Geometric models for plant leaf area estimation from 3D point clouds: a comparative study

Mélinda Boukhana, Joris Ravaglia, Franck Hétroy-Wheeler, Benoit de Solan

Geometric models for plant leaf area estimation from 3D point clouds: a comparative study

Mélinda Boukhana[a], Joris Ravaglia[a], Franck Hétroy-Wheeler[a,*], Benoît de Solan[b]

[a] *University of Strasbourg, ICube Research Institute, 300 bd Sebastien Brant, CS 10413, F-67412 Illkirch cedex, France*
[b] *Arvalis - Institut du Végétal, 228 route de l'Aérodrome, CS 40509, F-84914 Avignon Cedex 9, France*

ARTICLE INFO

Article history:
Accepted 23 August 2022

Keywords: Leaf area, 3D acquisition, 3D reconstruction, phenotyping

ABSTRACT

Measuring leaf area is a critical task in plant biology. Meshing techniques, parametric surface modelling and implicit surface modelling allow estimating plant leaf area from acquired 3D point clouds. However, there is currently no consensus on the best approach because of little comparative evaluation. In this paper, we provide evidence about the performance of each approach, through a comparative study of four meshing, three parametric modelling and one implicit modelling methods. All selected methods are freely available and easy to use. We have also performed a parameter sensitivity analysis for each method in order to optimise its results and fully automate its use. We identified nine criteria affecting the robustness of the studied methods. These criteria are related to either the leaf shape (length/width ratio, curviness, concavity) or the acquisition process (e.g. sampling density, noise, misalignment, holes). We used synthetic data to quantitatively evaluate the robustness of the selected approaches with respect to each criterion. In addition we evaluated the results of these approaches on five tree and crop datasets acquired with laser scanners or photogrammetry. This study allows us to highlight the benefits and drawbacks of each method and evaluate its appropriateness in a given scenario. Our main conclusion is that fitting a Bézier surface is the most robust and accurate approach to estimate plant leaf area in most cases.

1. Introduction

The accurate measurement of geometric characteristics of plants and trees, such as height [1], branch angles [2] or leaf areas [3], are critical in, among others, plant physiology, agronomy and forestry. Such measurements are used in developmental studies [4], phenotyping [1], or inventory [5]. Among these characteristics, the individual leaf area is of particular interest for monitoring the development of leaves over time, and to access physiological key parameters. For example, leaf area over time is key for monitoring plant transpiration [6]. It is also essential in computing the leaf area index (the one-sided leaf area per unit ground surface area) [5], which characterizes the plant canopy and the amount of light that can be intercepted by the plant [7].

Manual measurements on real plants face several limitations. They are time-consuming, labour intensive, prone to errors and sometimes even impossible. Most importantly, they are often destructive and thus do not allow to track changes over time. As a consequence, photogrammetry and laser scanning technologies have been proposed in the last decade to solve this problem [5, 8, 9]. These technologies create a virtual model of the plant under study in the form of a 3D point cloud. Measurements of the plant geometric characteristics are then automatically computed on this point cloud. Three stages are necessary to estimate the area of each leaf. In the first stage, points belonging to a single leaf blade are segmented from the rest of the plant. In the second stage, a continuous surface model is derived from these points. In the final stage, the leaf area is estimated from this continuous model.

In this paper, we focus on the second stage of this pipeline, namely the approximation or interpolation of a 3D point cloud sampled on a leaf blade by a continuous surface. Note that in the following we use the term "leaf" instead of "leaf blade" for simplicity purpose. A leaf is usually composed of a 2-dimensional leaf blade and a 1-dimensional petiole connecting the blade to a stem.

Three main approaches to approximates a leaf point cloud by a surface have been proposed in the literature. The first one directly connects neighbouring points into a triangle mesh [10, 11, 12, 13, 14, 9, 15]. The leaf area is then approximated as the sum of all triangle areas. The second approach uses a parametric surface model to approximate the input points [16, 17, 18, 19, 20, 21]. A mesh sampled on this

*Corresponding author.
e-mail: hetroywheeler@unistra.fr (Franck Hétroy-Wheeler)

model is then used to estimate the leaf area. The third approach constructs an implicit representation of the leaf from the point cloud [22]. As in the previous case, the leaf area is estimated by meshing the resulting implicit surface.

To the best of our knowledge, there is no consensus on whether one approach is better than the other. In this paper, we propose a thorough comparative study of eight methods selected from the three approaches described above. Rather than focusing on very specific methods, which would be hardly available, understandable and usable by plant biologists, we have chosen to select standard methods which are freely available in open-source software or libraries, except two of them for which we release a code. We also release our code to generate the synthetic datasets we used to evaluate the selected methods. Our code is freely available at the following address: `https://gitlab.unistra.fr/plant-leaf-area-estimation-gvc-2022`.

2. Related work

Existing pieces of work have tackled similar problems, but only partially. Loch et al. compared two meshing approaches generating respectively (flat) Euclidean triangles and a smooth surface made of piecewise cubic (non-flat) Clough-Tocher triangles [16]. More recently, [18] and [20] compared three parametric leaf surface models. In these studies, surface models were evaluated with respect to the maximum or root mean square distance between the computed parametric surface and the input point cloud, but no evaluation was given with respect to the leaf area.

The most related work to our study is that of [23]. As in [16], the authors compared a piecewise linear mesh model with a parametric surface, this time for leaf area estimation. The mesh was computed as a 2D triangulation of the data points projected onto the best fitting plane of the point cloud, and then lifted back to their original locations. The parametric model chosen was a B-spline surface, which was trimmed according to the boundary of an alpha-shape [24]. Both models were evaluated on point clouds acquired with a LiDAR from a flat gauge block. Only one parameter was tested: the thinness of the point cloud, which could be related to the noise level. Though Dupuis et al. only compared these models on simple, flat surfaces, they showed that the B-spline approach measures leaf areas with much greater accuracy than the direct triangulation approach. In our study, we have included the same meshing method, but a more advanced B-spline technique [25] is tested.

It is also worth mentioning that a comparative analysis across five geometric models has also been proposed for tree branches [26]. These models were evaluated on synthetic data with varying shape, length, noise level, sampling resolution and occlusion level. They were also tested on data acquired from a real tree with a terrestrial LiDAR. Though our methodology is similar to the one proposed by Åkerblom and his colleagues, our problem is more complex because leaves have greater geometrical variability than branches and are more sensitive to acquisition artefacts. In a broader context, Berger et al. [27] give a comprehensive survey of mesh reconstruction algorithms from 3D point clouds. However, their study was mostly focused on meshing watertight surfaces, while the leaves we are interested in are surfaces with boundaries.

Table 1. Overview of the selected criteria, their corresponding parameters, number of different values used for each parameter and their range used to generate the synthetic data. The lengths are indicated with respect to the distance between the base and the apex of the leaf: 0.12 means 12% of this distance. The default value is indicated in bold. By default the sampling density uses $5,000$ points.

Criterion	Parameter(s)	Nb Values	Range
Length/width ratio	**R**	11	**1** – 12
Curviness	**Cu**	19	**0** – 360°
Concavity	**Co**	6	**0** – 5 iterations
Sampling pattern	**S.p**	2	**Random** or Regular
	S.n	11	**0** – 0.008 m
Sampling density	**D**	16	50 – 75,000 pts/m^2
Nonuniform sampling	**NU**	11	**1** – 10
Acquisition noise	**N**	11	**0** – 0.15
Misalignment	**M.d**	8	**0** – 0.24 m
	M.l	6	**0** – 0.33 m
Holes	**H**	11	**0** – 0.6 m

3. Materials and methods

3.1. Leaf point clouds

Several criteria can affect the robustness and the accuracy of surface reconstruction approaches. We have identified nine criteria (Table 1) and we have built synthetic data to study their individual influence on area estimation results. In order to evaluate the eight approaches we have selected, we also used point clouds acquired from real plants associated with manual measurements of leaf area.

3.1.1. Criteria and synthetic data

Synthetic leaf model. Geometrically speaking, a plant leaf is a surface with one boundary because it is usually 2-dimensional. Several synthetic leaf models have been proposed in the literature, for example using ellipses [28] or polynomials [29]. We propose a simpler model which still covers a broad range of species, as we explain below.

We first approximate synthetic leaves by a flat hexagon. Two of its six vertices, opposite to each other, correspond to the base and the apex of the leaf. The distances between the base and the apex and between the opposite left and right sides are equal, and the hexagon area is set to 1. Let us now describe the nine criteria that we have used to generate synthetic data. The first three are related to the shape of the leaf and define our synthetic leaf model. The others simulate defects in the point clouds due to the acquisition process. A single parameter is usually associated with each criterion.

Most leaves are not isotropic and exhibit one principal direction, which corresponds to the base-to-apex axis (the midrib). We thus define the **length/width ratio (R)** of the leaf as the first of our nine criteria. Similarly, leaves are curved rather than planar objects. Hence we identified **curviness (Cu)** as a second criterion. We define curviness as the angle between the tangent planes at the base and at the apex of the leaf. We create a curvy leaf by smoothly bending it along its base-apex axis only. Note

that curviness is not limited to 360 °; a leaf with **Cu** > 360 takes the shape of a Swiss roll. Finally, the leaf boundary is not always convex and can contain so-called teeth; our third criterion is then **concavity (Co)**. Its value is defined recursively: it is 0 for a convex shape, 1 when a zigzag is created by breaking a straight boundary edge into three smaller straight edges, 2 when a zigzag is itself divided into three zigzags (thus defining nine edges), etc. We generated a set of synthetic leaves by varying each criterion independently (see Fig. 2-(a-c)). All generated shapes share the same unit area.

The perimeter, area and convex-hull perimeter of a leaf can easily be expressed with respect to the **length/width ratio** and the **concavity** criteria, as detailed in Appendix A. This enables us to give exact formulas about the aspect ratio, circularity and solidity indicators, which are used in [30] to classify plant taxonomic groups based on their leaf shapes. The definitions given in [30] for the three indicators used are:

- *Aspect Ratio*: $AR = \mathbf{R}$
- *Circularity*: $Ci = 4\pi(\frac{\mathcal{A}}{\mathcal{P}^2})$
- *Solidity*: $S = \frac{\mathcal{A}}{\mathcal{CH}}$

with $\mathcal{P}$ the perimeter of the leaf, $\mathcal{A}$ its area and $\mathcal{CH}$ the area of its convex hull. Based on the study detailed in Appendix A, we can easily express these indicators with respect to **R** and **Co**. For each value of **Co** between 0 and 5, we have plotted Ci vs. $1/AR$, S^8 vs. $1/AR$ and Ci vs. S^8 on the pictures given in [30], see Fig. 1 (note that solidity S is usually so close to 1 that it should be used to the power 8). It shows that our synthetic leaf model crosses the morphospace of the following taxonomic groups: **Solanaceae**, **Brassicaceae**, **Alstroemeria**, **Passiflora** and **Viburnum**. It is however too simple to cover other groups such as **Cotton**, **Grape** and **Ivy**. This is mainly because our definition of **concavity** does not allow for zigzags of various sizes along the leaf edge.

Impact of the acquisition process. Laser scanning and photogrammetry techniques sample a real plant as a 3D point cloud. While laser scanners follow a regular pattern to sample the 3D space, photogrammetry relies on the detection of feature points in the images, leading to point clouds with irregular sampling. We thus integrated a criterion related to the sampling technique: the **sampling pattern (S.p)** is defined as either regular or random in order to mimic either laser scanners or photogrammetry. The regularity of laser scanner sampling is also affected by the leaf orientation, hence we included a random noise affecting the location of the points on the leaf surface in the regular sampling process. This noise is in the leaf plane and is the second parameter **S.n** related to this criterion. In practice, its value corresponds to the maximum distance between the actual and the expected locations of a point. Laser scanning and photogrammetry can generate 3D point clouds of varying precision depending on the distance to the scanner or the number and quality of pictures used in the 3D reconstruction. Hence, we included an additional criterion related to the sampling quality: the **sampling density (D)** which is the number of sampled points per surface unit.

