Petiole Removal from Plant Leaf Binary Raster Images

Oleg Seredin[1[0000-0003-0410-7705]], Nikita Mityugov[1[0000-0001-7319-7068]], and Kharlampiy Tiras[2,3,4[0000-0001-9597-6994]]

[1] Tula State University, Tula, Russia
nikita.mityugov.2001@mail.ru
[2] Institute of Theoretical and Experimental Biophysics, Pushchino, Russia
[3] PushchGENI – Branch of BIOTECH University, Pushchino, Russia
[4] National Research Centre "Kurchatov Institute", Moscow, Russia

Abstract. This paper proposes an algorithm for automatic petiole removal from binary raster plant leaf images. The problem is addressed in the context of leaf morphometric analysis, used both for plant species identification and ecological monitoring. Accurate petiole extraction serves a dual purpose: on the one hand, it enables correct description of the leaf blade shape; on the other, it opens the possibility for independent morphometric analysis of the petiole itself, whose geometric characteristics are diagnostically informative in their own right. The core idea is to locate a narrow constriction ("bottleneck") on the binary object contour – the point where the petiole meets the leaf blade. For each pair of contour points, an optimization criterion combining geometric properties of the constriction is computed, and the pair with the highest value is selected. In ambiguous cases, an additional SVM-based classification step is applied. Since the search operates directly on the object contour without any assumptions about its orientation, the algorithm is invariant to leaf rotation. The algorithm was evaluated on the Swedish leaf dataset comprising 15 species with 75 images each. For the majority of species, the algorithm achieved zero errors with an average Jaccard measure of 0.952. A comparison with eight modern neural generative models further confirmed the advantages of the proposed deterministic approach: the algorithm produces stable, reproducible, and strictly binary results with processing times ranging from tens to hundreds of milliseconds.

Keywords: Image Analysis, Leaf Shape Analysis, Petiole Removal.

1 Introduction

Plant leaf morphology analysis plays an important role in modern digital biology and ecological monitoring. On the one hand, leaves are widely used for plant species identification: review papers in the field of computer vision demonstrate a wide range of classification methods based on leaf blade shape, texture, and color [1, 2], and interest in automated recognition systems is steadily growing. Recently, spectral and color parameters of plant leaves and flowers have been recorded using portable digital spectrophotometers [3]. On the other hand, leaf morphological characteristics naturally vary within a single species depending on growing conditions, a phenomenon known

as intra-species polymorphism [4] – a property that makes leaves useful as bioindicators. In particular, changes in leaf area, shape, and color have been shown to correlate with levels of air, soil, and water pollution [5], making digital morphometric analysis an effective tool for environmental monitoring. Furthermore, leaf blade area is one of the key agronomic parameters: automated tools for its measurement from scanned images enable high-throughput phenotypic analysis in plant breeding and physiological research [6], while transformer-based segmentation of three-dimensional point clouds offers the possibility of accurate leaf area extraction in field conditions during active plant growth [7]. Moreover, leaf shape variation is widely used in taxonomy and evolutionary studies: combining geometric morphometrics with deep learning enables reliable discrimination of closely related species and hybrids even under high intraspecific variability [8, 9].

This paper deals with scanned fresh leaf images. Working with scans presents specific challenges: since a leaf has a certain thickness, scanning produces a shadow with blurred edges and variable color close to that of the plant itself. This significantly complicates the extraction of the serrated leaf blade margin, which is critical for morphometric analysis, and limits the applicability of standard segmentation methods. A solution to the segmentation problem for scanned leaves in the presence of a colored shadow was proposed in [10, 11]. After segmentation, the shape of the extracted region is described using a shape descriptor known as the rotation profile [12] – a specific mathematical representation that provides accurate and compact extraction of geometric information, invariant to scaling, translation, and rotation. However, the accuracy of the shape descriptor largely depends on the quality of the input data. A significant source of distortion is the leaf petiole: being a morphologically distinct structure from the blade, it introduces asymmetric and atypical geometric components into the descriptor, which reduces classification accuracy. Figure 1 shows a comparison of rotation profiles for the same leaf with and without the petiole.

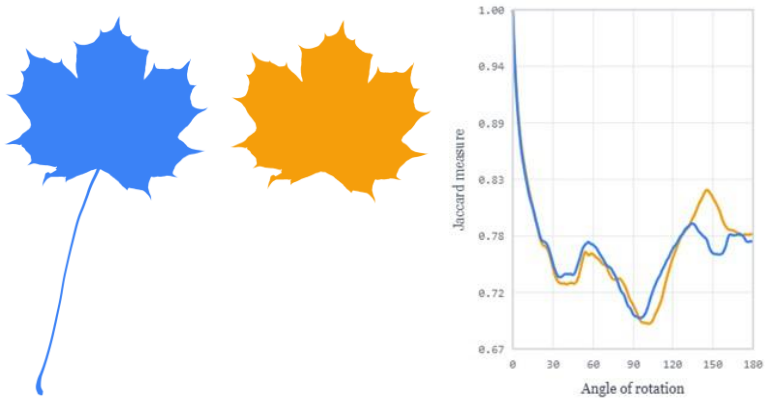

Fig. 1. Comparison of rotation profiles of the same maple leaf image (blue line – leaf with petiole, orange line – leaf without petiole)

