree species

We thus propose a method that analyzes the 3D geometric texture of the bark in order to classify and recognize the tree species. For the analysis a 30 cm long segment of the tree trunk at about 1.3 m from the ground (breast height) called a patch is considered. 30 cm is a good trade-off between a small patch

for rapid processing times and a long enough segment that contains sufficient texture patterns for recognition.

Our method consists of several steps. Firstly, a 3D deviation map is computed from the 3D point cloud of the 30cm long segment of the tree trunk at breast height. The first step is described in details in section 2. Secondly, the 3D deviation map is transformed into a 2D deviation map or height map that is next segmented in order to reveal characteristic shape features of the tree species. The second step is presented in section 3. Finally, classification features are computed from both the 2D deviation map and the segmented 2D deviation map and fed as input to a Random Forest classifier for tree species classification. The classification features and the selection of the most pertinent features are presented in section 4. In section 5, we describe the test set used to evaluate our method and discuss about the classification results obtained before concluding in section 6.

## 2 3D Deviation Map of the Tree Bark

We define the 3D geometric texture of a 3D surface as the local variations of the original meshed surface denoted by $M_o$ with respect to a smoothed version of the same meshed surface denoted by $M_s$ as illustrated in Fig. 2 for some tree trunk segments.


**Fig. 2.** 3D geometric texture model

To extract the 3D geometric texture of a patch, we first apply a denoising filter, implemented in the RapidForm software, to the 3D point cloud in order to remove ghost points. Next, the 3D point cloud is meshed and smoothed. Finally, the deviation between the original mesh and the smoothed mesh is computed. This yields a 3D deviation map representation of the geometric texture that can be modeled by a dataset $DM3 = \{(x, y, z, d) : x, y, z, d \in \mathbb{R}\}$ where $(x, y, z)$ are the coordinates of the 3D points or vertices of the smoothed mesh and $d$, the Euclidean distance between a point of the smoothed mesh $M_s$ of coordinates $(x, y, z)$, denoted by $\mathbf{v}(x, y, z)$, and its nearest neighbor in the original mesh $M_o$, denoted by $\tilde{\mathbf{v}}(x', y', z')$. $d$ is computed according to equation 1.

$$d(\mathbf{v}, \tilde{\mathbf{v}}) = \sqrt{(x - x')^2 + (y - y')^2 + (z - z')^2} \quad (1)$$

$\tilde{\mathbf{v}}(x', y', z')$ is determined using the efficient Aligned Axis Bounding Box (AABB) tree structure [6]. It is equal to

$$\tilde{\mathbf{v}} = argmin_{\mathbf{p}_i \in M_o} \|\mathbf{p}_i - \mathbf{v}\| \tag{2}$$

The smoothed mesh is computed using Taubin's $\lambda/\mu$ smoothing algorithm [7]. It consists in basically performing the Laplacian smoothing two consecutive times with different scaling factors denoted by $\lambda$ and $\mu$. A first step with $\lambda > 0$ (shrinking step) and a second step with a negative scaling factor $\mu < -\lambda < 0$ (unshrinking step). Laplacian smoothing consists in iteratively moving each of the vertices of the mesh to a new position that corresponds to the weighted average position of the neighboring vertices. The new position $\mathbf{v'}_i$ of a vertex $i$ is given by Eq. 3:

$$\mathbf{v'}_i = \mathbf{v}_i + \lambda \, \Delta\mathbf{v}_i \tag{3}$$

where $\mathbf{v}_i$ is the current position, $\lambda$ a scalar that controls the diffusion speed and $\Delta\mathbf{v}_i$ the Laplacian operator given by Eq. 4. It is a weighted sum of the difference between the current vertex $\mathbf{v}_i$ and all its neighbors $\mathbf{v}_j$.

$$\Delta\mathbf{v}_i = \sum_{j:\mathbf{v}_j \in i^*} w_{i,j}(\mathbf{v}_j - \mathbf{v}_i) \tag{4}$$

where $i^*$ is the set of all the neighbors of the vertex $\mathbf{v}_i$. Taubin's $\lambda/\mu$ smoothing algorithm is run with equal weights $w_{i,j}$ for each of the neighbors such that $\sum_{j:\mathbf{v}_j \in i^*} w_{i,j} = 1$ and with $\lambda$ and $\mu$ equal to 0.6307 and – 0.6732, respectively (values suggested by Taubin). It is run iteratively until the mesh is sufficiently smoothed.