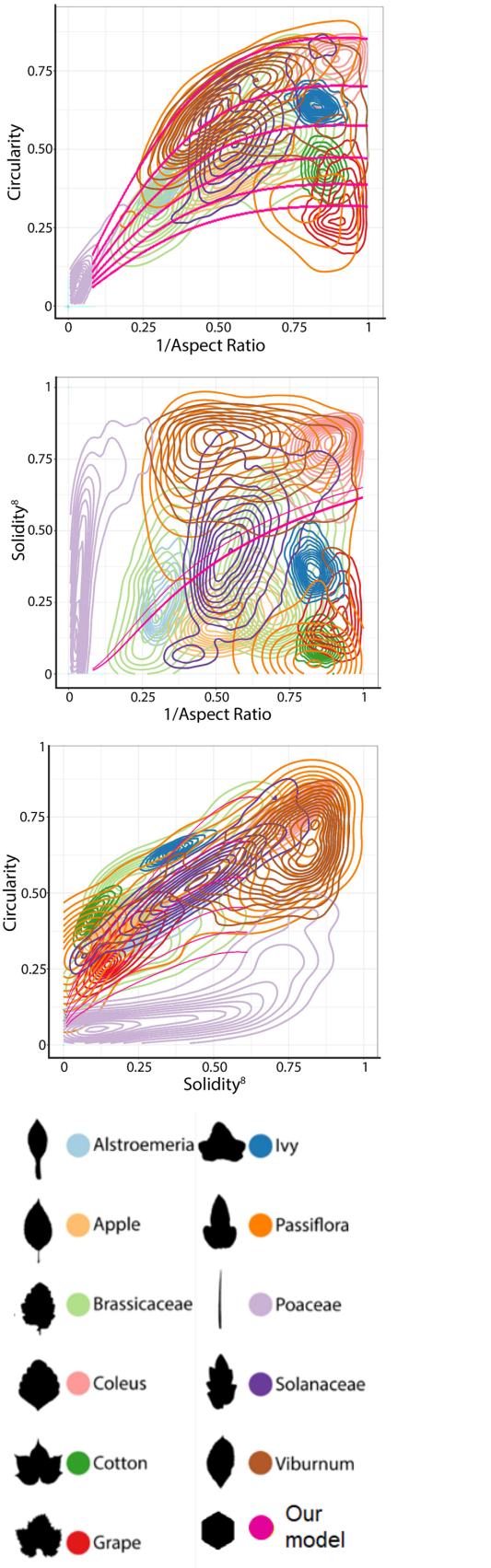


Fig. 1. Morphospaces spanned by traditional leaf shape descriptors for different taxonomic groups, and the position of our synthetic model (pink lines). Images from [30].

[27] has listed five potential artefacts in such point clouds: nonuniform sampling, noisy data, outliers, misaligned scans, and missing data. We do not consider outliers since, in our context, most authors suggest to filter the point cloud as a preprocessing step. Such filtering is however investigated, see Section 3.3.2. We consider the four remaining listed criteria. Nonuniform sampling can result from the registration of several scans from different viewpoints. It is modelled in our study by sampling the leaf with two different densities. The bottom half of the leaf (next to its base) is sampled using the density **D**, while the top half (next to the leaf apex) is sampled using the density multiplied by a user-defined **nonuniform sampling ratio (NU)**. Additionally, point cloud acquisition from a real plant is not perfectly accurate, leading to erroneous point coordinates. This can be due to imperfections in the sensor's optical and mechanical components, or to the environmental conditions, such as the ambient light or the leaf's texture. In this study, we name **acquisition noise (N)** the uncertainty in point coordinate measurement. We model it by a Gaussian noise affecting each point in the normal direction to the leaf surface. We express it as a percentage of the leaf size to make it independent of this size. The registration of several scans is also often imperfect and leads to **misalignment (M)**. We model it by separating the leaf into two overlapping parts, one starting from the base and the other from the apex. One part is then translated in its normal direction and points are sampled on both parts. The (vertical) distance between the two parts **M.d** and the overlap length **M.l** are the two parameters related to this criterion. Finally, our last criterion is the presence of holes inside the shape. They represent missing data due to occlusions during the acquisition process, or when the leaf has been partially eaten by an insect, for instance. To evaluate the influence of this criterion on the robustness of each geometric model, we have generated hexagonal leaves with square **holes** of varying **edge length (H)**.

Generated dataset. We generated a synthetic dataset by varying each criterion independently as summarized in Table 1. The amount of noise is defined with respect to the leaf length. For example, $\mathbf{N} = 0.15$ corresponds to an uncertainty of 1.5 cm for a 10 cm-long leaf. The default density value $\mathbf{D} = 5,000$ points is chosen so that the leaf boundary is densely sampled and to avoid sampling inhomogeneity to be interpreted as holes. Some examples of synthetic data are shown in Fig. 2. We have chosen to vary each criterion independently to better study their individual effects on leaf area estimation methods, although our code is able to generate synthetic leaves with as many different artefacts as desired. In order to take randomness of the sampling and/or the noise into account, we created 10 point clouds for each parameter value, leading us to generate 1240 synthetic point clouds in total. Note that each point cloud is a different approximation of an hexagon which area remains equal to 1.

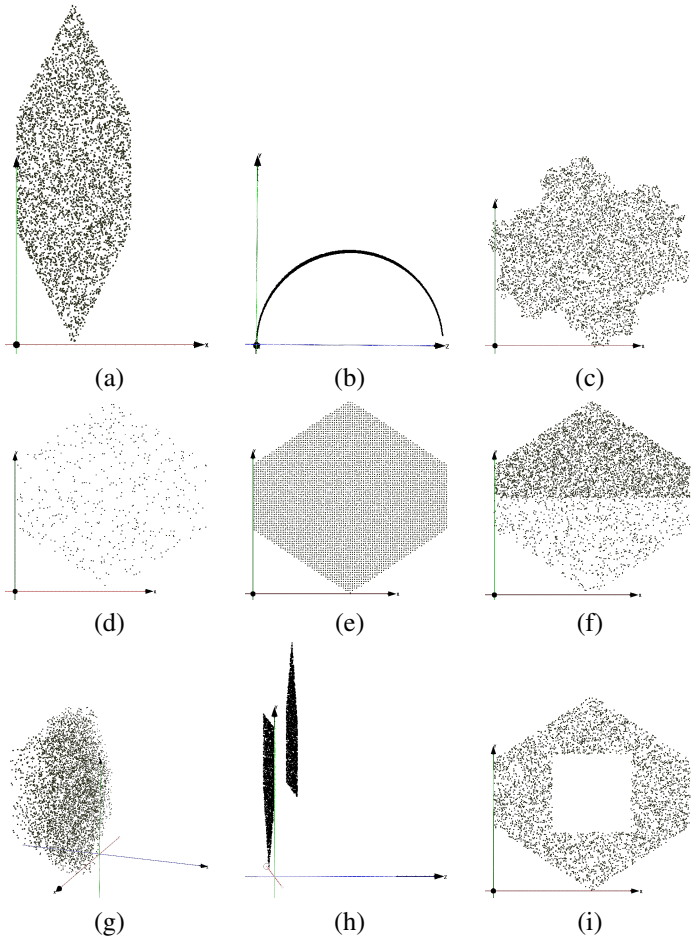


Fig. 2. Examples of synthetic data generated from a flat hexagon: (a) shape with an increased length/width ratio, (b) shape with an increased curviness, (c) shape with an increased concavity, (d) shape sparsely sampled, (e) shape regularly sampled, (f) shape with a nonuniform sampling, (g) shape with a high acquisition noise, (h) shape with misalignment, (i) shape with a hole.

3.1.2. Real data

In order to evaluate the geometric models in real-life scenarios, we have scanned a set of selected plants, collected and measured their leaves to obtain ground truth area measurements. Species were chosen in order to test leaves with different

Table 2. Leaf area, number of points and point density ranges for each of our five datasets of real plants scans.

Dataset	Number of leaves	Leaf areas (cm^2)	Points per leaf	Point density (pts/cm^2)
Chestnut	19	4 – 52	964 – 18108	20 – 50
Oak	10	9 – 110	2950 – 45673	23 – 45
Birch	7	8 – 27	3301 – 11745	31 – 35
Maize (LiDAR)	12	36 – 202	3541 – 39314	47 – 234
Maize (photo)	10	36 – 202	3334 – 20579	7 – 42

length/width ratios, curvatures and concavity levels. Some resulting point clouds are shown in Fig. 3. Table 2 provides an overview of the selected dataset.

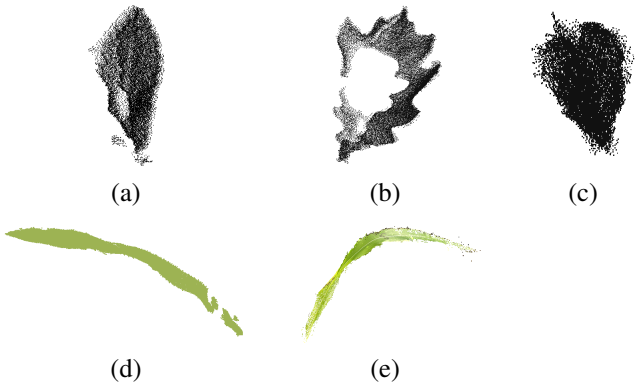


Fig. 3. Examples of real data: (a) sweet chestnut leaf, (b) red oak leaf, (c) silver birch leaf, (d) maize leaf obtained with a LiDAR, (e) maize leaf obtained by photogrammetry.

We first used the data set described in [11] where the authors scanned three plants seedlings using a Leica Geosystems HDS-6100 LiDAR from three positions at 1.5 m: a sweet chestnut (*Castanea sativa* Mill.), a red oak (*Quercus rubra* L.), and a silver birch (*Betula pendula* Roth). The point clouds were then segmented as detailed in [11]. For our study we only considered the segments corresponding to leaves. All leaves in this data set are non-planar. In addition, some sweet chestnut leaves suffer from misalignment and holes caused by insects. Red oak leaves are all concave and three of them contain holes. The silver birch leaves exhibit significant noise, mostly located on their boundaries. The reference leaf areas were measured in the laboratory with a laser area meter (CI-203, CID, Camas, WA).

The last plant is a maize plant (*Zea mays* L.) scanned with two different devices: a Faro 3D X130 laser scanner, and a multiview system made of 9 Sony-RX0 cameras. We used the Colmap software [31] to generate the 3D point clouds from the images. Leaves are elongated, curved and show some torsion. Both the LiDAR and photogrammetry point clouds contain occluded areas, hence holes. The registration of LiDAR point clouds led to small but noticeable misalignments. The photogrammetry datasets exhibit a variable point density. The leaf areas were measured with a Licor LI-3100 Leaf Area Meter.

3.2. Geometric models for leaves

In this section, we present the eight geometric models and methods we selected for comparison. Since our goal is to provide a practical guide for users in application fields (e.g. plant physiology, agronomy, etc.), we explain how to use them and highlight their parameters. In our comparative study, we have chosen the parameters for each algorithm after a sensitivity analysis, see Section 3.3.1 and Table 4.

3.2.1. Mesh reconstruction

We selected four simple and readily available meshing methods, that have already been applied to leaf surface reconstruction. We discarded those requiring additional manual work, such as the ones presented in [32, 10].

2.5D Triangulation. Considering a leaf is a 2-dimensional object, the simplest approach to reconstruct its surface is to triangulate the projections of its points onto a plane and to lift the resulting triangles back to the original points [11, 9]. This process, sometimes called *2.5D triangulation*, uses the least-square fitting plane to project the 3D points followed by a 2D Delaunay triangulation. In our tests, we use the CloudCompare software [33] to perform such computations. It adds an optional post-processing stage which removes triangles with edges longer than a given threshold. We introduce a meta parameter *dmn* related to the dimensions of the bounding box in order to adapt the threshold to the leaf size. This threshold is the only parameter of the 2.5D triangulation and the post-processing filter is disabled by default. Our sensitivity analysis demonstrated that ignoring the post-processing gives, on average, optimal results.

Alpha-Shape. The boundary of a Delaunay triangulation is the convex hull of the point set. Thus, it is a poor approximation for concave objects. As a subset of a Delaunay triangulation, alpha-shapes are better suited to mesh concave objects [24]. They depend on a single parameter $\alpha \geq 0$ restricting the set of valid triangles, hence allowing for non-convex reconstructions. When $\alpha = +\infty$ the alpha-shape is the convex hull of the input points, while when $\alpha = 0$ it degenerates to the set of points. Alpha-shapes have been used in 2D to estimate the surface of plant leaves [12]. They have also been used in 3D [13], with an empirically chosen value $\alpha = 0.6$. However in this case the reconstructed surface is watertight, meaning that a leaf is approximated by a thin volume. Since the result is a volume with a negligible thickness, the leaf area can still be approximated by halving the total mesh area. Alpha-shapes are implemented in Meshlab [34] and in the Open3D library [35]. In our tests we use 3D alpha-shapes as implemented in Open3D. The optimal α value depends on the point cloud density and the size of the smallest feature. We have developed an automatic optimization method to set the α value for our point clouds based on distance to the nearest neighbour statistics in the point cloud. Two statistics were tested independently, median and first-decile (see Section 3.3.1). Our tests have shown that using the median gives slightly better results than using the first-decile statistics. As a consequence, only the first is used in the followings, and we define the median distance mc_α between a point and its closest neighbour over the point cloud, multiplied by a set factor, as the sole parameter for our study. Our sensitivity analysis has shown that two different values of factor are optimal depending on the criterion to test: $mc_{\alpha 1} = 1$ for data with concavities, and $mc_{\alpha 2} = 15$ in all other cases

Ball-Pivoting Algorithm (BPA). The Ball-Pivoting Algorithm (BPA) [36] is a classical 3D mesh reconstruction method, that outputs a subset of an alpha-shape as a 2-manifold mesh interpolating the input points. Starting from a seed triangle, it incrementally builds the mesh by pivoting a ball around the boundary edges until it hits a point, from which a new triangle is created. The parameter of the BPA is the ball radius ρ. The algorithm is provided in Meshlab and the Open3D library. We have chosen to use Meshlab [34], since it proposes an improved version that produces more consistent meshes than the strict implementation found in Open3D, which generates non-watertight meshes with holes, as we will explain later. This version comes with two additional parameters: a clustering radius cr and an angle threshold θ. The value of ρ is automatically chosen based on the bounding box diagonal of the point cloud.