At the same time, the leaf petiole is itself a morphologically informative structure worthy of independent study. Its geometric characteristics – length, thickness, and shape – vary depending on the plant species, growing conditions, and physiological state. In particular, it has been shown that petiole length is significantly reduced in plants growing under exhaust gas pollution conditions [13]. Changes in petiole shape and size are also observed under abiotic stressors – drought, insufficient lighting [14]. More broadly, disease development causes measurable morphological changes in leaf tissue [15], which further motivates the use of digital image analysis for plant health assessment. Anatomical features of the petiole are used as taxonomic markers for distinguishing closely related species [16]. Thus, accurate petiole extraction opens the possibility for its independent morphometric analysis, further increasing the value of digital ecological monitoring methods. The problem of automatic petiole extraction has previously been addressed in a number of works. It should also be noted that detecting narrow constrictions ("bottlenecks") on the contour of a binary object is a more general computer vision problem, arising in other application domains as well. Wang et al. [17] proposed a method for splitting clumped cells based on bottleneck detection: for each pair of contour points, the algorithm seeks pairs that minimize the Euclidean distance while simultaneously maximizing the distance along the contour. Similar approaches are used for separating overlapping cells in microscopic images [18] and organ segmentation in medical tomograms [19]. In leaf shape analysis, bottleneck detection directly corresponds to the problem of localizing the petiole attachment point. Mzoughi et al. [20] proposed a method based on local translational symmetry: the petiole is identified as the contour region where the object width remains approximately constant under parallel displacement of the cross-section. The method achieves accuracy above 90% on diverse leaf shapes and is independent of petiole orientation, but requires careful tuning of symmetry thresholds and may fail on leaves with complex attached structures. Wang et al. [21] proposed an algorithm based on a dual-channel pulse coupled neural network (PCNN) model and HSI color space, tested on 169 leaf types. However, the method requires prior orientation of the image – the petiole must be positioned at the bottom with a deviation of no more than 45 degrees – and, as the authors themselves acknowledge, is not applicable to compound leaves. Elen and Avuçlu [22] proposed a different approach: the binary image is projected onto the X and Y axes, and the petiole is detected as a local minimum in the cumulative pixel distribution plots. The method is simple to implement and requires no training data, but is fundamentally dependent on leaf orientation: when the image is arbitrarily rotated, the petiole no longer appears as a minimum in the projections. In some works, the petiole is removed manually prior to further analysis [23], which is impractical for processing large image collections. Thus, existing methods either depend on image orientation or require manual intervention, which significantly limits their applicability for large-scale processing of scanned leaf collections. The present work proposes an automatic petiole removal algorithm free from these limitations: it operates directly on the contour of the binary object, is invariant to image rotation and scale, and requires no assumptions about leaf orientation.

2 Petiole removal algorithm for binary plant leaf images

2.1 Goal function

The algorithm leverages geodesic and Euclidean distances along the object contour to detect bottleneck candidates, which are then classified using a support vector machine (SVM). The core idea is to locate a so-called "bottleneck" on the leaf contour – a narrow constriction where the petiole meets the leaf blade. The leaf petiole is typically an elongated stalk attached to the base of the blade at a point where the object width drops sharply compared to the rest of the leaf. At this location, two boundary segments on opposite sides of the petiole converge, forming a local minimum in the cross-sectional width. Detecting this constriction on the contour allows us to precisely localize the cut line, remove the petiole, and restore the closed leaf blade contour. At the preprocessing stage, the largest outer contour is extracted from the binary image using the findContours function from the OpenCV computer vision library [24], which returns ordered list of boundary pixel coordinates for connected components.

Let $\{p_0, p_1, \ldots, p_{N-1}\}$ denote the segmentation object contour, where p_i is the contour point at index i; N is the total number of contour points. To remove the petiole from a binary raster image, we need to find a pair of points on the segmentation object contour along which the petiole will be cut: $(p_{i^*}, p_{j^*}) = \underset{0 \le i < N, i < j < N}{\arg\max}\, T(p_i, p_j)$, where T is an optimization criterion in the range $[0,1]$, consisting of three sub-criteria T_1, T_2, T_3. Let the target points p_{i^*} and p_{j^*} be denoted A and B. We then define:

- $D(A,B)$ – the Euclidean distance between the target pair of points A and B;
- G – the total geodesic length of the contour. The geodesic distance between a pair of contour points is defined as the pixel-by-pixel distance along the contour considering eight-connected neighbours. Let the minimum geodesic distance between points A and B be denoted $G(A,B)$, then the maximum geodesic distance is $G - G(A,B)$;
- $D(C,D)$ – the Euclidean distance between points C and D, representing the petiole width at its midpoint, where C and D are the midpoints of the shortest arcs from A to S and from S to B respectively, and S is the midpoint of the shortest arc from A to B.

