



# A ridge-based detection algorithm with filament overlap identification for 2D mycelium network analysis

Oscar Sten<sup>a,b,\*</sup>, Emanuela Del Dottore<sup>a</sup>, Nicola Pugno<sup>b,c</sup>, Barbara Mazzolai<sup>a,\*</sup>

<sup>a</sup> Bioinspired Soft Robotics Laboratory, Istituto Italiano di Tecnologia, Via Morego 30, Genova 16163, Italy

<sup>b</sup> Laboratory for Bioinspired, Bionic, Nano, Meta Materials and Mechanics, Department of Civil, Environmental and Mechanical Engineering, University of Trento, Via Mesiano 77, Trento 38122, Italy

<sup>c</sup> School of Engineering and Materials Science, Queen Mary University of London, Mile End Road, London E1 4NS, United Kingdom

## ARTICLE INFO

### Keywords:

Mycelium  
Network topology  
Image analysis

## ABSTRACT

Many terrestrial ecosystems engage mycorrhizal symbiotic associations, potentially to enhance nutrition, increase resistance to soil-borne pests and diseases, and improve resilience and soil structure. Mycorrhizal fungi create dynamic networked structures through branching and anastomosis that connect multiple plants and consent to transport resources underground from nutrient-rich patches to demanding plants. Controlled laboratory experiments are fundamental to improving our knowledge of mycelium network growth dynamics and further understanding its role in preserving ecological niches. We propose a method for highly automated analysis of the mycelium network structure and other morphological properties, such as hyphal length, hyphal density, and number of crossing and branches, in 2D microscopy images of fungal samples. Available tools for automated network analyses suffer from overestimating network connectivity since filament crossings are not considered. In particular, we propose a) a ridge-based mycelium detection algorithm and b) a geometrical-based approach to identify overlapping filaments crossing each other. The algorithmic solution is evaluated on a total of 135 real mycelium sample images over different validation steps, originating from different datasets and having different characteristics, including background, contrast, image acquisition system, fungal species, and clearness (e.g., level of transparency, homogeneity, dirtiness of the medium) of the sample. Results show that 1) the proposed detection method can be used to measure the length of mycelium in an image, replacing manual tracing and allowing for less laborious analysis ( $\hat{\rho}_c = 0.96$ ), 2) the filament detection is on par with state-of-the-art techniques ( $F_1 = 0.88 - 0.94$ ) with a more intuitive parameterization, and 3) the proposed algorithm correctly identifies filament crossings ( $F_1 = 0.89$ ) in most common cases, yielding a reduction in the overestimation of network connectivity. The latter feature consents to applying the proposed fully automated solution to complex and irregular fungal structures, advancing mycelium detection and reconstruction performance accuracy with respect to the state-of-the-art.

## 1. Introduction

### 1.1. Motivations

Fungal networks consist of filamentous structures called hyphae, which grow from multiple apices or tips of a mycelium assuming topologies that evolve through two processes: a) branching, meaning the creation of a new filament, and b) anastomosis, meaning that an apex merges into an existing filament that is part of the same colony (Dikec et al., 2020). Specifically, Ledoux et al. (2022) consider two types of branching behavior: a) lateral branching, where a mother hypha

continues growing, but it shoots off a lateral branch, and b) apical branching, where the mother hypha splits into a primary branch and an exploratory branch. The creation of cyclic structures and hypha-hypha interaction has been observed for different species in several studies ((Dikec et al., 2020), (Martínez-Galicia et al., 2023)). For instance, Martínez-Galicia et al. (2023) observed that their fungi initially grew in a tree-like manner characterized by apical growth and branching and, later (after about 11 h), anastomosis began to occur. Mycorrhizal fungi live in symbiotic relationships with one or multiple plants. When the mycorrhizal mycelium connects two or more plants, it is called a Common Mycorrhizal Network (CMN) (Simard et al., 2012). The presence of

\* Corresponding author at: Bioinspired Soft Robotics Laboratory, Istituto Italiano di Tecnologia, Via Morego 30, Genova 16163, Italy.

E-mail addresses: [oscar.sten@iit.it](mailto:oscar.sten@iit.it) (O. Sten), [barbara.mazzolai@iit.it](mailto:barbara.mazzolai@iit.it) (B. Mazzolai).