Incremental Reconstruction. Developed with robotics applications in mind, the mesh reconstruction algorithm described in [37] is designed to be fast and robust to noise. To these ends, it is incremental and resamples the input point cloud. As a consequence, it neither requires the surface to be closed nor interpolates the data. This algorithm is implemented in the Point Cloud Library (PCL) [38] and has been used in [14] for leaf area estimation after downsampling and smoothing the input point cloud. This algorithm has seven parameters, but the most impactful one is the nearest neighbour distance ratio μ, set to 5 in our tests as recommended in the original paper. The others are set to their default values after the sensitivity analysis.

3.2.2. Parametric modelling

In the context of Functional-Structural Plant Modeling (FSPM), parametric surfaces have been used to approximate leaf shapes. Usually, the proposed models are flexible and adapt themselves to morphological shape variations. We only focus on 3D point clouds in our study, hence we discard methods requiring additional information, such as veins or the midrib of the leaves [17, 39, 40, 41]. There are two main methods that respect this constraints: B-spline and Bézier surfaces.

B-spline. [23] compare leaf area estimation using B-Spline surfaces and mesh models. In that study, a regular 2D grid is first defined using the principal axes of the point cloud to define the knot points of the spline. The B-spline is then trimmed by computing the boundary of an alpha-shape. Recently Harmening and Paffenholz have proposed an alternative B-spline-based method to compute individual leaf area [21]. This method relies on the accurate computation of boundary curves around the leaf point cloud. In our study, we have chosen to consider the approach proposed by [25] for two reasons. First, it is implemented in the PCL, and hence is easy to use for a biologist. Second, the trimming is done robustly using a closed B-spline curve carefully designed to handle concavities. This approach requires tuning 19 parameters. However, as stated in [25], a standard set of values provides satisfactory results for most point clouds. Indeed, for most parameters the default value is robust to a large variety of input point clouds. Most of the parameters are dedicated to a very specific use case (e.g., *boundary_weight* allows to adapt the surface boundary to the boundary points). We refer to the PCL [38] for more details. We thus use these default values, except for the B-spline order o, the number of iterations i and the mesh resolution rs. Finally the refinement parameter rf has a great impact on the computation time, as shown in Table 3. We made a trade-off between the global accuracy of the final mesh and the computation time to choose its value after a sensitivity analysis.

Table 3. Execution time (s) for 10 point clouds with B-Spline order 3, number of iterations equal to 5, and mesh resolution set to 64.

Nb of points \ Refinement	2	3	4	5
1000	3	5	60	4185
5000	10	19	171	5804
10000	22	39	318	7716

Bézier Leaf Model. A parametric leaf model based on Bézier surfaces has been proposed by Chaurasia and Beardsley [19]. In this model, a leaf is represented as a Bézier surface derived from three main components: the leaf midrib, together with its left and right silhouettes. The model uses 3^{rd} degree Bézier curves to approximate these components (4^{th} degree for the midrib), along with four additional Bézier curves describing the interior part of the leaf. Given the midrib curve and the silhouettes, a UV mapping of the point cloud is computed from which the model's parameters are optimised, including the midrib curve. The resulting Bézier surface is then regularly sampled and meshed. To our knowledge, no implementation of this model is publicly available. Therefore in this study, we have used our own implementation. The main parameter to set is the number of scanlines s, defined at regular intervals of the midrib curve.

Trimmed Bézier. The use of a midrib curve along with silhouette curves in the work of [19] matches botanical observations of real leaves. However, this specific model induces rigidity, notably with the imposed symmetry around the midrib. Therefore we also considered general Bézier surface modelling to alleviate this limitation and to allow reconstructed surfaces to closely match the input point clouds. In this proposed method we first used an affine transformation to project the leaf point cloud onto the 2D domain using the best fit plane where the transformed point cloud fits in the unit bounding box (from (0,0) to (1,1)), and the base and apex of the leaf are set to (0.5, 0) and (0.5, 1) respectively. We then considered this projection as a uv mapping of the input 3D point cloud and used linear least-square fitting to obtain the desired Bézier surface. As the last step of this method, we propose an additional trimming stage inspired by the scanlines of [19]: the Bézier surface is only meshed in the $[u_{min}, u_{max}]$ interval in each of the n bins of v parameter. The proposed method is based on four parameters: dU and dV indicate the degree of the desired Bézier surface, $meshU$ and $meshV$ the meshing resolution along u and v (the number of scanlines n is automatically set equal to $meshV$).

3.2.3. Implicit modelling

Implicit modelling has been applied with success to point clouds and constitutes a major approach of surface reconstruction from such data [42, 43, 44, 45]. Methods from this ap-

proach fit an implicit function to a set of points. A level-set of this function is then extracted and meshed to approximate the sampled surface.

Screened Poisson Reconstruction. Surface reconstruction from a point cloud with normal vectors can be formulated as a Poisson problem. This formulation was proposed by Kazhdan et al. to reconstruct watertight surfaces [43] and was further extended to open surfaces [44]. This method is known as *screened Poisson reconstruction*. The authors provide the source code as well as executable software on their web page together with usage examples. The main parameter of the screened Poisson surface reconstruction algorithm is the importance *pw* of point interpolation during the screened Poisson formulation (so-called point weight in the provided software). The other parameters are called tree depth *td*, exponent scale *es* and accuracy *acc*. They are set as shown in Table 4. This method was used in [22] to compute maize leaf areas, but additional manual trimming was required. The output of **Poisson** surface reconstruction is a mesh with an estimation of local point density associated with each vertex. [44] then recommended trimming the resulting mesh according to this density value, i.e. to remove vertices corresponding to a density lower than a user-defined threshold. The sensitivity analysis led us to use the mean of local density over the point cloud as the threshold *tt*.

Table 4. Optimal value(s) for each parameter of each method.

Method	Parameter (unit)	Optimal value(s)
2.5D Triangulation	*dmn*	0
Alpha-Shape Median	mc_α	1, 15
BPA	ρ (% *BBOX*)	0
	cr (% ρ)	50
	θ (°)	180
Incremental Reconstruction	μ (% *BBOX*)	5
	Radius (% *BBOX*)	0.025
	Number of points	100
	Maximum surface angle (*rad*)	$\frac{\pi}{4}$
	Minimum angle (*rad*)	$\frac{\pi}{18}$
	Maximal angle (*rad*)	$\frac{2\pi}{3}$
	Normal consistency	false
B-Spline	*o*	3
	rf	3
	i	5
	rs	64
Bézier Leaf Model	*meshU*	5
	meshV	40
	s	50
Trimmed Bézier	*dU*	2
	dV	4
	meshU	5
	meshV	40
Poisson	*td*	8
	pw	4
	es	3
	acc (*m*)	0.001
	tt	50th -percentile

3.3. Methodology

We evaluated the accuracy of the selected methods by comparing their results to the ground truth values. For each method, we used the set of best parameters identified through the sensitivity analysis described in Section 3.3.1. Each method has been applied on each of the 1240 synthetic leaf models generated as described in Section 3.1.1. Additionnally, for each real dataset presented in Section 3.1.2, methods have been tested on raw and on filtered point clouds (see Section 3.3.2).

Once the leaf areas were estimated, each method has been evaluated based on its signed mean relative error (SMRE):

$$SMRE = \frac{1}{n}\sum_{i=1}^{n}\frac{A_{e_i} - A_{t_i}}{A_{t_i}}$$

where n is the number of point clouds considered and A_{t_i} and A_{e_i} are the ground truth area and the estimated area for the ith point cloud, respectively.

We have also used the unsigned mean relative error (UMRE), defined as $UMRE = \frac{1}{n}\sum_{i=1}^{n}\frac{\|A_{e_i} - A_{t_i}\|}{A_{t_i}}$ to determine the norm of the relative error on each dataset, in order to check for compensation phenomenon in the SMRE computation. According to our study the UMRE is less informative than the SMRE and the reliability of the SMRE is enough to draw our conclusions.

3.3.1. Sensitivity analysis

We performed a parameter sensitivity analysis to determine which parameters need to be carefully tuned in order to reach the most accurate leaf area estimate with each method, and how to tune these parameters. In addition to the leaf area, we also checked the geometry and topology of the generated surface to discard inaccurate surfaces: parameter values leading to non-manifold surfaces, surfaces with holes or self-intersections are considered as non-optimal regardless of the accuracy. A subset of the synthetic data described in Section 3.1.1 has been randomly chosen and used to conduct this analysis. 130 of the 1240 generated point clouds have been clustered according to the 11 criterion parameters. This allowed us to test each method parameter with respect to each criterion independently.

We varied the value of each parameter individually according to the identified set of values to be tested. Table 4 summarizes the optimal values found. The SMRE was computed for each parameter value among all point clouds of a data set defining a criterion parameter. Among all tested values of a parameter, the one that reaches the SMRE closest to zero among all data sets while generating a geometrically and topologically correct surface was chosen as the optimal value of the parameter. Finally, we have checked that the combination of optimal values for all parameters of the method generates better results than the combination of one optimal value for a given parameter and default values for the other ones.

For some methods, parameters may depend on the point cloud density or size. In such cases, a fixed value for such parameters is not suitable. Therefore we also introduce *meta-parameters* linking method parameters and point cloud features and have optimized their values using the same strategy as described above. For **2.5D Triangulation** we propose a meta-parameter *dmn* defined as the diagonal of the point cloud's bounding box divided by the number of points in the cloud. For **Alpha-Shape** we propose to compute the distribution of the distances from each point to its closest neighbour. We have

tested both the median mc_α and the first decile dc_α of this distribution as a meta-parameter. When studying the median, we noticed that results for the **Co** criterion were very different from results in all other cases. This is why we decided to keep two optimal values for mc_α instead of only one. Finally for **Poisson** reconstruction, we use the mean of the local point cloud density as the trimming threshold *tt*.

3.3.2. Point cloud pre-processing

In this study, we assume that the point clouds have been segmented to only include leaf. However, input point clouds may still contain noise. As an additional contribution, we studied the effect of pre-processing filters on the leaf area estimated by the selected methods. To do so we selected two classical filters. The Statistical Outlier Removal filter (SOR) [46] is based on the distance from a point to its neighbours: it discards points with a distance higher than the standard deviation computed on the entire point cloud. Based on the definitions of Moving Least Square surfaces (MLS) [47], the MLS filter proposed in CloudCompare projects each data point on a smooth surface defined as the stationary point of a projection operator. While it can be used to up- or down-sample a point cloud, we only used it to reduce the amount of noise in the point clouds. Many variants of the MLS approach have been proposed in the literature. In addition to the version proposed in CloudCompare, we have experimented the Algebraic Point Set Surface (APSS) definition proposed by [48]. We ran the leaf area estimation methods on raw input point clouds as well as SOR filtered point clouds, MLS filtered point clouds and SOR + MLS filtered point clouds. Results show a negligible difference of leaf area estimation between raw data with SOR filtered data on the one hand, and between MLS filtered data and SOR+MLS filtered data on the other hand. This demonstrates that the SOR filter may be useful to remove major outliers but is not enough to improve leaf area estimation in noisy point clouds. On the contrary, the MLS filter both improves the accuracy of all methods and reduces the dispersion of the estimates. Results are similar with both MLS definitions (see Fig.7), except for **Poisson** reconstruction. Consequently, we have only considered the data filtered with the MLS filter proposed in CloudCompare in our study.

4. Results

The methods described in Section 3.2 have been tested on the data introduced in Section 3.1. Tests on synthetic data allowed us to evaluate the accuracy of the methods with respect to each criterion independently. The results of this quantitative assessment are presented and discussed in Section 4.1. We draw general comments about the capability of each model to handle real-life cases based upon the tests on real acquisitions in Section 4.2.