Figure 2 illustrates the notation introduced above.

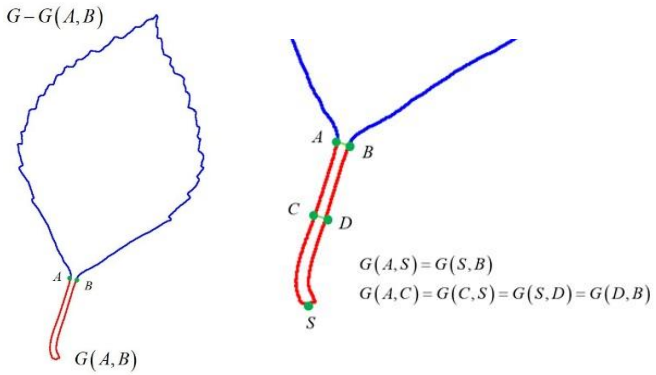

Fig. 2. Schematic illustration of the notation used for computing criterion T

Sub-criterion T_1 evaluates the constriction width relative to the petiole length and is computed as:

$$T_1 = 1 - \frac{\min\left[\varphi\left(D(A,B) + 2\cdot\max\left(D(A,B) - D(C,D), 0\right), De\right), G(A,B)\right]}{G(A,B)}, \qquad (1)$$

where $\varphi(D, De) = \alpha 1.15^{D-\max[0,10\cdot(1.1\cdot De-D)]} + (1-\alpha)\log_{1.5}(D)$ is a Euclidean distance correction function (Fig. 3) that amplifies the contrast between narrow and wide constrictions;

De – the estimated Euclidean distance between points A and B, (see Sect. 2.1);

α – a weight coefficient controlling the balance between the two terms of $\varphi(D, De)$: the exponential and the logarithmic. In this paper $\alpha = 0.3$.

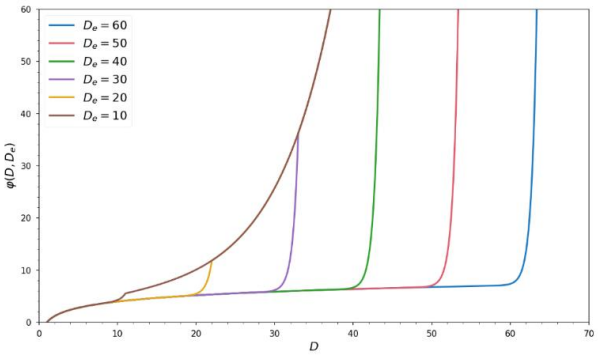

Fig. 3. Plots of $\varphi(D, De)$ for different values of De with $\alpha = 0.3$

A high value of T_1 corresponds to a narrow petiole relative to its length. Figure 4 compares $\varphi(D, De)$ values for a petiole and for an arbitrary constriction on the leaf.

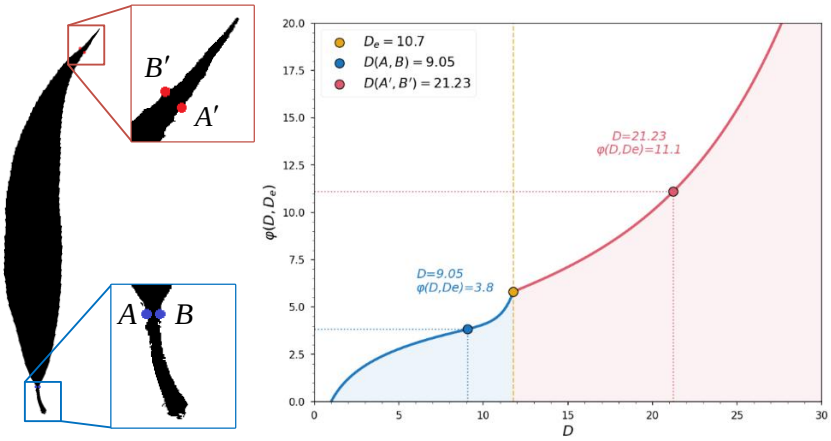

Fig. 4. Comparison of $\varphi(D, De)$ for the petiole (blue points A, B) and an arbitrary constriction on the leaf (red points A', B')

The correction function value for the competing constriction $D(A', B')$ is significantly larger than for $D(A, B)$, so sub-criterion T_1 will penalize the pair A' and B' more heavily by $\varphi(D, De)$, reducing its chance of being selected as the target points.

It is worth noting that the term $2 \cdot \max(D(A, B) - D(C, D), 0)$ in the numerator penalizes narrowing appendages while leaving widening ones unpenalized. This prevents the leaf tip from being erroneously identified as a petiole.