<https://doi.org/10.1016/j.ecoinf.2024.102670>

Received 29 January 2024; Received in revised form 6 May 2024; Accepted 5 June 2024

Available online 20 June 2024

1574-9541/© 2024 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

symbiotic fungi and CMNs in the soil has been shown to positively affect plants connected to it. These advantages include increased resistance against diseases and herbivorous insects (Babikova et al., 2013; Song et al., 2010), improving resilience to environmental disturbances (e.g., droughts), and incentive plant growth (Simard et al., 2012). Mycorrhizal fungi have recently sparked attention as a possible strategy in crop management to reduce the usage of artificial fertilizers (Alaux et al., 2021; Babikova et al., 2014; Bonfante and Genre, 2010; Gilbert and Johnson, 2017). Meanwhile, other researchers have predicted that mycorrhizal fungi may make natural ecosystems more resilient to climate change (Simard et al., 2012). Because of the high potential benefits, there is a growing interest in deepening knowledge of this natural organism and understanding mycorrhizal fungi's network formation and evolution.

Automated analysis of images of filamentous organisms can provide a more in-depth analysis of the organism's development (De Ligne et al., 2019) and save a substantial amount of time (Cardini et al., 2020) compared to manual measurements in controlled experiments. Recent works have utilized automated image analyses to discover the growth characteristics of other types of fungi. In previous work, we proposed an automated image analysis tool for the non-invasive quantification of spores of *Rhizophagus irregularis* (Sten et al., 2023). Vidal-Diez de Ulzurrun et al. (2015) developed an automated image analysis tool to obtain growth curves of mycelium area and the number of apices of different species of pathogenic fungi aiming to find the optimal growth conditions for the different species (De Ligne et al., 2019). Instead, Sánchez-Orellana et al. (2019) developed a software tool for automatically tracking individual tips and observed that, before branching, the hyphal apex stops growing for a few minutes. In (Fricker et al., 2017; Obara et al., 2012; Pain et al., 2019), algorithmic solutions were proposed to generate and perform statistical analyses of network structures in 12 diverse species of fungi. The authors considered branch and anastomosis points as nodes and the connecting filaments as edges: each edge was assigned a weight depending on their predicted transport resistance, which scales linearly with the hyphal length and inversely with the hyphal thickness (Aguilar-Trigueros et al., 2022). The aforementioned studies are examples of applying microscopy techniques and post-processing image analysis to observe and characterize fungal behaviors in controlled laboratory experiments.

This study proposes an end-to-end solution for the automated analysis of key features in filamentous organisms, such as filament length and network topology. The focus is on the *Rhizophagus irregularis*, a model fungus known for establishing symbiotic relationships with numerous crop plants. Our objective is to provide automated analysis tools to spark light on the morphological characteristics and dynamics of this organism.

### 1.2. Adjacent applications

Two major different applications might challenge similar problems to ours. The first application is the analysis of network structures in biomedicine. Several tools have been developed in recent works for analyzing 3D volumetric microscopy images. In particular, in the field of neuroscience, two important datasets have been created: DIADEM, which contains images of 3D volumetric microscopy images of olfactory neurons imaged with 2-channel confocal microscopy Liu et al. (2018), and BigNeuron, which contains images of 3D volumetric microscopy images of various types of nerve cells. Images are acquired through various microscopy techniques: laser scanning microscopy or 2-photon confocal microscopy and brightfield or fluorescent microscopy. All the nerve cells have been marked with various techniques, including genetic labeling, virus, dye, and biocytin injection (Manubens-Gil et al., 2023). Also, CT scans of bronchi and images of retinal blood vessels have been utilized (Tan et al., 2020). In order to derive useful information from these images, it is crucial that the represented connectivity is as correct as possible; therefore, extensive effort has been put towards improving

the fidelity of such topological reconstructions. Some recent contributions include the analysis reported in Liu et al. (2018), where an algorithm for iterative tracing of the neuron structure is proposed by applying a global threshold followed by applying a fast marching technique. Differently, Tan et al. (2020) focused on detecting branch points in biomedical 3D images to enable graph reconstruction of tree-like biomedical structures. First, a 3D artificial neural network (ANN) was utilized to identify initial candidate points in a computationally efficient manner. Then, a different 2D ANN was trained to process patches projected from multiple scales and multiple views around these candidate branch points to remove false positives and improve reconstruction accuracy. Instead, (Chen et al., 2021) first applied an ANN for segmenting neurons, yielding a binary tensor with ones representing the neurons. The binary tensor was then skeletonized, and the branch points in the skeleton representation were identified as critical point candidates. Around these candidate points, spherical patches were extracted and transformed into multiple 2D patches, which were used as input in a second ANN trained to classify candidate critical points as termination (endpoint), branch, crossover point (overlap), and non-critical point (false positive). This work was extended in (Guo et al., 2021) with a model to extract features like angles and intensities of neural fibers in crossover structures, which allowed separating crossover structures and obtaining adequate reconstruction.