4.1. Results on synthetic data

The influence of each criterion on the area estimation is illustrated in Fig. 4. Each curve shows the leaf area SMRE for a given method with respect to a criterion parameter value. Note that each computation has been done ten times to factor randomness in generated synthetic data. Each dot in each curve of Fig. 4 shows the average leaf area over these ten trials. These results on synthetic data provide the material to discuss the effects of each criterion separately and to highlight specific cases of failure.

4.1.1. Leaf shape criteria

Overall, the length/width ratio has a low impact on the area estimation: SMRE does not vary much (at most 0.05), except for **Poisson** reconstruction. This is because the trimming step, based on density, in **Poisson** reconstruction reaches limitations. As the density increases in one preferred direction with the ratio, the expected accuracy in the orthogonal direction decreases at the trimming step.

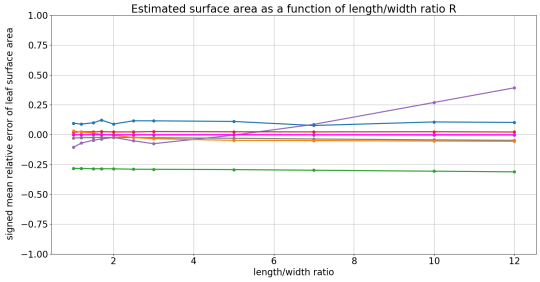
The most robust methods to curviness are the **BPA**, the **Incremental reconstruction** and **Poisson** reconstruction, which produce consistent meshes even for highly curved leaves. **Trimmed Bézier**, **Bézier Leaf Model** and the **2.5D Triangulation** assume that a leaf is similar to an elevation surface. Therefore, these methods are not suited to reconstruct leaves with a curviness higher than 180° where folds appear and their accuracy decreases (Fig. 6-(a)). When curviness reaches almost 360°, the **Alpha-Shape** and **B-Spline** methods produce erroneous meshes, as illustrated in Fig. 6-(b). Therefore, even though the estimated areas may be accurate with respect to the ground truth, these methods cannot be applied to such data representing rolled leaves. Moreover the **Alpha-Shape** produces 3D volumes that do not take into account the curvature, which explains why the computed surface area decreases when the curvature increases.

According to our experiments, all methods give almost constant leaf area estimates with respect to concavity. Even though the **Bézier Leaf Model** and the **Trimmed Bézier** models are not expected to correctly capture concave shapes, we hypothesize that they benefit from concavities being evenly distributed around the boundary of the reconstructed surface (Fig. 6-(c)), which could not be the case for more complex concavity models. Since the **2.5D Triangulation** is based on a Delaunay triangulation, the resulting mesh is very close to the point cloud convex hull, leading to an area overestimation when the shape is concave.

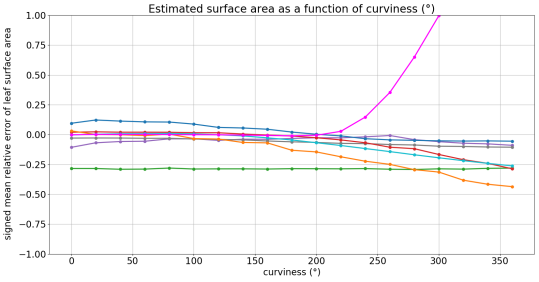
4.1.2. Sampling criteria

All methods are stable with the regular sampling, whatever the amount of horizontal noise **S.n**, given that enough points are produced.

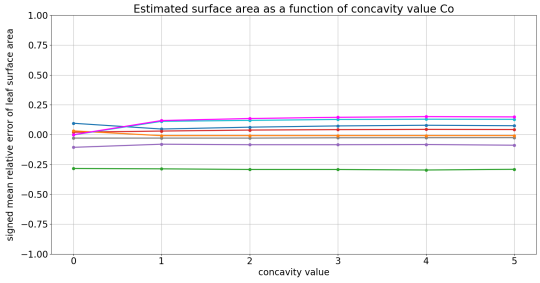
Unsurprisingly, in the case of a sufficiently high point density ($\geq 1,000$ points per unit area), area estimation is very stable for all methods, except **Incremental Reconstruction** which needs around $6,000$ points to be accurate. When density decreases, all methods tend to underestimate the leaf area due to the appearance of holes in the reconstructed meshes, or because the leaf boundary is no longer accurately described by the point cloud. The more consistent methods with respect to this criterion are **Poisson** reconstruction and the **2.5D Triangulation**.



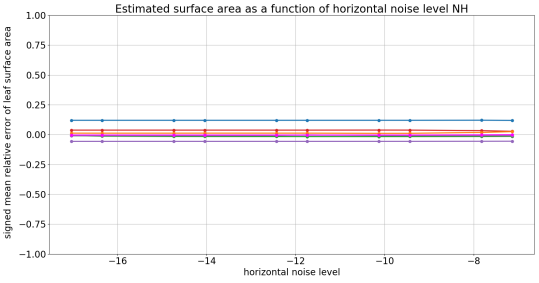
R



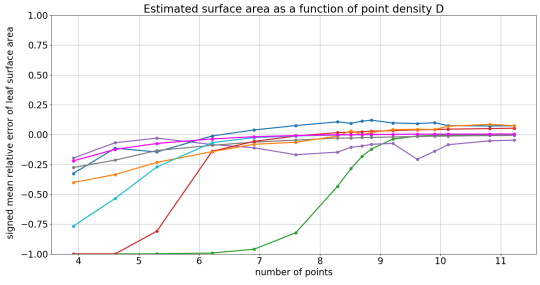
Cu



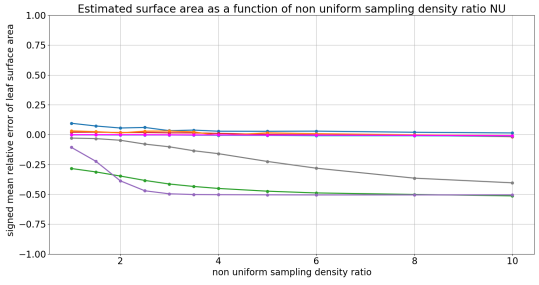
Co



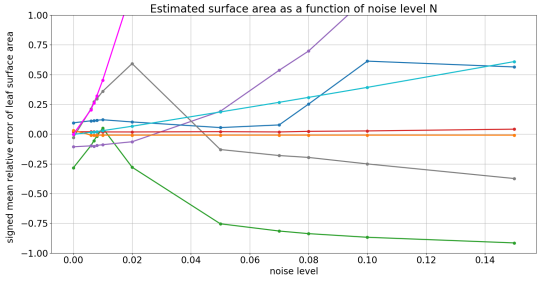
S.n



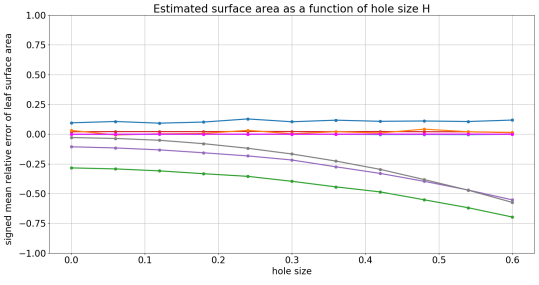
D



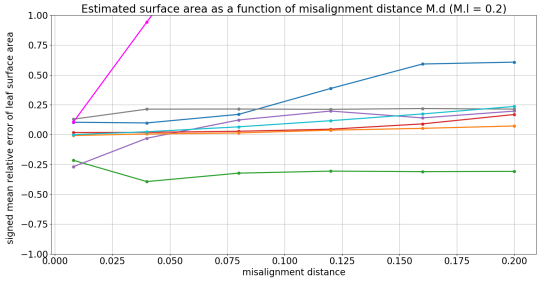
NU



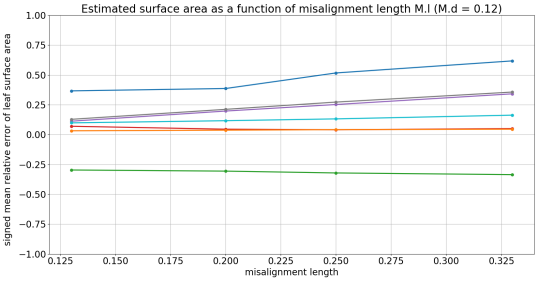
N



H



M.d



M.l

Fig. 4. Influence of each criterion on the leaf area estimation using 2.5D Triangulation, Alpha-Shape Median, BPA, Incremental Reconstruction, B-Spline, Bézier leaf model, Trimmed Bézier and Poisson. More results for misalignment can be found in Fig. 5.

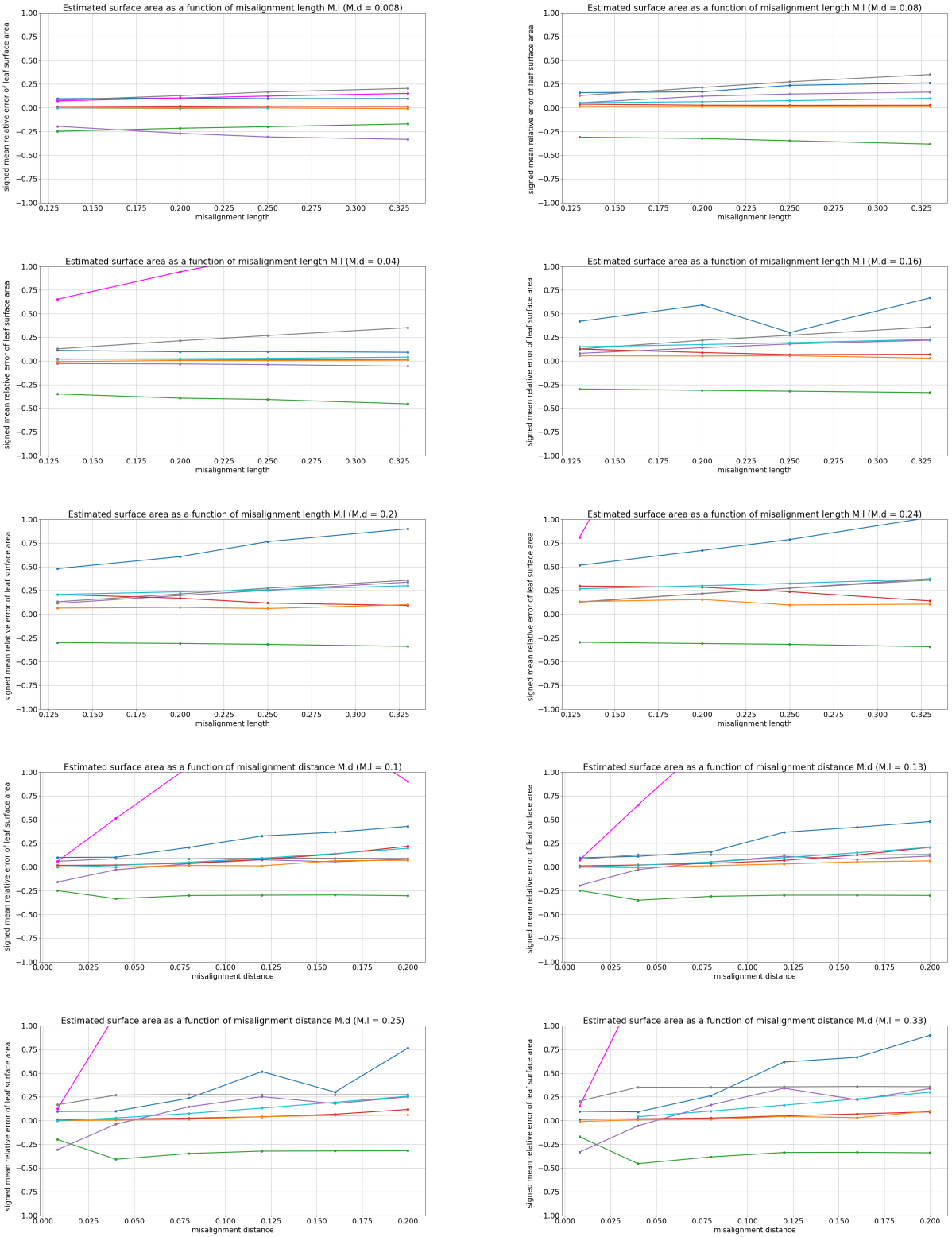


Fig. 5. Misalignement results on synthetic data, for different values of M.d and M.l.