Sub-criterion T_2 evaluates the uniformity of the petiole width and its linearity. Let P_{AB} be the midpoint of segment AB; P_{CD} – the midpoint of segment CD (Figure 5), then:

$$T_2 = \frac{2 \cdot \left(D(P_{AB}, P_{CD}) + D(P_{CD}, S) \right)}{G(A, B) + D(A, B) + \varphi\left(\left| D(A, B) - D(C, D) \right| \right)}. \quad (2)$$

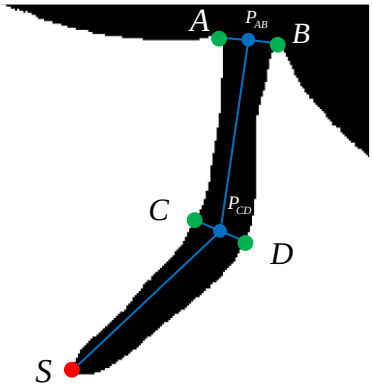

Fig. 5. Illustrations for sub-criterion T_2

The term $D(A,B)$ penalizes excessively wide petioles, while $\varphi\left(\left|D(A,B)-D(C,D)\right|\right)$ penalizes petioles that widen too much towards the middle. This avoids false positives on certain leaf shapes, such as rowan.

Sub-criterion T_3 characterizes the proportion of the contour occupied by the petiole, i.e., its relative length:

$$T_3 = \frac{G(A,B)}{\left(G-G(A,B)\right)+G(A,B)+D(A,B)} = \frac{G(A,B)}{G+D(A,B)}. \quad (3)$$

This sub-criterion favors longer petioles that occupy a larger share of the contour: the larger T_3, the more likely the detected constriction corresponds to the petiole base. The term $D(A,B)$, as in sub-criterion T_2, penalizes target points that are too far apart.

Thus, each of the three sub-criteria captures a distinct geometric aspect of the constriction: T_1 evaluates its width relative to length, T_2 – linearity and uniformity, T_3 – the proportion of the contour occupied by the petiole. To obtain a single criterion T, the sub-criteria are combined into a linear combination with empirically selected weights. Since T_1 and T_2 lie in the range $[0,1]$ while T_3 lies in $[0,0.5]$, the latter is additionally multiplied by 2. The final formula for T is:

$$T = 0.35\cdot T_1 + 0.2\cdot T_2 + 0.45\cdot 2\cdot T_3. \quad (4)$$

Among all valid candidate pairs, the one with the highest criterion T is retained. Once such a pair is found, the petiole is removed: the object contour is cut along segment AB, and the shorter arc – corresponding to the petiole – is discarded. The cut is closed by a straight segment between A and B, restoring the leaf blade contour. The algorithm thus splits the binary image into two independent regions – the leaf blade

and the petiole – enabling independent morphometric analysis of either structure depending on the task at hand.

2.2 Rough estimation of petiole width *De*

Since images may have different resolutions, petiole widths vary accordingly. We therefore perform a preliminary search to estimate the characteristic petiole width De, used in the computation of sub-criterion T_1. For each pair of points A, B, the Euclidean distance $D(A,B)$ is computed over a fixed set of ratios $r = 100 \cdot G(A,B) \ / \ G \in \{[2],[3],[4],[5]\}$, where $[\ ]$ denotes rounding to the nearest integer. For each integer ratio r, an independent candidate list is maintained: among all pairs (A,B) that pass all constraints (see the corresponding subsection), the pair with the smallest Euclidean distance $D(A,B)$ is stored.

After the full traversal, at most four candidates are retained – one per value of r. The final value of De is taken from the candidate with the smallest r for which at least one valid pair was found. This strategy is necessary for two reasons. First, point pairs are traversed counter-clockwise along the contour: for each point A, the first valid point B with the smallest $D(A,B)$ is recorded. This means that if a candidate with a certain $D(A,B)$ has already been stored for the 5% threshold, then when searching for a candidate at the 2% threshold, the distance $D(A,B)$ will be larger – so the 2% threshold cannot improve the stored minimum if all thresholds are processed in a single pass. Maintaining separate candidate lists for each r allows independent tracking of the best candidate within each range. Second, for some images the minimum Euclidean distance $D(A,B)$ at small geodesic ratios (2–3%) cannot be found at all: candidate pairs fail one or more constraints. In such cases the stored value for that r remains empty, and the algorithm automatically falls back to the next wider range.

The priority of smaller r values when selecting the final De is driven by the petiole's shape: at $r = 5\%$, the pair A and B falls roughly at the middle of the petiole, where it narrows, yielding an underestimated De. At $r = 2-3\%$, the pair is located near the petiole tip, where the width is more representative and stable, giving a more accurate estimate of De.

The estimated value De is subsequently used as a parameter of the function $\varphi(D, De)$, representing the typical petiole width for the given leaf. During the same pass, $D_{\max}$ is also computed – the maximum Euclidean distance between any pair of contour points, with no constraints applied.