An issue with deep learning methods is that it can be difficult to obtain enough ground truth data. Sharma et al. (2017) utilized crowd-sourcing, involving non-expert users for obtaining manual annotation, and demonstrated the efficacy of their method on karyotyping images of chromosomes.

The other adjacent application is the morphological segmentation of a binary mask where the foreground pixels represent a certain type of land cover deriving from a remote sensor image, and the background pixels represent all other land cover types. The GUIDOS ToolBox and Fragstats are software examples that can segment a binary image into morphological categories (McGarigal et al., 2023; Vogt and Riitters, 2017). The GUIDOS ToolBox includes the Morphological Spatial Pattern Analysis (MSPA) package (Soille and Vogt, 2009), which enables a more detailed categorization: core, connected elements exceeding a size threshold, islet, a smaller disconnected element, and edge and perforation (i.e., external and internal border pixels of core elements, respectively). Furthermore, the narrow corridor segments are divided into bridges (connecting two core elements), branches (emanating from a core element without a destination), and loops (connecting element to itself). Consequently, how the bridges connect the core elements to each other can be modeled as a graph (Vogt and Riitters, 2017). Similarly, the software Fragstats enables the identification of segments with a linear morphology in the categorical map and generates a graph where these segments are represented as edges, and their intersections are represented as nodes. This graph may then be used to calculate ecologically significant parameters (McGarigal et al., 2023). These large-scale 2D networks are of interest, for instance, in landscape ecology (McGarigal et al., 2023; Vogt and Riitters, 2017). There are two key differences in this application with respect to mycelium analysis: 1) since the categorical land cover network is truly a 2D network, overlap detection is not of interest, and 2) a substantial part of the network derives for blob-like objects with irregular shapes.

### 1.3. Related works

The process of automatically analyzing an image of a filamentous organism can be divided into two core steps: 1) detecting or segmenting the organism of interest from a background and 2) creating a graph representation of its structure for further analysis.

Several strategies have been implemented to detect the filaments in the sample image. One option is global thresholding (i.e., binarization by setting a threshold), which can be done manually (Barry et al., 2009, 2015; Brunk et al., 2018) or automatically (Dikec et al., 2020; Martínez-

Galicía et al., 2023). Specifically, to automatically determine an optimal threshold value, Kapur's maximum entropy method was adopted (Dikec et al., 2020; Ledoux et al., 2022, 2024). Global thresholding has been suggested to be unsuitable for filamentous structure since, depending on the threshold, thinner filaments may be omitted, or artifacts may be introduced (Fricker et al., 2017). Furthermore, it is sensitive to uneven illumination. This limitation has been surpassed with local adaptive thresholding, as adopted in (Sachs et al., 2019). Another option to segment filamentous images is to adopt ridge detection methods, as done in (Fricker et al., 2017; Lopez-Molina et al., 2015; Obara et al., 2012; Pain et al., 2019; Vidal-Diez de Ulzurrun et al., 2015; Vidal-Diez de Ulzurrun et al., 2019). Filaments, in fact, appear in an image as lines displaying a local maximum towards a certain direction. In particular, in (Lopez-Molina et al., 2015; Vidal-Diez de Ulzurrun et al., 2015, 2019; Wang et al., 2019), second-order anisotropic Gaussian kernels (SOAGK) were utilized. Differently, Obara et al. (2012) detected a skeleton representation of the mycelium by 1) enhancing the image ridges using phase congruency-based tensors (PCT), 2) identifying the image watersheds using the watershed transform, and 3) pruning branches using a cost function to reduce over-segmentation (Obara et al., 2012). Fricker et al. (2017) evaluated diverse ridge enhancement techniques in the framework of filamentous fungi. The authors found that SOAGK can achieve a higher detection performance but is more parameter-sensitive than PCT (Fricker et al., 2017). Zhang et al. (2019) proposed a combination of thresholding and a level-set method, where the segmentation was initialized manually and, as filaments grew, the growth was tracked automatically. The user can also adjust the filament segmentation manually through a graphical user interface (GUI) (Zhang et al., 2019). Concerning biological filamentous network analyses, Xu et al. (2021) trained a Convolutional Neural Network (CNN) with 727 manually annotated images to segment leaf vein structure. This strategy outperformed other segmentation methods previously applied to leaf vein networks (Xu et al., 2021).