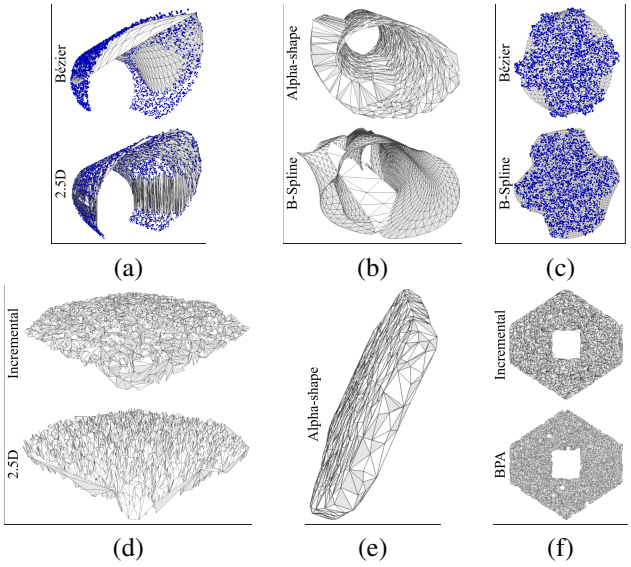


Fig. 6. Some specific cases where the quality of the surface reconstruction impacts the accuracy of the leaf area estimation: (a,b) Cu , (c) Co, (d,e) N, (f) H.

All methods are stable with nonuniform sampling except **Incremental Reconstruction**, the **BPA** and **Poisson** reconstruction. This is because **Incremental Reconstruction** and the **BPA** depend on the uniformity of the distances between points to create the mesh. As for **Poisson** reconstruction, its statistical parameters are skewed with nonuniform sampling and the trimming step is incorrect.

4.1.3. Acquisition defects

The parametric models are by far the most robust to noise. Since **2.5D Triangulation** directly triangulates the input points, it produces many small triangles for which height is directly correlated to the noise strength (Fig. 6-(d)). This artificially increases the computed area, which is then always overestimated. The same problem arises with **Incremental Reconstruction**, but in this case, the area is underestimated because the method does not allow for large triangles, leading to holes in the mesh. For a very small amount of noise, the **Alpha-Shape** surface is close to being two parallel sheets. Its area is thus twice the area of the leaf. For a higher noise amount, the surface does not look like a thin object anymore, and the overestimation arises from the additional volume boundary (Fig. 6-(e)). The trimming step in **Poisson** reconstruction, being based on point density, is compromised by the noise. Results of the **BPA** are quite unpredictable in presence of noise, as evidenced by the chaotic behaviour of the related curve. Either a surface or a volume is created, making it difficult to establish a general rule for area computation, and the density unevenness introduced by the noise leads to many holes in the reconstructed mesh.

The more robust approach to misalignment is the **Incremental Reconstruction**. This is because its triangle size restriction allows the reconstruction of two separate surface sheets. The **2.5D Triangulation** is highly sensitive to misalignment, leading to large leaf area overestimates ($SMRE > 1$, hence the corresponding curves are not visible on the graphs in Fig. 4 and 5). This method creates triangles between the two parts of the surface. A great misalignment length leads to many triangles while a large misalignment distance leads to big triangles, meaning the leaf area overestimation by the method increases as soon as the value of one of the two parameters increases. The **BPA** is sensitive to misalignment length while the **Alpha-Shape** and to some extent **Bézier Leaf Method** and **Trimmed Bézier** are sensitive to misalignment distance. In the **BPA** case, this is because the bounding box of the point cloud is much more modified by the misalignment length than by the misalignment distance. As for both **Bézier** methods, the projection step cancels the misalignment length problem, but not the misalignment distance. **B-spline** is robust to misalignment when the distance between the two parts of the point cloud is small (< 0.05), because this case is similar to a single surface with nonuniform sampling and noise, two criteria this method is robust to. On the contrary **Poisson** reconstruction underestimates the leaf area in this case since it is not robust to these criteria because of density modification with the global shape. When the distance is too large **B-spline** overestimates the leaf area as the length of misalignment increases, as it tries to create a single, folded surface approximating the two parts of the alignment. For a better understanding, we refer to the graphs in Fig. 5 showing the behaviour of the methods for various values of the two parameters **M.d** and **M.l**.

All methods are robust to holes except the **BPA**, the **Incremental Reconstruction** and **Poisson** reconstruction. Since the **BPA** and the **Incremental reconstruction** exploit the point cloud density, holes greater than a threshold are not meshed, leading to an underestimation of the area (Fig. 6-(f)). The trimming step in **Poisson** reconstruction being based on the local point density, the result is similar.

4.2. Results on real data

We evaluated the accuracy of each selected method on the point clouds derived from real acquisitions. In this evaluation, we considered as error measure the signed relative error between the estimated leaf area and the ground truth as recovered from manual measurements. Fig. 7 shows the box-and-whisker charts of errors for all leaves of each plant and each geometric model. Point clouds were filtered using either the MLS filter proposed in CloudCompare or the APSS framework of [48], with 30 nearest neighbours and 5 successive projections on the implicit surface.

For the sweet chestnut, the red oak and the silver birch data, the most robust and accurate methods are the **Bézier Leaf Model** and the **Trimmed Bézier** method. It is to be noted that the **B-Spline** method has failed to produce a mesh for some leaves, leading to a zero estimate for their area. As for the **2.5D Triangulation** and the **Screened Poisson** method, they tend to overestimate the leaf areas. This can be explained by holes caused by insects being filled as well as concavities.

Results on the maize scanned with a laser scanner and the maize acquired with a multiview system are very similar, indicating that the acquisition process has little impact on the methods' efficiency. Most of them exhibit a large dispersal of the computed leaf areas, meaning they are not robust. This is because the leaf geometry causes problems, more precisely the

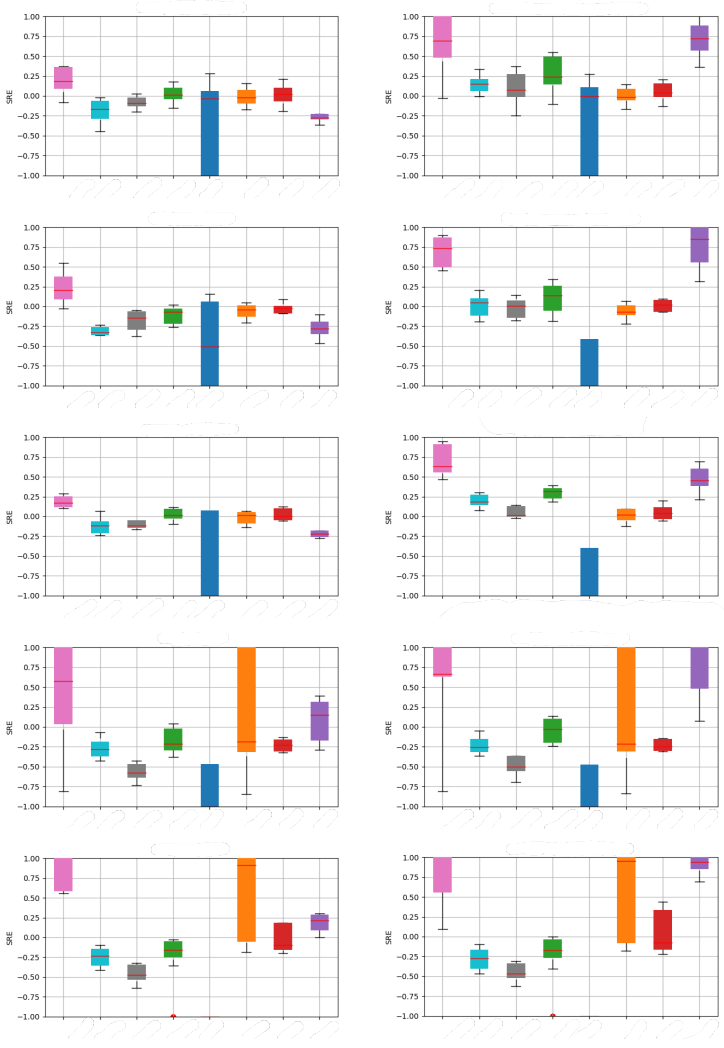


Fig. 7. Box-and-whisker charts of leaf area SMRE on real data filtered using either the CloudCompare MLS filter (left) or the APSS framework (right), using 2.5D Triangulation, Alpha-Shape Median, BPA, Incremental Reconstruction, B-Spline, Bézier leaf model, Trimmed Bézier and Poisson: from top to bottom, sweet chestnut, red oak, silver birch, maize (LiDAR acquisition) and maize (photogrammetry acquisition). Whiskers represent minimum and maximum values of the computed errors for a given model on a given dataset (excepting outliers), a box represents errors between the 25th and 75th percentile, the line inside this box shows the median error. Computed errors are considered as outliers if they lie 1.5 times the length of the box, from either of its ends.

high **R** for **Poisson** reconstruction, the high **Cu** for the **B-Spline** method, and the combination of both for the **2.5D Triangulation** and **Bézier Leaf Model**, since both have trouble finding the right projection plane for the data. The four other methods are more robust, but **Alpha-Shape** and **BPA** however underestimate the leaf area. The most accurate methods are the **Trimmed Bézier** method and the **Incremental Reconstruction**.

These results are consistent with those on synthetic data since the behaviour of each model can be explained by the combination of multiple criteria. However, results on real data highlight the impact of additional criteria on the estimated areas, such as the intrinsic nature of the noise near leaf borders like in the birch dataset or the presence of holes separating a point cloud into multiple clusters like in the oak or the maize photogrammetry datasets.

We show in Fig. 8 to 10 the meshes computed by each method for one leaf of each dataset.

5. Discussion

From our experiments, we can draw three conclusions. As a first conclusion, our sensitivity analysis has highlighted the benefit of using an automatic and objective solution for setting each parameter of the selected method to an optimal value. In particular, since it appeared that the behaviour of many methods highly depends on the density of the point cloud, we have proposed to automatically set the main parameters of **2.5D Triangulation**, **Alpha-Shape** and the trimming step of **Poisson** reconstruction using meta-parameters based on the global or local density of the point cloud. Overall, our sensitivity analysis showed that the choice of parameters influences all methods except for the three parametric models (**B-Spline**, **Bézier Leaf Model** and **Trimmed Bézier**) which tend to be rather unaffected. The tuning of the **Incremental Reconstruction** method across our synthetic data was the most sensitive.

A second conclusion drawn from the results obtained on synthetic data sets is that the leaf curviness, acquisition noise and misalignment are the criteria impacting the most the accuracy of the results. The parametric models we tested are robust to noise as well as to misalignment up to a threshold, but cannot properly fit a highly curved leaf. The **Incremental Reconstruction** method is robust to curviness and misalignment but is sensitive to noise.

An additional conclusion derived from the experiments on real data sets is that the MLS filter improved the accuracy and robustness of all methods, but not the SOR filter. Hence we recommend systematically filtering the raw point clouds with this filter before further processing.

These conclusions lead us to recommend the use of the MLS filter combined with the **Trimmed Bézier** method, which is the most robust and accurate method in most cases and is easy to set: the parameter values $dU = 2$, $dV = 4$, $meshU = 5$ and $meshV = 40$ can most often be used. However, in the case of a plant with highly curved leaves such as the maize we used for our tests, this method may underestimate the leaf areas. Since results are consistent over the plant leaves (Fig. 7), a solution can be to manually measure a batch of individual leaves, report

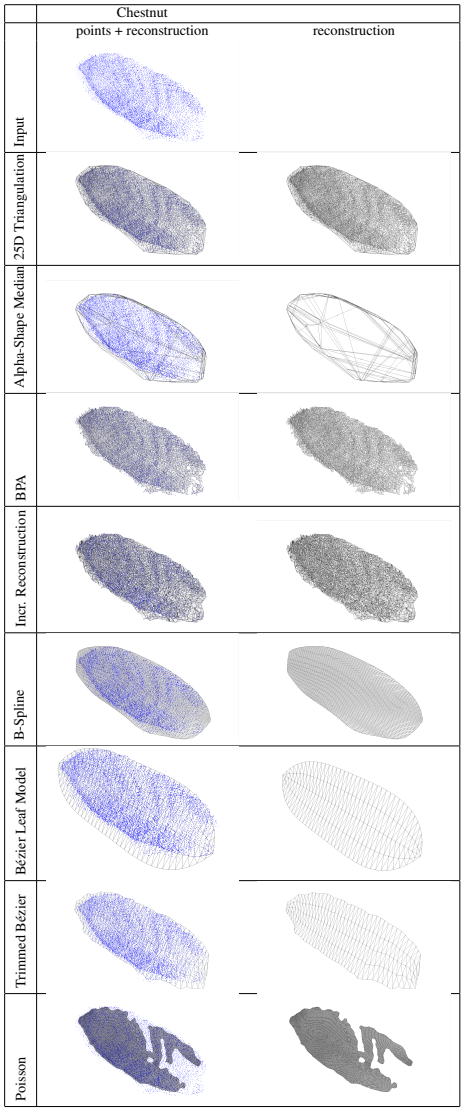


Fig. 8. Reconstructed meshes on a sweet chestnut leaf.