2.3 Decision making

The main traversal selects the pair of points with the highest criterion T, but in some cases the detected pair may correspond to the leaf tip rather than the petiole. This occurs for two reasons. First, some images contain no petiole at all – the leaf was detached before scanning – and the algorithm may then erroneously identify a geometrically similar leaf tip as a petiole. Second, even when a petiole is present, its geometry can be insufficiently pronounced in certain species: a short or wide petiole closely resembles a leaf tip, making it difficult to reach a reliable decision based on T alone.

To resolve these ambiguities, an SVM classifier (Fig. 6) is applied, trained on species with geometrically indistinct petioles. The training set includes leaf samples from the class Salix aurita and several specimens of Ulmus carpinifolia – species in which the petiole is short and geometrically close to the leaf tip (a detailed description of the leaf dataset is given in the experimental section). The final training set comprised 53 petiole samples and 40 tip samples. The classifier takes a feature vector (T_1, T_2, T_3) as input and reliably discriminates ambiguous cases that criterion T alone cannot resolve.

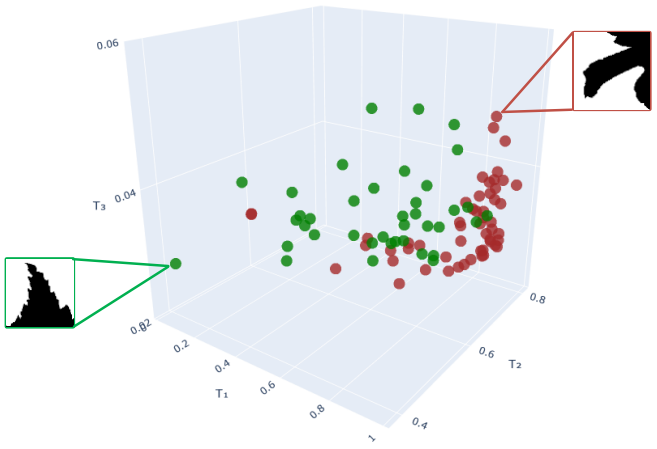

Fig. 6. Feature space of sub-criteria T_1, T_2, T_3 used to train the SVM. Brown points correspond to the positive class (petiole) and green points to the negative class (tip)

As seen in Figure 6, sub-criterion T_1 is the best class separator: the two classes are most clearly separated along the T_1 axis, whereas along T_2 and T_3 the class regions overlap considerably. The decision rule described below takes into account the following observation: in uncertain cases – for example when T falls in the range $0.3 \le T \le 0.65$ – the algorithm may erroneously select a leaf tip as the final result. In such cases, a competing candidate is searched for on approximately the opposite side

of the contour, and the choice between the two candidates is then made using the pre-trained SVM classifier.

The petiole removal decision follows these rules: if $T \geq 0.65$, the result is accepted as confident without consulting the SVM. If $T \leq 0.3$, the leaf is deemed to have no petiole. In the range $0.3 \leq T \leq 0.65$, as described above, a competing candidate with target points A', B' is searched for near point S of the first candidate, with tolerance $\left(D(S,S') - D_{\max}\right) \leq 0.05 \cdot D_{\max}$, where S' is the midpoint of the geodesic arc between the target points A' and B' of the second candidate. If the second candidate is found and $\left(T - T'\right) \leq 0.1$, both candidates are passed to the SVM. The final selection follows these rules: if the predictions differ, the candidate classified as "petiole" is accepted; if both are classified as "petiole", the one with the greater distance to the decision boundary is selected; if both are rejected, no petiole is detected.

If the second candidate is not found, or the condition $\left(T - T'\right) \leq 0.1$ is not satisfied, only the first candidate is passed to the SVM.

2.4 Constraints and search range narrowing

We introduce the following constraints on the search for points A and B (Figure 7):

1. assuming black pixels represent the object and white pixels the background, there must be no white pixels between the target points A and B; for efficiency, only 3 equidistant points along the segment are checked;
2. $D(A,B) + D(C,D) > 20$ – the sum of Euclidean distances between points A, B and C, D must exceed 20 pixels to filter out noise;
3. $0.03 < \dfrac{G(A,B)}{\left(G - G(A,B)\right) + G(A,B)} < 0.45$ – the search is bounded by geodesic distance bounds between points A and B. The lower bound excludes adjacent contour points that produce noise, while the upper bound confines the search to the smallest portion of the contour corresponding to the petiole.
4. verification that point Z lies outside the object. A control point is computed as $Z = S + (S - A) \cdot 0.05$, placing Z slightly beyond S in the direction away from A, i.e., just outside the object relative to S. If the pixel at Z is black (belongs to the object), the pair is discarded. This constraint complements constraint 1): when cuts or tears are present on the leaf, the three check pixels along segment AB may accidentally land on black object pixels and miss the violation, whereas point Z, located beyond S outside the object, catches such false pairs.
5. $D(A,B) < \max\left(D(A,S), D(B,S)\right)$ – given that points A, B, and S form a triangle $\triangle ABS$, side AB must not be the longest, ensuring that angle $\angle ASB$ is non-degenerate. This eliminates noise and excludes small wide petioles.