The smoothness of a 3D surface is usually quantified by the minimum, maximum, mean and Gaussian curvatures of each of the points of the mesh. Consequently, we have studied the evolution of the median of these curvature values as a function of the number of iterations of Taubin's algorithm for several samples of the five species to classify. Our study showed that there is no significant difference in the curvature against number of iterations curve for the four types of curvature values. We have thus decided to consider only the median of the maximum curvatures curve, as shown in Fig. 3, to determine the smoothing stopping criterion.

We have chosen to stop the smoothing process when the slope of the tangent to the curve is less than or equal to -0.01. For this value of the slope we have noticed that the mesh is sufficiently smoothed while preserving the main structures of the trunk.

## 3 2D Deviation Map

The next step of our method is the transformation of the 3D deviation map, a dataset $DM3 = \{(x, y, z, d) : x, y, z, d \in \mathbb{R}\}$, into a 2D deviation map, a dataset $DM2 = \{(X, Y, d) : X, Y, d \in \mathbb{R}\}$. It is a dimensionality reduction problem

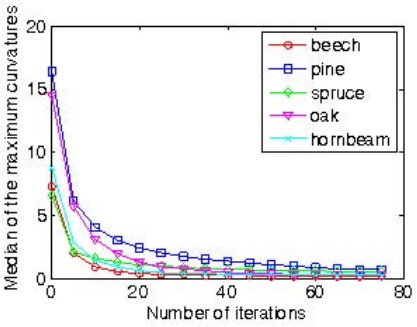



**Fig. 3.** The median of the maximum curvatures against the number of iterations for each of the five tree species

that should preserve the intrinsic geometry of the data. The two classical techniques for dimensionality reduction is Principal Component Analysis (PCA) and Multidimensional Scaling (MDS). However, they are not appropriate for our 3D deviation map because they are linear techniques and our 3D deviation map is a non linear structure.

One can find in the literature several non linear techniques that are more appropriate for our 3D deviation map. Among all these techniques we have chosen the Maximum Variance Unfolding (MVU) dimensionality reduction algorithm proposed by Weinberger et al. [8] because it is fast and it gives good results. An example of the 2D deviation map obtained using MVU is illustrated in Fig. 4.

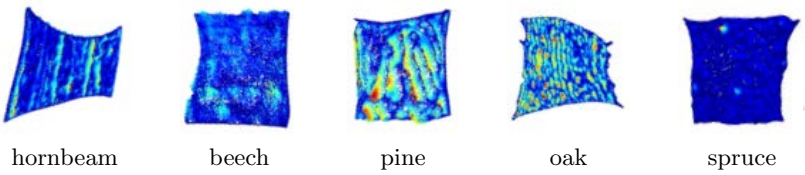



**Fig. 4.** Example of 2D deviation map for each of the five species

The 2D deviation map is a set of points in a 3D space. It is different from the point cloud in the sens that the third dimension represents a distance from a plane surface defined by the other two dimensions. It is like a height map or relief map defined by a set of points. The idea now is to cluster the points that are above a certain distance or "height" from the plane in order to define clusters or regions whose shape will allow us to classify the different species. The result of the clustering is a segmented 2D deviation map. To acheive this, a first thresholding step is done in order to keep the most salient features, e.g. the points with the highest deviation values. The threshold value is empirically defined as the median value of the deviation values. In this way only all the points in the 2D deviation map that have a height value greater than the median value are kept.

Secondly, DBSCAN a density based algorithm for discovering clusters proposed by Ester et al. [9] is used to cluster the points in order to build regions. DBSCAN is chosen because it is a fast and efficient algorithm even for large spatial set of points which is our case. DBSCAN is based on the notion of density of points in an $\epsilon$-neighborhood. It has two required parameters: the neighborhood size ($\epsilon$) and the minimum number of points in the $\epsilon$-neighborhood ($minPts$). For our application $\epsilon = 0.6$ and $minPts = 8$.