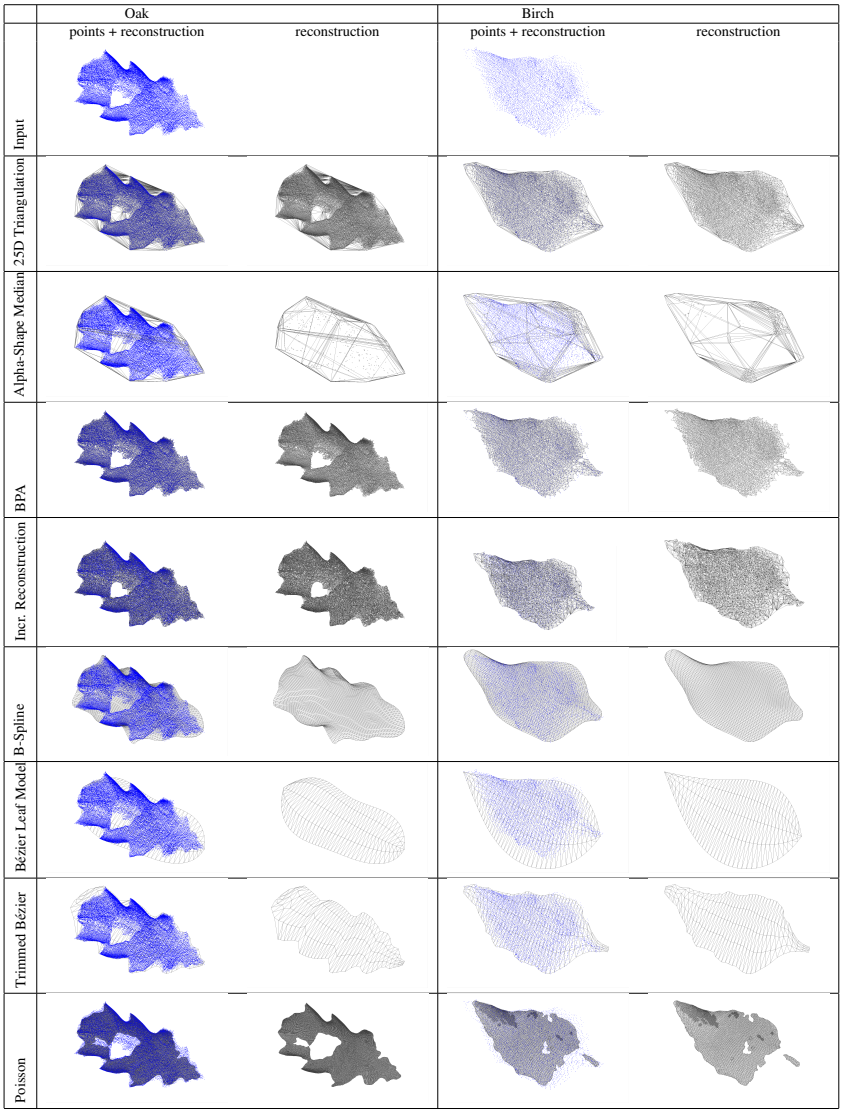


Fig. 9. Reconstructed meshes on a red oak and a silver birch leaves.

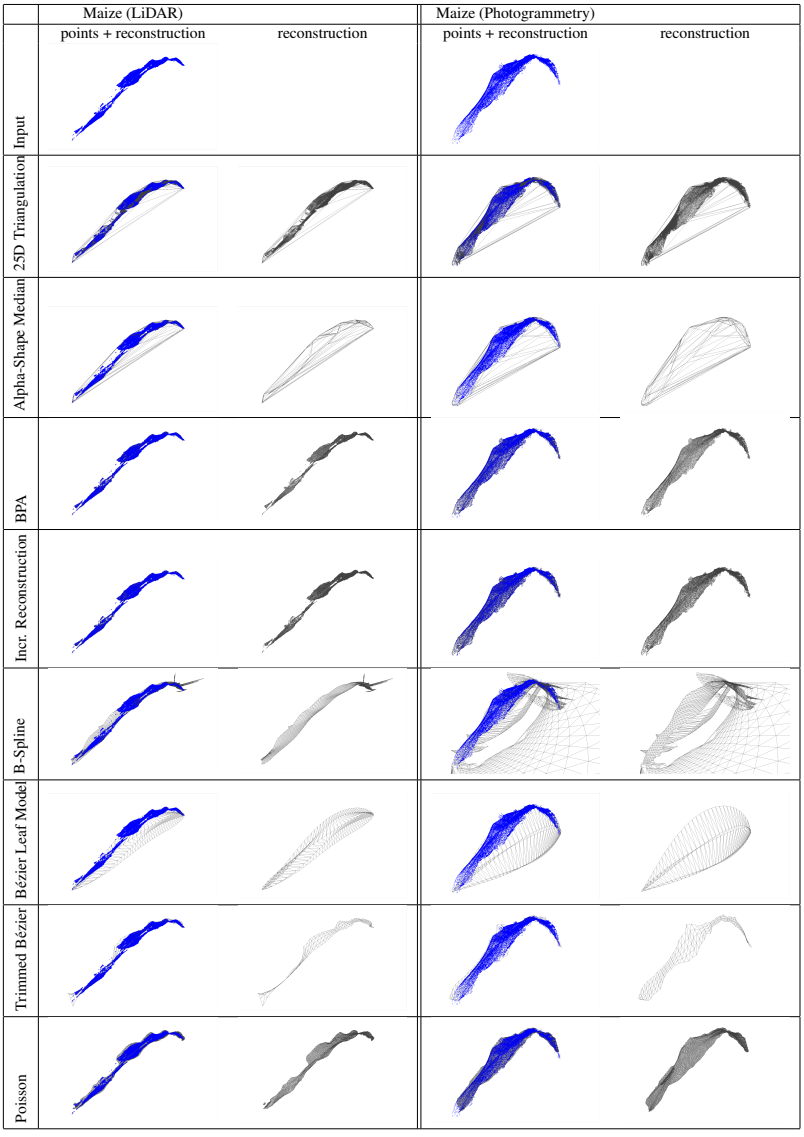


Fig. 10. Reconstructed meshes on maize leaves.

the ratio between the measured and the computed area for each leaf, compute the mean ratio and then multiply the computed area for any other leaf of the plant by this mean ratio.

6. Conclusion

Many parametric, implicit or mesh-based models have been designed to reconstruct a continuous surface from a point cloud and successfully used to measure the geometry of the surface (area, normal, curvature, ...). In order to automatically estimate the leaf area of small plants, seedlings and crops from 3D point clouds, and despite the relatively complex geometry of their leaves, plant biologists usually use simple, generic and readily available geometric models. In this paper, we evaluated six of them for this specific purpose, as well as one parametric model dedicated to this aim [19] along with a more classical Bézier surface fitting. We have defined three criteria about the leaf geometry and six criteria about the acquisition process that could affect the robustness and the accuracy of these geometric models. We have designed 1240 synthetic leaf point clouds based on these criteria and tested the methods on them, after a careful sensitivity analysis that led us to design an automatic way to set the parameter values for each of them. We have also evaluated the methods on six acquisitions of real plants that had been previously manually measured and we have tested the additional effect of denoising the point clouds as a pre-processing step.

Our results show that the use of both a moving-least squares (MLS) filter and **Bézier** surface fitting is recommended. However, this may not work in some cases and precisely estimating leaf area from point clouds remains a challenging issue. Our analysis is a first work highlighting useful data pre-processing and assessing how various geometric models can estimate leaf areas. It is inherently limited by the data and criteria we have selected. Our synthetic data was generated starting from a simple hexagon, which is not adapted to the modelling of multi-lobed leaves. Another difficulty with real-life scenarios is that data is generally missing on the surface boundary. In this case, a geometric model should take into account the expected boundary shape as prior knowledge in order to be robust.

Three directions seem promising to overcome such limitations. Similar to [19], the first one is to design a model integrating some prior knowledge about the leaf geometry. However, such a model would, by nature, be specific to some plant species. The second research direction is to investigate more advanced generic approaches, such as the combination of partitioning and local modelling as in [49, 15], or local flattening [50]. The last research direction is to use machine learning to implicitly incorporate prior knowledge about the leaves while maintaining a generic approach. Since manual measurement is a tedious task, a promising idea would be to learn on synthetic data, despite the difficulty to mimic the real world. The criteria we propose in this paper are a first step towards this goal. However, as shown on real leaves, more work is necessary to account for more complex phenomena such as the leaf torsion, the presence of outliers (since a clean filtering or segmentation is not always easy to recover), occlusions leading to

holes at the leaf boundaries, as well as the combination of several criteria. Nevertheless, a generic method will only be useful in practice for plant experts if its code is freely available and easy to use. Moreover, its parameters should be meaningful and easy to tune, whatever the species and the acquisition conditions. Our code to generate the synthetic data associated with our study, as well as for both Bézier methods, has be released for free[1]. Since the other selected approaches have been implemented in freely available software or libraries, this allows both plant practitioners to compare the geometric models on their own data and researchers to evaluate other models.

Acknowledgements

The authors would like to thank Frédéric Larue for his initial participation to this project, Éric Casella for the tree seedling acquisitions as well as useful suggestions, and Jonas Mehtali for his help segmenting the maize data.

Author contributions

MB, JR and FHW designed the research; FHW and MB designed and coded the synthetic data generation method; MB and BdS performed the real data acquisition; MB and JR coded the Alpha-Shape and the Bézier Leaf Model methods; JR designed and coded the Trimmed Bézier method; MB performed the filtering and designed and performed the sensitivity analysis; all authors analysed the results; MB, JR and FHW wrote the paper; all authors proof-read the paper.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

MB is supported by a CIFRE PhD grant funded by Arvalis and the Association Nationale Recherche Technologie (ANRT). JR and FHW are supported by the University of Strasbourg through the "IDEX Attractivité" grant W18RAT96. BdS acknowledges the support of Arvalis.

References

[1] Madec, S, Baret, F, de Solan, B, Thomas, S, Dutartre, D, Jezequel, S, et al. High-throughput phenotyping of plant height: Comparing unmanned aerial vehicles and ground lidar estimates. Frontiers in Plant Science 2017;8:2002. doi:10.3389/fpls.2017.02002.

[2] Lou, L, Liu, Y, Shen, M, Han, J, Corke, F, Doonan, JH. Estimation of branch angle from 3d point cloud of plants. In: International Conference on 3D Vision (3DV). 2015, p. 554–561.

[3] Casella, E, Sinoquet, H. A method for describing the canopy architecture of coppice poplar with allometric relationships. Tree Physiology 2003;23(17):1153–1170. doi:10.1093/treephys/23.17.1153.

[4] Mirabet, V, Besnard, F, Vernoux, T, Boudaoud, A. Noise and robustness in phyllotaxis. PLOS Computational Biology 2012;8(2):1–12. doi:10.1371/journal.pcbi.1002389.

[5] Dassot, M, Constant, T, Fournier, M. The use of terrestrial lidar technology in forest science: application fields, benefits and challenges. Annals of Forest Science 2011;68:959–974. doi:10.1007/s13595-011-0102-2.

[6] Vadez, V, Kholová, J, Hummel, G, Zhokhavets, U, Gupta, S, Hash, CT. LeasyScan: a novel concept combining 3D imaging and lysimetry for high-throughput phenotyping of traits controlling plant water budget. Journal of Experimental Botany 2015;66(18):5581–5593. doi:10.1093/jxb/erv251.

[7] Casella, E, Sinoquet, H. Botanical determinants of foliage clumping and light interception in two-year-old coppice poplar canopies: assessment from 3-d plant mockups. Annals of Forest Science 2007;64:395–404. doi:10.1051/forest:2007016.

[8] Gibbs, JA, Pound, M, French, AP, Wells, DM, Murchie, E, Pridmore, T. Approaches to three-dimensional reconstruction of plant shoot topology and geometry. Functional Plant Biology 2016;44:62–75.

[9] Paulus, S. Measuring crops in 3d: using geometry for plant phenotyping. Plant Methods 2019;15(103).

[10] Bradley, D, Nowrouzezahrai, D, Beardsley, P. Image-based reconstruction and synthesis of dense foliage. ACM Trans Graph 2013;32(4). doi:10.1145/2461912.2461952.

[11] Hétroy-Wheeler, F, Casella, E, Boltcheva, D. Segmentation of tree seedling point clouds into elementary units. International Journal of Remote Sensing 2016;37(13):2881–2907. doi:10.1080/01431161.2016.1190988.