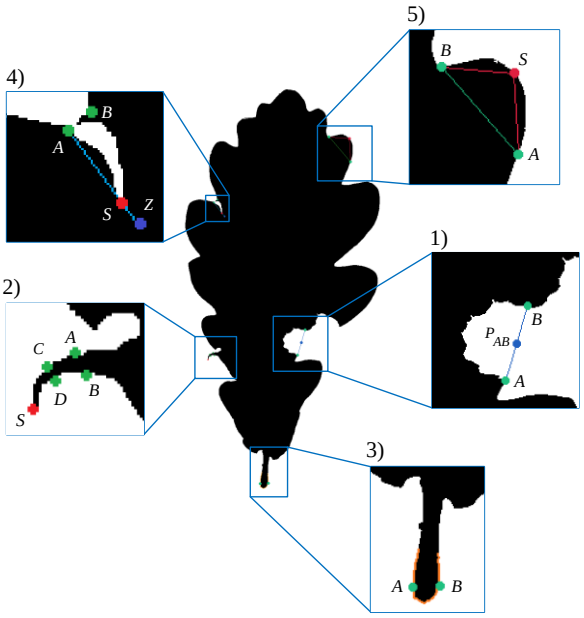

Fig. 7. Visual representation of the constraints on point pairs A and B: examples of pairs discarded by selected algorithm filters

3 Experiments

We evaluate the proposed petiole removal algorithm on the Swedish leaf dataset [25] (Table 1), which contains 15 species (classes) of plant leaf images with 75 images per class (1125 images in total). Since the Swedish leaf dataset contains color images while the algorithm operates on binary inputs, all images had to be segmented and binarized. For this purpose, we applied the one-class color segmentation method based on a probabilistic gamma-normal model described in [10, 11]. However, this method does not account for holes within objects and fills them in. Therefore, for two plant classes (Sorbus aucuparia and Sorbus intermedia), the Background Erase Network (BEN) segmentation model [26] was used instead.

To evaluate petiole removal quality, the algorithm's output was compared against expert annotations. For each binary image, an expert manually marked a pair of points on the petiole to form the ground truth mask. Quality was assessed using the Jaccard measure [27], computed as the ratio of the intersection area of the predicted petiole mask and the ground truth mask to the area of their union.

To quantitatively evaluate the algorithm across the full dataset, errors were counted. Whether a petiole is present in an image is determined directly from the expert annotation: if the expert marked a pair of points, a petiole is present; otherwise it is not. The algorithm always produces a definitive output: petiole found or not found. If a petiole is present and the algorithm detected it, the result is evaluated by the Jaccard measure: a value below 0.7 is counted as an error. If a petiole is present but the algo-

rithm failed to detect it, this is an error. If no petiole is present but the algorithm found one, this is also an error (the algorithm may have detected a leaf tip instead). Since the algorithm operates within a defined geodesic distance range (see the Constraints subsection), a missed petiole is not counted as an error when the expert points fall outside that range: the algorithm is not designed to handle petioles beyond its working range.

Table 1 presents the algorithm results for all 15 species of the Swedish leaf dataset. For each species, the number of incorrectly processed images out of 75 is reported, along with the average Jaccard measure (Avg Jaccard) and the average processing time per class. The class Ulmus glabra deserves special mention: none of its images contain a petiole within the algorithm's working range – either no petiole is present, or the expert points fall outside the defined geodesic distance range – and therefore the average Jaccard index for this class is not computed.

Table 1. Algorithm results for classes of the Swedish leaf dataset

Acer	Alnus incana	Betula pubescens	Fagus sylvatica	Populus tremula
				
Errors: 0 Avg *J*: 0.984 Avg time: 732 ms	Errors: 0 Avg *J*: 0.97 Avg time: 69 ms	Errors: 0 Avg *J*: 0.937 Avg time: 266 ms	Errors: 0 Avg *J*: 0.978 Avg time: 217 ms	Errors: 0 Avg *J*: 0.997 Avg time: 210 ms
Populus	Quercus	Salix alba Sericea	Salix aurita	Salix cinerea
				
Errors: 0 Avg *J*: 0.971 Avg time: 214 ms	Errors: 1 Avg *J*: 0.906 Avg time: 291 ms	Errors: 0 Avg *J*: 0.966 Avg time: 326 ms	Errors: 8 Avg *J*: 0.869 Avg time: 505 ms	Errors: 0 Avg *J*: 0.943 Avg time: 217 ms
Sorbus aucuparia	Sorbus intermedia	Tilia	Ulmus carpinifolia	Ulmus glabra
				
Errors: 0 Avg *J*: 0.973 Avg time: 1492 ms	Errors: 0 Avg *J*: 0.982 Avg time: 909 ms	Errors: 0 Avg *J*: 0.995 Avg time: 189 ms	Errors: 21 Avg *J*: 0.862 Avg time: 472 ms	Errors: 20 Avg *J*: *0.0** Avg time: 563 ms

*–see text for details.