An example of the segmented 2D deviation map for each of the five species is illustrated in Fig. 5.

hornbeam beech pine oak spruce

**Fig. 5.** An example of segmented 2D deviation map for each of the five species

## 4 Classification Features

From the 2D deviation map and the segmented 2D deviation map, we can compute about 128 features for classification. These features are, for example :

- The number of clusters per segmented 2D deviation map.
- The roughness features described in section 4.1.
- The principal component analysis features described in section 4.2.
- The shape and intensity features presented in section 4.3. Each cluster is defined by a set of shape and intensity features. We calculate the median, the mean, the standard deviation, the minimum and the maximum of the intensity and the shape features of all the clusters and use them as classification features.

### 4.1 Roughness Features

The 2D deviation map represents the geometric details of the surface of the bark from which roughness measures can be computed. Many types of measures can be found in the literature. The most common ones are statistical values such as the root mean square value, the arithmetical mean and the standard deviation respectively given by Eqs. (5)-(7).

$$s_q = \sqrt{\frac{1}{n}\sum_{i=1}^{n} d(\mathbf{v}_i, \tilde{\mathbf{v}}_i)^2} \tag{5}$$

$$s_a = \frac{1}{n}\sum_{i=1}^{n} |d(\mathbf{v}_i, \tilde{\mathbf{v}}_i)| \tag{6}$$

$$s_D = \sqrt{\left(\frac{1}{n}\sum_{i=1}^{n} d(\mathbf{v}_i, \tilde{\mathbf{v}}_i)^2\right) - \left(\frac{1}{n}\sum_{i=1}^{n} d(\mathbf{v}_i, \tilde{\mathbf{v}}_i)\right)^2} \tag{7}$$

### 4.2 Principal Component Analysis Features

Principal Component Analysis (PCA) is a technique for dimension reduction and feature extraction. It uses linear transformations to map data from a high dimensional space to a low dimensional space. The new variables in the low dimensional space are called principal components. Some of the features that can be calculated for each of the clusters of points of the segmented deviation map using PCA are listed below.

- The percentage of the total variance explained by each principal component.
- The maximum and the median distances between the observations and the center of the data set as well as the ratio between the maximum and the median distances.
- The longest and shortest diameter (length of the major and the minor axis).
- The aspect ratio defined as the major axis length divided by the minor axis length.
- The orientation or the direction of the first and the second principal component.

The minimum, maximum, mean, median and standard deviation values of these PCA features for all the clusters of the segmented 2D deviation map are used as features for classification.

### 4.3 Shape and Intensity Features from Segmented 2D Deviation Map

To characterize the clusters of points in the segmented deviation map, shape and intensity features are calculated for each cluster. The intensity features are the maximum and the median intensity of the points of the cluster. To calculate the shape features, the alpha shape ($\alpha$-shape) algorithm proposed by Edelsbrunner et al. [10] is used to compute the $\infty$-shape that corresponds to the convex hulls of the clusters of points. The 0.6-shape that best represents the real shape has also been computed. From the $\infty$-shape and the 0.6-shape, shape features such as perimeter, area, solidity, roundness, compactness, RFactor, shape, convexity and concavity are computed and used as classification features. The mathematical expressions of some of these features are given by Eqs. (8)-(12). To discriminate between the strips of the pine that are close and narrow and those of the oak that are touching each other, the ratio between the area of the convex hull and the area of the 0.6-shape is used as feature.

$$Solidity = \frac{Area}{ConvexArea} \tag{8}$$

$$Roundness = \frac{4 \times Area}{\pi \times MajorAxisLength^2} \tag{9}$$

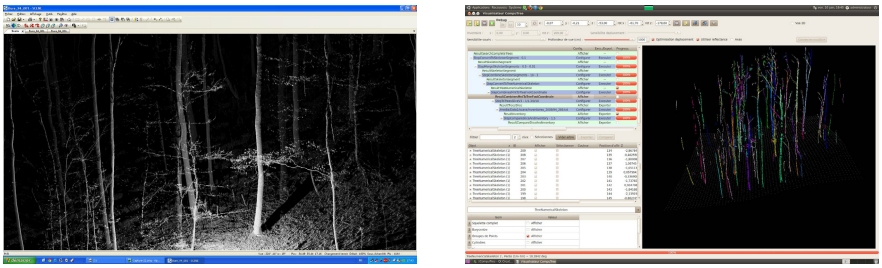
$$Compactness = \frac{\sqrt{\frac{4}{\pi} \times Area}}{MajorAxisLength} \tag{10}$$

$$RFactor = \frac{PerimeterOfConvexHull}{MajorAxisLength \times \pi} \tag{11}$$

$$Shape = \frac{Perimeter^2}{Area} \tag{12}$$

## 5 Experiments and Results

The 3D TLS data used to evaluate our method were captured using either a FARO Photon 120 or a FARO Focus3D scanner with a resolution of about 6mm at 10m. All trees at a distance of about 6m and with a Diameter at Breast Height (DBH) of about 30cm are manually located in the scans. Next, non occluded tree trunk segments of about 30cm long and at a height of about 1.30m from the ground are extracted using a software that we have developed, the "Computree" software [2], in order to constitute the evaluation database.