[12] Pound, MP, French, AP, Fozard, JA, Murchie, EH, Pridmore, TP. A patch-based approach to 3d plant shoot phenotyping. Machine Vision and Applications 2016;27:767–779.

[13] Chaudhury, A, Ward, C, Talasaz, A, Ivanov, AG, Brophy, M, Grodzinski, B, et al. Machine vision system for 3d plant phenotyping. IEEE/ACM Transactions on Computational Biology and Bioinformatics 2019;16(6):2009–2022.

[14] Li, D, Cao, Y, Shi, G, Cai, X, Chen, Y, Wang, S, et al. An overlapping-free leaf segmentation method for plant point clouds. IEEE Access 2019;7:129054–129070. doi:10.1109/ACCESS.2019.2940385.

[15] Li, D, Shi, G, Kong, W, Wang, S, Chen, Y. A leaf segmentation and phenotypic feature extraction framework for multiview stereo plant point clouds. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing 2020;13:2321–2336.

[16] Loch, BI, Belward, JA, Hanan, JS. Application of surface fitting techniques for the representation of leaf surfaces. In: International Congress on Modelling and Simulation: Advances and Applications for Management and Decision Making (MODSIM). 2005, p. 1272–1278.

[17] Wang, H, Zhang, W, Zhou, G, Yan, G, Clinton, N. Image-based 3d corn reconstruction for retrieval of geometrical structural parameters. International Journal of Remote Sensing 2009;30(20):5505–5513. doi:10.1080/01431160903130952.

[18] Kempthorne, DM, Turner, IW, Belward, JA. A comparison of techniques for the reconstruction of leaf surfaces from scanned data. SIAM Journal on Scientific Computing 2014;36(6):B969–B988. doi:10.1137/130938761.

[19] Chaurasia, G, Beardsley, P. Editable parametric dense foliage from 3d capture. In: 2017 IEEE International Conference on Computer Vision (ICCV). 2017, p. 5315–5324. doi:10.1109/ICCV.2017.567.

[20] Oqielat, MN. Surface fitting methods for modelling leaf surface from scanned data. Journal of King Saud University - Science 2019;31(2):215 – 221. doi:10.1016/j.jksus.2017.03.008.

[21] Harmening, C, Paffenholz, JA. A fully automated three-stage procedure for spatio-temporal leaf segmentation with regard to the b-spline-based phenotyping of cucumber plants. Remote Sensing 2021;13(1). doi:10.3390/rs13010074.

[22] Vázquez-Arellano, M, Reiser, D, Paraforos, D, Garrido-Izard, M, Griepentrog, H. Leaf area estimation of reconstructed maize plants using a time-of-flight camera based on different scan directions. Robotics 2018;7(4):63. doi:10.3390/robotics7040063.

[23] Dupuis, J, Holst, C, Kuhlmann, H. Laser Scanning Based Growth Analysis of Plants as a New Challenge for Deformation Monitoring. Journal of Applied Geodesy 2016;10(1):37–44. doi:10.1515/jag-2015-0028.

[24] Edelsbrunner, H, Mücke, EP. Three-dimensional alpha shapes. ACM

[1] https://gitlab.unistra.fr/plant-leaf-area-estimation-gvc-2022/

Trans Graph 1994;13(1):43–72. doi:10.1145/174462.156635.
[25] Mörwald, T, Balzer, J, Vincze, M. Modeling connected regions in arbitrary planar point clouds by robust b-spline approximation. Robot Auton Syst 2016;76(C):141–151. doi:10.1016/j.robot.2015.11.006.
[26] Åkerblom, M, Raumonen, P, Kaasalainen, M, Casella, E. Analysis of geometric primitives in quantitative structure models of tree stems. Remote Sensing 2015;7(4):4581–4603. doi:10.3390/rs70404581.
[27] Berger, M, Tagliasacchi, A, Seversky, LM, Alliez, P, Guennebaud, G, Levine, JA, et al. A survey of surface reconstruction from point clouds. Comput Graph Forum 2017;36(1):301–329. doi:10.1111/cgf.12802.
[28] Strauss, S, Lempe, J, Prusinkiewicz, P, Tsiantis, M, Smith, RS. Phyllotaxis: is the golden angle optimal for light capture? New Phytologist 2020;225(1):499–510.
[29] Coussement, J, Steppe, K, Lootens, P, Roldán-Ruiz, I, De Swaef, T. A flexible geometric model for leaf shape descriptions with high accuracy. Silva Fennica 2018;52(2).
[30] Li, M, An, H, Angelovici, R, Bagaza, C, Batushansky, A, Clark, L, et al. Topological data analysis as a morphometric method: using persistent homology to demarcate a leaf morphospace. Frontiers in plant science 2018;9:553.
[31] Schönberger, JL, Zheng, E, Pollefeys, M, Frahm, JM. Pixelwise view selection for unstructured multi-view stereo. In: European Conference on Computer Vision (ECCV). 2016,.
[32] Quan, L, Tan, P, Zeng, G, Yuan, L, Wang, J, Kang, SB. Image-based plant modeling. In: ACM SIGGRAPH 2006 Papers. SIGGRAPH '06; New York, NY, USA: Association for Computing Machinery. ISBN 1595933646; 2006, p. 599–604. doi:10.1145/1179352.1141929.
[33] Cloudcompare (version 2.11.0). http://www.cloudcompare.org/; 2020.
[34] Cignoni, P, Callieri, M, Corsini, M, Dellepiane, M, Ganovelli, F, Ranzuglia, G. MeshLab: an Open-Source Mesh Processing Tool. In: Scarano, V, Chiara, RD, Erra, U, editors. Eurographics Italian Chapter Conference. The Eurographics Association. ISBN 978-3-905673-68-5; 2008,doi:10.2312/LocalChapterEvents/ItalChap/ItalianChapConf2008/129-136; http://www.meshlab.net/.
[35] Zhou, QY, Park, J, Koltun, V. Open3D: A modern library for 3D data processing. arXiv:180109847 2018;Http://www.open3d.org/.
[36] Bernardini, F, Mittleman, J, Rushmeier, H, Silva, C, Taubin, G. The ball-pivoting algorithm for surface reconstruction. IEEE Transactions on Visualization and Computer Graphics 1999;5(4):349–359. doi:10.1109/2945.817351.
[37] Marton, ZC, Rusu, RB, Beetz, M. On fast surface reconstruction methods for large and noisy point clouds. In: Proceedings of the 2009 IEEE International Conference on Robotics and Automation. ICRA'09; IEEE Press. ISBN 9781424427888; 2009, p. 2829–2834.
[38] Rusu, RB, Cousins, S. 3D is here: Point Cloud Library (PCL). In: IEEE International Conference on Robotics and Automation (ICRA). 2011,Http://pointclouds.org/.
[39] Wang, X, Li, L, Chai, W. Geometric modeling of broad-leaf plants leaf based on b-spline. Mathematical and Computer Modelling 2013;58(3):564 – 572. doi:10.1016/j.mcm.2011.10.064; computer and Computing Technologies in Agriculture 2011 and Computer and Computing Technologies in Agriculture 2012.
[40] Yin, K, Huang, H, Long, P, Gaissinski, A, Gong, M, Sharf, A. Full 3d plant reconstruction via intrusive acquisition. Comput Graph Forum 2016;35(1):272–284. doi:10.1111/cgf.12724.
[41] Wen, W, Li, B, Li, Bj, Guo, X. A leaf modeling and multi-scale remeshing method for visual computation via hierarchical parametric vein and margin representation. Frontiers in Plant Science 2018;9:783. doi:10.3389/fpls.2018.00783.
[42] Hoppe, H, DeRose, T, Duchamp, T, McDonald, J, Stuetzle, W. Surface reconstruction from unorganized points. SIGGRAPH Comput Graph 1992;26(2):71–78. doi:10.1145/142920.134011.
[43] Kazhdan, M, Bolitho, M, Hoppe, H. Poisson surface reconstruction. In: Proceedings of the Fourth Eurographics Symposium on Geometry Processing. SGP '06; Goslar, DEU: Eurographics Association. ISBN 3905673363; 2006, p. 61–70.
[44] Kazhdan, M, Hoppe, H. Screened poisson surface reconstruction. ACM Trans Graph 2013;32(3):29:1–29:13. doi:10.1145/2487228.2487237.
[45] Morel, J, Bac, A, Véga, C. Surface reconstruction of incomplete datasets: A novel poisson surface approach based on csrbf. Computers and Graphics 2018;74:44 – 55. doi:10.1016/j.cag.2018.05.004.
[46] Rusu, RB, Blodow, N, Marton, Z, Soos, A, Beetz, M. Towards 3d object maps for autonomous household robots. In: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems. 2007, p. 3191–3198. doi:10.1109/IROS.2007.4399309.
[47] Alexa, M, Behr, J, Cohen-Or, D, Fleishman, S, Levin, D, Silva, CT. Point set surfaces. In: Proceedings of the Conference on Visualization '01. VIS '01; Washington, DC, USA: IEEE Computer Society. ISBN 0-7803-7200-X; 2001, p. 21–28.
[48] Guennebaud, G, Gross, M. Algebraic point set surfaces. In: ACM siggraph 2007 papers. 2007, p. 23–es.
[49] Cohen-Steiner, D, Alliez, P, Desbrun, M. Variational shape approximation. ACM Trans Graph 2004;23(3):905–914. doi:10.1145/1015706.1015817.
[50] Ando, R, Ozasa, Y, Guo, W. Robust surface reconstruction of plant leaves from 3d point clouds. Plant Phenomics 2021;2021.

Appendix A. Synthetic leaf model: perimeter, area and convex hull formulas

In this section we express the perimeter $\mathcal{P}$, area $\mathcal{A}$ and convex hull area $\mathcal{CH}$ of our synthetic leaf model with respect to our leaf shape criteria, namely the **length/width ratio R** and the **concavity level Co**. In this study we restrict to flat leaves, that is to say the **curviness Cu** is always zero. We start by deriving the perimeter, area and convex hull in the simplest case, and then progressively extend our approach.

Appendix A.1. Simplest case: **R** = 1 *and* **Co** = 0

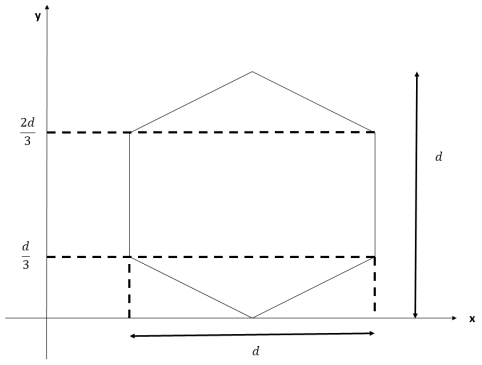


Fig. A.11. Geometry of our synthetic leaf model in case R = 1 and Co = 0.

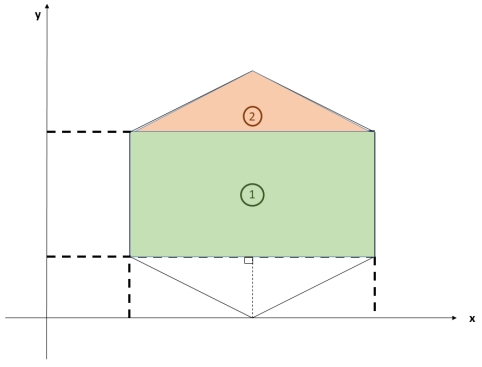


Fig. A.12. Geometry of our synthetic leaf model in case of R = 1 and Co = 0, with areas $\mathcal{A}_1$ and $\mathcal{A}_2$ depicted in green and orange respectively.

By definition of our model $\mathcal{A} = 1$. The underlying hexagon shape relies on a length d as shown on Fig. A.11. Its value can easily be computed since the area $\mathcal{A}$ is the sum of a rectangle area $\mathcal{A}_1$ and two triangle areas $\mathcal{A}_2$ (see Fig. A.12):

$$\mathcal{A} = \frac{d^2}{3} + 2\frac{d}{2}\frac{d}{3} \Leftrightarrow \frac{2d^2}{3} = 1 \Leftrightarrow d = \sqrt{\frac{3}{2}} \tag{A.1}$$

The perimeter $\mathcal{P}$ of the hexagon is equal to:

$$\mathcal{P} = 2(\frac{d}{3} + 2\sqrt{\frac{d^2}{3^2} + \frac{d^2}{2^2}}) = \frac{2d}{3} + \frac{4}{6}\sqrt{13}d = (1 + \sqrt{13})\sqrt{\frac{2}{3}} \tag{A.2}$$

Finally, since the hexagon is convex, the convex hull area $\mathcal{CH}$ is equal to $\mathcal{A} = 1$.