As part of the experimental study, the proposed algorithm was additionally compared with a number of modern neural generative models available through the fal.ai aggregator. The following models were tested: flux-2-flex, flux-2-klein-4b, flux-2-klein-9b, gpt-image-1.5, grok-imagine-image, nano-banana-pro, qwen-image-2511, qwen-image-to-image. All models were given the same prompt: remove the petiole from a binary leaf image while preserving the strictly binary format (black silhouette on white background), the image dimensions and position, and the shape of the leaf blade; if no petiole is present, return the original image unchanged.

Our experiments revealed a number of systematic shortcomings common to all tested neural network models:

1. the output for the same image varied between requests: generative models are non-deterministic, which precludes reproducibility;
2. despite an explicit requirement in the prompt to preserve a strictly binary format, the output images contained gradients and halftones, i.e., were no longer binary;
3. some models changed the output image dimensions relative to the input image;
4. in addition to removing the petiole, the models introduced undesirable changes to the leaf blade shape: smoothing the serrated margin, adding or removing contour details;
5. model processing times are prohibitively long: from 3 to 25 or more seconds per image.

These shortcomings make generative neural network models unsuitable for this task in its current formulation, whereas the proposed deterministic algorithm is free from all of the above issues and delivers a stable, reproducible, and strictly binary result.

4 Conclusion

This paper proposes an algorithm for automatic petiole removal from binary plant leaf images. Since the search for cut points is conducted directly on the object contour without any assumptions about its orientation, the algorithm is invariant to leaf rotation. The problem is addressed in the context of leaf morphology analysis, used both for species identification and ecological monitoring. Beyond its role as a preprocessing step before computing the rotation profile-based shape descriptor [12], accurate petiole extraction opens the possibility for its independent morphometric analysis: petiole length, thickness, and shape are informative bioindicators responding to environmental pollution and stress factors, and serving as taxonomic markers for distinguishing closely related species. The algorithm is based on detecting a narrow constriction ("bottleneck") on the binary object contour and comprises three sequential passes: preliminary estimation of the characteristic petiole width , a main pass over point pairs with computation of sub-criteria T_1, T_2, T_3, and final classification using a support vector machine. To resolve ambiguous cases – leaves without a petiole and species with geometrically indistinct petioles – the SVM classifier was trained on a sample of 93 objects including Salix aurita and Ulmus carpinifolia.

The algorithm was evaluated on the Swedish leaf dataset comprising 15 species with 75 images each. Removal quality was assessed using the Jaccard measure against the expert ground truth. For the majority of species, the algorithm achieved zero errors with an average Jaccard measure of 0.952, demonstrating high accuracy in petiole localization and removal. The highest error counts were observed for Ulmus glabra (21 errors), Ulmus carpinifolia (20 errors), and Salix aurita (8 errors) – species with atypical petiole base geometry.

Comparison with eight modern neural generative models from the fal.ai aggregator revealed fundamental limitations of the neural network approach for this task: non-deterministic outputs, loss of binarization, changes in image dimensions, and undesirable distortions of the leaf contour. The proposed algorithm is free from all of these drawbacks: it is deterministic, invariant to rotation and scale, preserves strict binarization, and introduces no artefacts into the leaf blade shape. The implementation is written in C++ using the OpenCV library [24]; processing times for images from the Swedish leaf dataset ranged from tens to hundreds of milliseconds depending on contour length and complexity.

Funding. This research is funded by the Ministry of Science and Higher Education of the Russian Federation within the framework of the state task FEWG-2024-0001.

References

1. Wäldchen, J., Mäder, P.: Plant species identification using computer vision techniques: a systematic literature review. Arch. Comput. Methods Eng. **25**(2), 507–543 (2018). https://doi.org/10.1007/s11831-016-9206-z
2. Cope, J.S., Corney, D., Clark, J.Y., Remagnino, P., Wilkin, P.: Plant species identification using digital morphometrics: a review. Expert Syst. Appl. **39**(8), 7562–7573 (2012). https://doi.org/10.1016/j.eswa.2012.01.073
3. Tiras, Kh.P., Zelenskaya, N.N., Valkov, Il.N., Tiras, N.R.: Biological colorimetry as a new field of biophotonics. Light & Eng. **32**(3), 30–37 (2024). https://doi.org/10.33383/2023-083
4. Senin, A.N., Tiras, Kh.P., Mestetsky, L.M.: Generating shape features from plant leaf images for bioindication. Ekologiya. Ekonomika. Informatika. **1**(7), 66–75 (2022). https://doi.org/10.23885/2500-395X-2022-1-7-66-75
5. Molnar, V.E., Tozser, D., Szabo, S., Tothmeresz, B., Simon, E.: Use of leaves as bioindicator to assess air pollution based on composite proxy measure (APTI). Plants **9**(12), 1743 (2020). https://doi.org/10.3390/plants9121743
6. Suarez, E., Blaser, M., Sutton, M.: Automating leaf area measurement in citrus: the development and validation of a Python-based tool. Appl. Sci. **15**(17), 9750 (2025). https://doi.org/10.3390/app15179750
7. Peng, X., et al.: Leaf area extraction framework: transformer-based segmentation and geometric-based completion approaches for accurate extraction of field maize leaf area from point clouds. Comput. Electron. Agric. **248**, 111793 (2026).
8. Hodac, L., et al.: Deep learning to capture leaf shape in plant images: validation by geometric morphometrics. Plant J. **120**(4), 1437–1452 (2024). https://doi.org/10.1111/tpj.17053