(a) (b)

**Fig. 6.** (a) 3D point cloud captured by FARO Photon 120 scanner (b) Computree Software screenshot

We used two different datasets D1 and D2 to validate our approach experimentally. The test site of D1 is a state mixed forest in Montiers-sur-Saulx, France. The second test site of D2 is a mixed forest stands of grove and coppice-under-grove forests in Lorraine, France. Both datasets contain the five different species of trees represented in Fig. 1. D1 and D2 are composed of 20 patches per species and 33 patches per species, respectively. Classification is done using the R Language implementation of the Random Forest (RF) classifier proposed by Breiman [11]. The RF classifier is built with recommended values for the number

of decision trees (1000) and the number of features used to split the node in the decision tree growing process denoted by $Mtry$ ($Mtry = \sqrt{D}$ where $D$ is the feature vector size). We have tested 128 features and used RF to select the 30 most pertinent ones to evaluate our method. So, the value of $D$ is 30.

Three tests are done using datasets D1, D2 and a combination of D1 and D2, respectively. For each test, stratified 10-fold cross-validation is performed multiple times with the dataset reshuffled and re-stratified before each round. Also, for each round, a new classifier build with the recommended values of the number of trees and of the $Mtry$ parameter is used. The confusion matrices with average results are reported in Tables (1)-(3) and the accuracy rates for the three tests are summarized in Table 4.

**Table 1.** Confusion matrix for D1 cross-validation (1: hornbeam, 2: oak, 3: spruce, 4: beech, 5: pine)

| | 1 | 2 | 3 | 4 | 5 | Accuracy |
|---|---|---|---|---|---|---|
| 1 | 20 | 0 | 0 | 0 | 0 | 100 % |
| 2 | 0 | 20 | 0 | 0 | 0 | 100 % |
| 3 | 0 | 0.8 | 19.2 | 0 | 0 | 96 % |
| 4 | 1.8 | 0 | 0.2 | 18 | 0 | 90 % |
| 5 | 2.2 | 0 | 1 | 0.2 | 16.6 | 83 % |

**Table 2.** Confusion matrix for D2 cross-validation (1: hornbeam, 2: oak, 3: spruce, 4: beech, 5: pine)

| | 1 | 2 | 3 | 4 | 5 | Accuracy |
|---|---|---|---|---|---|---|
| 1 | 33 | 0 | 0 | 0 | 0 | 100 % |
| 2 | 0.6 | 31.8 | 0 | 0 | 0.6 | 96.3 % |
| 3 | 0 | 0 | 31.6 | 0.6 | 0.8 | 95.7 % |
| 4 | 1.8 | 0 | 0.6 | 32.4 | 0 | 98.1 % |
| 5 | 0.8 | 1 | 0 | 0 | 31.2 | 94.5 % |

For both datasets, each with a different terrain and architecture characteristics, we obtained good classification results. One can note from the confusion table that the accuracy ranges from 83% to 100%. The worst accuracy of 83% is for the pine that is misclassified as any one of the other four species in the three tests. Pine is mainly confused with hornbeam: the two species have strips. We can differentiate two families of tree species from the five tested species: species that have straps or cracks such as the hornbeam, the oak and the pine, and smooth surface species such as the beech and the spruce. We note that our algorithm mixes mainly intra-family species (spruce and beech) but also inter-family species. Future work will focus on finding more pertinent features to discriminate inter and intra family species.