*Appendix A.2. Convex case: **R** variable and* **Co** $= 0$

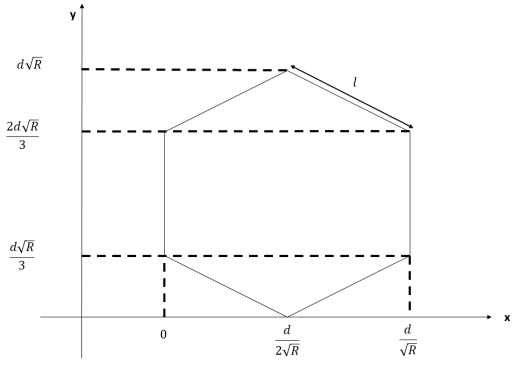


Fig. A.13. Geometry of our synthetic leaf model in case Co $= 0$.

This case is depicted on Fig. A.13. The hexagon area, which is by definition equal to 1, can be expressed as:

$$\mathcal{A} = 2(\frac{d\sqrt{R}}{3})\frac{d}{\sqrt{R}} = 2\frac{d^2}{3} = 1 \tag{A.3}$$

which lead to the same value for d as in the previous simplest case.

Again, the shape being convex we have $\mathcal{CH} = 1$.

The hexagon perimeter $\mathcal{P}$ can be expressed as:

$$\mathcal{P} = 2\frac{d}{3}\sqrt{R} + 4l \tag{A.4}$$

with $l^2 = (\frac{d}{3}\sqrt{R})^2 + (\frac{d}{2\sqrt{R}})^2 = \frac{R}{6} + \frac{3}{8R}$, which leads to:

$$\mathcal{P} = \sqrt{\frac{2R}{3}} + \sqrt{\frac{8R}{3} + \frac{6}{R}} \tag{A.5}$$

*Appendix A.3. Simplest concave case: **R** variable and* **Co** $= 1$

When **Co** $= 1$ the basic shape is not a hexagon anymore, but a non-convex polygon with 18 edges (an octadecagon), see Fig. A.14. This is because each edge of the initial hexagon is split into three edges, as shown on Fig. A.15. Let us note $P_1P_2^{\perp}$

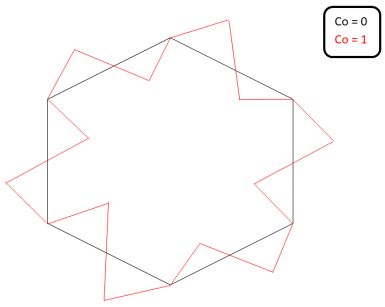


Fig. A.14. Geometry of our synthetic leaf model in case Co $= 1$.

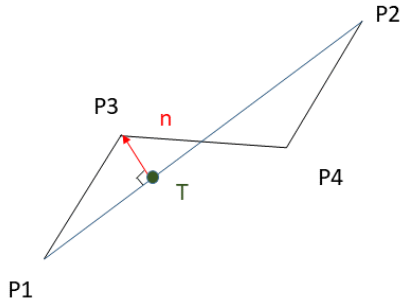


Fig. A.15. Each edge P_1P_2 is split in three edges P_1P_3, P_3P_4 and P_4P_2 when Co is increased.

the orthogonal vector to P_1P_2 and $n = \frac{1}{9}P_1P_2^{\perp}$. Then P_3 and P_4 are defined such that $P_1P_3 = \frac{1}{3}P_1P_2 + n$ and $P_1P_4 = \frac{2}{3}P_1P_2 - n$. Note that the $\frac{1}{9}$ ratio defining n is arbitrary and can be changed.

When an edge P_1P_2 is split, its length L changes to

$$L' = ||P_1P_3|| + ||P_3P_4|| + ||P_4P_2|| \tag{A.6}$$

We have:

$$||P_1P_3||^2 = ||P_4P_2||^2 = (\frac{L}{3})^2 + (\frac{L}{9})^2 \tag{A.7}$$

and

$$(\frac{||P_3P_4||}{2})^2 = (\frac{L}{9})^2 + (\frac{L}{6})^2 \tag{A.8}$$

which leads to:

$$L' = 2\sqrt{(\frac{L}{3})^2 + (\frac{L}{9})^2} + 2\sqrt{(\frac{L}{9})^2 + (\frac{L}{6})^2} \tag{A.9}$$

$$= \frac{L}{9}(2\sqrt{10} + \sqrt{13}) \tag{A.10}$$

Let us note $\alpha = \frac{1}{9}(2\sqrt{10} + \sqrt{13})$. We have shown that when the edge P_1P_2 is split into three smaller edges, its total length is multiplied by the constant factor α. Consequently, the perimeter of the octadecagon is $\alpha\mathcal{P}$, with $\mathcal{P}$ the perimeter of the hexagon as computed in the previous section.

By construction the area of the octadecagon is the same unit area $\mathcal{A}$ as the hexagon, while the convex hull area $\mathcal{CH}$ is increased by the area of each triangle $P_1P_3P_2$, see Fig. A.16.

For each edge P_1P_2, let us define T such that $P_1T = \frac{1}{3}P_1P_2$ (see Fig. A.15). Then the added area $P_1P_3P_2$ is the sum of the areas of the two triangles P_1TP_3 and TP_3P_2, which can easily be expressed as:

$$P_1TP_3 = \frac{1}{2}||P_1T||||TP_3|| \tag{A.11}$$
$$= \frac{1}{2}\frac{L}{3}\frac{L}{9} \tag{A.12}$$
$$= \frac{L^2}{54} \tag{A.13}$$

and

$$TP_3P_2 = \frac{1}{2}||TP_2||||TP_3|| \tag{A.14}$$
$$= \frac{1}{2}\frac{2L}{3}\frac{L}{9} \tag{A.15}$$
$$= \frac{L^2}{27} \tag{A.16}$$

Along each edge P_1P_2 of length L, the convex hull area is thus increased by $\frac{L^2}{18}$.

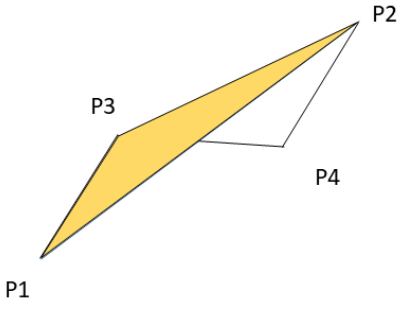


Fig. A.16. Added area to the convex hull (in yellow) in our synthetic leaf model, in case Co = 1.

Noticing that in the convex case the length of an edge of the hexagon is either $\frac{d\sqrt{R}}{3} = \sqrt{\frac{R}{6}}$ (the two vertical edges on Fig. A.13) or $l = \sqrt{\frac{R}{6} + \frac{3}{8R}}$ (the four diagonal edges on Fig. A.13) and remembering that our hexagon has a unit area, we can derive that the convex hull area $\mathcal{CH}$ of the octadecagon is:

$$\mathcal{CH} = 1 + \frac{2}{18}\frac{R}{6} + \frac{4}{18}(\frac{R}{6} + \frac{3}{8R}) \tag{A.17}$$
$$= 1 + \frac{R}{18} + \frac{1}{12R} \tag{A.18}$$

Appendix A.4. General case: induction study

We now derive values for $\mathcal{P}$ and $\mathcal{CH}$ in the general case ($\mathcal{A}$ is always equal to 1, by construction). This is done by induction on **Co**.

Let $\mathcal{P}_n$ and $\mathcal{CH}_n$ be the perimeter and convex hull area values for **Co** $= n$. We know from the previous sections that:

$$\mathcal{P}_0 = \frac{2R}{3} + \sqrt{\frac{8R}{3} + \frac{6}{R}} \tag{A.19}$$
$$\mathcal{P}_1 = \alpha\mathcal{P}_0 \tag{A.20}$$
$$\mathcal{CH}_0 = 1 \tag{A.21}$$
$$\mathcal{CH}_1 = 1 + \frac{R}{18} + \frac{1}{12R} \tag{A.22}$$

Generalizing the study done in section Appendix A.3, it is easy to see that when **Co** is increased by one then each edge is split in three and the associated length is multiplied by a constant factor $\alpha = \frac{1}{9}(2\sqrt{10} + \sqrt{13})$. Hence:

$$\forall n \geq 1, \mathcal{P}_n = \alpha\mathcal{P}_{n-1} = \alpha^n\mathcal{P}_0 \tag{A.23}$$

Now let P_1P_2 be one edge of the initial hexagon. When **Co** $= n, n \geq 1$, this edge is split into 3^n edges (see an example on Fig. A.17 for **Co** $= 2$). Let us note $P_1P_{3^{n-1}+2}$ the first of these edges.

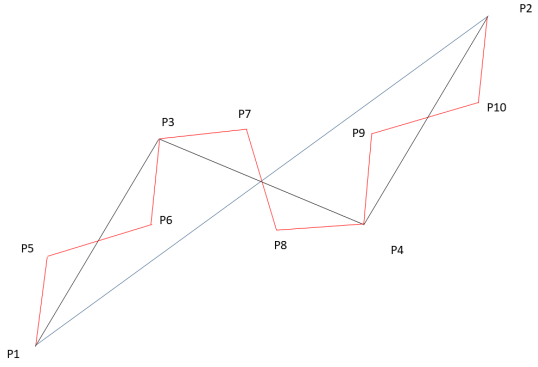


Fig. A.17. Split of an edge P_1P_2 into 9 edges when Co = 2.

We have seen in the previous section that when **Co** $= 1$, only the triangle $P_1P_3P_2$ is added to the convex hull of the shape. Similarly when **Co** $= 2$, only the triangle $P_1P_5P_3$ is added to the convex hull (see Fig. A.18). More generally, when **Co** $= n$, only the triangle $P_1P_{3^{n-1}+2}P_{3^{n-2}+2}$ is added to the convex hull.

Using the same reasoning as in the previous section, the area of such a triangle is $\frac{L_{n-1}^2}{18}$, with L_{n-1} the length of edge $P_1P_{3^{n-2}+2}$. L_{n-1} can be computed recursively thanks to Eq. (A.7):

$$L_0 = ||P_1P_2|| = L \tag{A.24}$$
$$L_1 = ||P_1P_3|| = \frac{\sqrt{10}}{9}L_0 = \frac{\sqrt{10}}{9}L \tag{A.25}$$
$$L_2 = ||P_1P_5|| = \frac{\sqrt{10}}{9}L_1 = (\frac{\sqrt{10}}{9})^2L \tag{A.26}$$
$$\dots \tag{A.27}$$
$$L_{n-1} = ||P_1P_{3^{n-2}+2}|| = \frac{\sqrt{10}}{9}L_{n-2} = (\frac{\sqrt{10}}{9})^{n-1}L \tag{A.28}$$

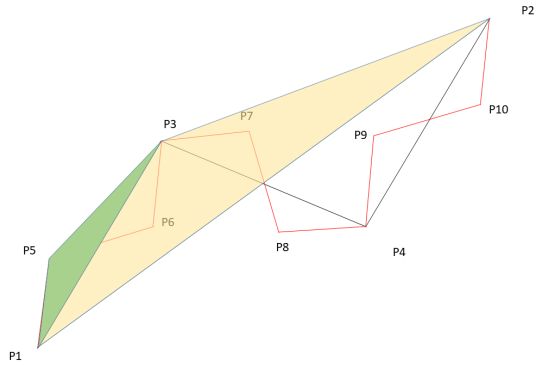


Fig. A.18. Added area in to the convex hull (in green) in synthetic leaf model, in case Co = 2.

Consequently, for each edge of length L of the original hexagon, increasing **Co** from $n-1$ to n, $n \geq 1$, locally increases the convex hull area by a factor of $\frac{L_{n-1}^2}{18} = (\frac{10}{81})^{n-1}\frac{L^2}{18}$. Since two of these edges have length $\sqrt{\frac{R}{6}}$ and the four other ones have length $\sqrt{\frac{R}{6}+\frac{3}{8R}}$ (see above), the convex hull area can be recursively computed as:

$$\mathcal{CH}_0 = 1 \tag{A.29}$$

$$\forall n \geq 1, \mathcal{CH}_n = \mathcal{CH}_{n-1} + (\frac{10}{81})^{n-1}\frac{1}{18}(2\frac{R}{6} + 4(\frac{R}{6}+\frac{3}{8R})) \tag{A.30}$$

$$= \mathcal{CH}_{n-1} + (\frac{10}{81})^{n-1}\frac{1}{18}(R+\frac{3}{2R}) \tag{A.31}$$

$$= \mathcal{CH}_0 + \frac{1}{18}(R+\frac{3}{2R})\sum_{i=0}^{n-1}(\frac{10}{81})^i \tag{A.32}$$

$$= 1 + \frac{9}{142}(R+\frac{3}{2R})(1-(\frac{10}{81})^n) \tag{A.33}$$