9. Qi, M., Du, F.K., Guo, F., Yin, K., Tang, J.: Species identification through deep learning and geometrical morphology in oaks (*Quercus* spp.). Ecol. Evol. **14**(2), e11032 (2024). https://doi.org/10.1002/ece3.11032
10. Liakhov, D.V., Mityugov, N.S., Gracheva, I.A., et al.: Scanned plant leaves boundary detection in the presence of a colored shadow. Pattern Recognit. Image Anal. **32**(3), 575–585 (2022). https://doi.org/10.1134/S1054661822030221
11. Lyakhov, D.V., Mityugov, N.S., Gracheva, I.A., et al.: Scanned leaves boundary detection based on the consistent one-class segmentation. In: Proc. SPIE, vol. 12267, p. 1226716. SPIE, Berlin (2022). https://doi.org/10.1117/12.2641160
12. Seredin, O., Lomov, N., Liakhov, D., et al.: Rotation profile-based binary shape descriptor. Vis. Comput. **41**(11), 8911–8933 (2025). https://doi.org/10.1007/s00371-025-03906-9
13. Shakeel, M., et al.: Micro-morphological adaptations and tolerance assessment of selected vascular plant species exposed to vehicular exhausts. Environ. Res. Commun. **5**(7), 075021 (2023). https://doi.org/10.1088/2515-7620/ace612
14. Tadrist, L., et al.: Petiole morphology is adapted to mechanical demands in understory plants. New Phytol. **228**(4), 1434–1447 (2020). https://doi.org/10.1111/nph.16774
15. Barbedo, J.G.A.: An automatic method to detect and measure leaf disease symptoms using digital image processing. Plant Dis. **98**(12), 1709–1716 (2014). https://doi.org/10.1094/PDIS-03-14-0290-RE
16. Karaismailoglu, M.C.: Petiole anatomy of 21 representatives of the tribe Alysseae (Brassicaceae) from Turkey. Nord. J. Bot. **37**(9) (2019). https://doi.org/10.1111/njb.02390
17. Wang, H., Zhang, H., Ray, N.: Clump splitting via bottleneck detection and shape classification. Pattern Recognit. **45**(7), 2780–2787 (2012). https://doi.org/10.1016/j.patcog.2011.12.017
18. Zafari, S., Eerola, T., Sampo, J., Kalviainen, H., Haario, H.: Improving concave point detection to better segment overlapped objects in images. Multimed. Tools Appl. **82**, 33879–33905 (2023). https://doi.org/10.1007/s11042-023-15382-1
19. Cheng, Y., Bai, X.: Efficient liver segmentation in CT images based on graph cuts and bottleneck detection. Comput. Methods Programs Biomed. **140**, 81–91 (2017). https://doi.org/10.1016/j.cmpb.2016.11.014
20. Mzoughi, O., Yahiaoui, I., Verroust-Blondet, A., Berretti, S.: Extraction of leaf parts by image analysis. In: Proceedings of ICIAP, LNCS, vol. 9280. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-319-23234-8_41
21. Wang, Z., Zheng, X., Sun, X., Wang, H., Zhu, Y., Liu, J., Ma, Y.: A new petiole detection algorithm based on leaf image. Proceedings of the IEEE 28th Canadian Conference on Electrical and Computer Engineering, pp. 1430–1434 (2015). https://doi.org/10.1109/CCECE.2015.7129490
22. Elen, A., Avuçlu, E.: Automatic detection of petiole border in plant leaves. Meas. Control **54**(3–4), 446–456 (2021). https://doi.org/10.1177/0020294020917701
23. Coussement, J.R., et al.: A flexible geometric model for leaf shape descriptions with high accuracy. Silva Fenn. **52**(2), 7740 (2018). https://doi.org/10.14214/sf.7740
24. Bradski, G.: The OpenCV library. Dr. Dobb's J. Softw. Tools **25**(11), 120–123 (2000)
25. Söderkvist, O.: Computer vision classification of leaves from Swedish trees. M.Sc. thesis, Linköping University (2001)
26. Meyer, M., Spruyt, J.: BEN: using confidence-guided matting for dichotomous image segmentation. arXiv:2501.06230 (2025)
27. Jaccard, P.: Étude comparative de la distribution florale dans une portion des Alpes et des Jura. Bull. Soc. Vaudoise Sci. Nat. **37**, 547–579 (1901)