Approaches to creating a graph representation typically depend on the segmentation process. Turning a binary image into a graph representation, sometimes referred to as vectorization, is common in biological and biomedical applications and specifically for analyzing vascular structures (Kaufhold et al., 2012). This approach to represent a structure has been implemented for analyzing filamentous organisms in several studies (Aguilar-Trigueros et al., 2022; Dikec et al., 2020; Dirnberger et al., 2015; Fricker et al., 2017; Martínez-Galicía et al., 2023; Obara et al., 2012; Pain et al., 2019; Sachs et al., 2019; Vidal-Diez de Ulzurrun et al., 2015; Vidal-Diez de Ulzurrun et al., 2019). Most vectorization approaches are based on skeletonizing or thinning the binary image, thus reducing it to a one-pixel-wide skeleton representation, often achieved using the medial axis transform (Kaufhold et al., 2012). A problem with skeletonization methods, including the medial axis transform, is the ill-conditioning, meaning they frequently output a result with several small, spurious branches (Kaufhold et al., 2012; Lasser and Katifori, 2017). Skeletonization was utilized for analyzing filamentous organisms in (Barry et al., 2015; Cardini et al., 2020; Dirnberger et al., 2015; Pain et al., 2019; Sachs et al., 2019; Sánchez-Orellana et al., 2019).

The conversion of a skeleton representation to a morphological graph can be done by identifying the junctions (nodes) and their connectivity (edges) (Sachs et al., 2019). By contrast, in (Lopez-Molina et al., 2015; Vidal-Diez de Ulzurrun et al., 2015, 2019), the adopted ridge detection algorithm directly outputs the binary skeleton representation. A different approach for generating a graph representation was proposed by Lasser and Katifori (2017), where the binary image was divided into triangles. Then, the triangles were classified based on their connection to other triangles. In some works, Geographic Information Systems (GIS) software was used to analyze the rasterized filamentous fungi. Specifically, the software ArcGIS Pro was used to a) vectorize the segmented mycelium (Dikec et al., 2020; Ledoux et al., 2022, 2024), b) compute intra-thallus surface areas (Dikec et al., 2020; Ledoux et al.,

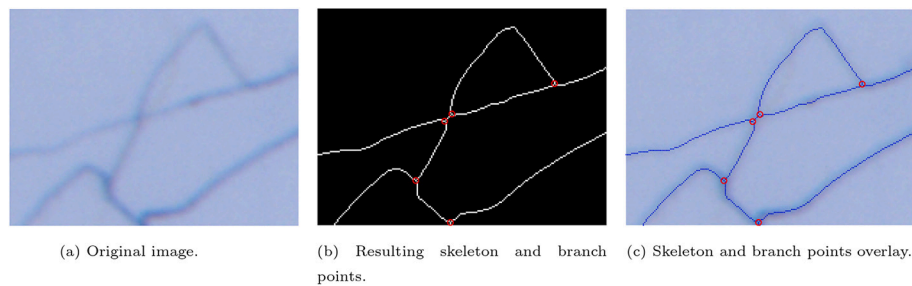
2024), c) compute branching angles (Ledoux et al., 2022), and d) compute line density (Ledoux et al., 2024).

Both segmentation and vectorization are operations where errors may be introduced. Several strategies have been implemented to filter out artifacts and reduce the influence of these errors. In (Brunk et al., 2018), segmented components were filtered out based on area and circularity (Brunk et al., 2018). When analyzing images of filamentous structures, it is common to set a minimum branch length threshold and remove shorter branches. This operation is sometimes referred to as pruning and has been applied 1) when the skeleton representation is obtained through skeletonization (Barry et al., 2009, 2015), 2) in the case in which the skeleton representation is obtained directly from the detection algorithm (Vidal-Diez de Ulzurrun et al., 2019), and 3) when the structure is obtained from a segmented binary image (Lasser and Katifori, 2017). Indeed, finding a method to discriminate with certainty between a short filament and a spurious branch remains an open problem (Kaufhold et al., 2012). A common strategy is to provide a Graphical User Interface (GUI) to let the user manually remove artifacts (Cardini et al., 2020; Lasser and Katifori, 2017; Zhang et al., 2019). A peculiar strategy for making the manual removal of artifacts safer and more intuitive was the graph filters proposed by Dirnberger et al. (2015). This work provided a semi-automatic GUI to filter out components based on their topology (Dirnberger et al., 2015).