**Table 3.** Confusion matrix for D1 and D2 cross-validation (1: hornbeam, 2: oak, 3: spruce, 4: beech, 5: pine)

| | 1 | 2 | 3 | 4 | 5 | Accuracy |
|---|---|---|---|---|---|---|
| 1 | 53 | 0 | 0 | 0 | 0 | 100 % |
| 2 | 0 | 52.6 | 0 | 0 | 0.4 | 99.3 % |
| 3 | 0 | 0 | 51.6 | 0.4 | 1 | 97.4 % |
| 4 | 0.4 | 0 | 0.4 | 52.2 | 0 | 98.5 % |
| 5 | 1.4 | 0.4 | 0.4 | 0.4 | 50.4 | 95.1 % |

**Table 4.** Accuracy rates for the three tests

| | Min | Average | Max | $\sigma$ |
|---|---|---|---|---|
| Test 1 | 83% | 93.8% | 100% | 6.5% |
| Test 2 | 94.5% | 96.92% | 100% | 1.9% |
| Test 3 | 95.1% | 98.06% | 100% | 1.7% |

We can note that training and testing with dataset D3 (test 3) gives overall better accuracy rate than the other tests. This may be due to the fact that D3 contains more samples. Nevertheless, the result is relatively good enough and we can consider that the classifier performs reasonably well since the datasets D1 and D2 are composed of patches extracted from 3D data acquired from two different forest sites.

## 6 Conclusions

We have proposed a method for classifying five different tree species using TLS data only. The method is based on the analysis of the 3D geometric texture of the bark in order to identify the species class to which pertains the tree under analysis. In our method, the 3D geometric texture is transformed into a 2D deviation map on which roughness measures and shape features are computed and used as features for classification using the Random Forest classifier. Results obtained on a dataset composed of 265 samples with equal number of samples of each species are quite good (83% to 100%). In future work we plan to do more tests with other datasets in order to verify this observation. Moreover, we would also like to study the influence of the distance to scanner and of the DBH on the results. Indeed, in the current dataset the samples are all extracted from trees located at about 6m from the scanner and with a DBH of about 30cm.

**Acknowlegments.** This work is financially supported by the "Conseil Régional de Bourgogne" under contract Nº 2010 9201AAO048S06469 and Nº 2010 9201CPERO007S06470, and the 'Office National des Fôrets".

## References


1. Dassot, M., Constant, T., Fournier, M.: The Use of Terrestrial LiDAR Technology in Forest Science: Application fields, Benefits and Challenges. Annals of Forest Science 6, 959–974 (2011)
2. Othmani, A., Piboule, A., Krebs, M., Stolz, C., Lew Yan Voon, L.F.C.: Towards Automated and Operational Forest Inventories with T-LiDAR. In: SilviLaser - 11th International Conference on LiDAR Applications for Assessing Forest Ecosystems, Hobart, Australia, October 16-19 (2011)
3. Puttonen, E., Suomalainen, J., Hakala, T., Rikknen, E., Kaartinen, H., Kaasalainen, S., Litkey, P.: Tree species classification from fused active hyperspectral reflectance and LIDAR measurements. Forest Ecology and Management 260, 1843–1852 (2010)
4. Haala, N., Reulke, R., Thies, M., Aschoff, T.: Combination of terrestrial Laser Scanning with high resolution panoramic Images for Investigations in Forest Applications and tree species recognition. In: Proceedings of the ISPRS Working Group V/1, IAPRS - XXXIV (PART 5/W16), Dresden, Deutschland, February 19-22. Panoramic Photogrammetry Workshop (2004)
5. Reulke, R., Haala, N.: Tree species recognition with fuzzy texture parameters. In: Klette, R., Žunić, J. (eds.) IWCIA 2004. LNCS, vol. 3322, pp. 607–620. Springer, Heidelberg (2004)
6. Alliez, P., Tayeb, S., Wormser, C.: AABB Tree, CGAL 3.5 edn. (2009)
7. Taubin, G.: Geometric Signal Processing on Polygonal Meshes. State of the Art Report, Eurographics (2000)
8. Weinberger, K.Q., Packer, B.D., Saul, L.K.: Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization. In: Proceedings of the Tenth International Workshop on AI and Statistics (AISTATS 2005), Barbados, WI (2005)
9. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD 1996), Portland, OR, USA, pp. 226–231 (1996)
10. Edelsbrunner, H., Kirkpatrick, D.G., Seidel, R.: On the Shape of a Set of Points in the Plane. IEEE Transactions on Information Theory IT-29(4) (July 1983)
11. Breiman, L.: Random Forests. Machine Learning, 5–32 (October 2001)