In works where a graph representation of the mycelium was created, overlapping crossings of filaments were either not considered or manually identified (Aguilar-Trigueros et al., 2022; Dikec et al., 2020; Dirnberger et al., 2015; Fricker et al., 2017; Obara et al., 2012; Pain et al., 2019; Sachs et al., 2019; Vidal-Diez de Ulzurrun et al., 2015; Vidal-Diez de Ulzurrun et al., 2019). It is worth mentioning that the instances of such crossings in the image sets considered in these works were very few (Aguilar-Trigueros et al., 2022; Dikec et al., 2020; Dirnberger et al., 2015; Sachs et al., 2019; Vidal-Diez de Ulzurrun et al., 2015, 2019). Nevertheless, Lasser and Katifori (2017) (Lasser and Katifori, 2017) discussed how the 2D projection of 3D objects in images introduces spurious nodes and distorts edge lengths. The authors addressed this issue by implementing a GUI for graph manipulation. In addition, Aguilar-Trigueros et al. (2022) pointed out that disregarding the possibility of overlapping hyphae may overestimate the network connectivity. As a future possibility to identify such overlaps, Dikec et al. (2020) proposed to track the growth of the filaments. In the same work, since one of the main inquiries was to quantify the number of branches and anastomoses, the authors proposed a proportional relation between the number of geometrical nodes and actual nodes (Dikec et al., 2020). In Zhang et al. (2019), a GUI allowed the user to mark each filament, and the software tracked each filament's growth even if it overlapped with other filaments (Zhang et al., 2019). However, to the best of our knowledge, fully automatically identifying and separating overlapping, crossing filaments in fungal images, such as the example shown at the center of Fig. 1, remains an open problem.

#### 1.4. Paper contribution

Working towards the automated analysis of the network topology of filamentous organisms, this paper proposes a six-step ridge-based mycelium detection strategy combining a) second-order steerable filtering for enhancing ridge-like features and b) local adaptive thresholding for obtaining a continuum detection output (i.e., output without gaps). The approach is then demonstrated on fungal images that differ in fungus species, image acquisition system, stained and non-stained samples, contrast, scale, background, and noise level (e.g., introduced by dirtiness), thus demonstrating its generalizability. We refer to our mycelium detection procedure as a detection algorithm rather than a segmentation algorithm since the output does not correspond to the actual filaments, but it is a binary image with positive pixels around the center of the filaments. Then, we propose a novel, fully automated technique to identify and match probable overlaps of filaments in a



**Fig. 1.** Resulting skeleton output around an overlapping pair of hyphae. Original image adapted from (Cardini et al., 2020).

binary skeleton matrix, and we evaluate its performance. The result is a highly automated software tool for analyzing mycelium images that can be applied to filamentous organisms that grow into complex and irregular networks. This advances the state of the art in fungal analysis since it can allow studying biological models not yet fully characterized due to their complexity, such as the case of *Rhizophagus irregularis*.

## 2. Methodology

### 2.1. Test images

For the algorithm development and testing, three different publicly available sets of fungal images were used. The first image set originates from (Cardini et al., 2020). It contains 20 images of low-density, extraradical mycelium of the mycorrhizal fungus *R. irregularis* stained with Trypan blue in lactic acid. This set is available at <https://gitlab.iit.it/EDelDottore/hylength/>, and it has previously been used to validate a semi-automated length measurement tool (Cardini et al., 2020). In the current work, we annotated these images for our study. Example images from this dataset are used throughout the manuscript to exemplify the methodology and show validation results. The second dataset is the Ghent University Fungal Images (GUF1-1), available from <https://doi.org/10.13140/RG.2.1.4441.1607>. This data set comprises 100 images of five commercially relevant fungal mycelium species and corresponding hand-made ground truth annotations. The image set has been used in previous studies, including (Lopez-Molina et al., 2015; Shokouh et al., 2021; Wang et al., 2019). An example is shown in Fig. 2a. The third image set consists of ten images of the nematode-trapping fungus *A. oligospora* stained with SR2200 and available at <https://github.com/hsueh-lab/FFT>, plus five extra

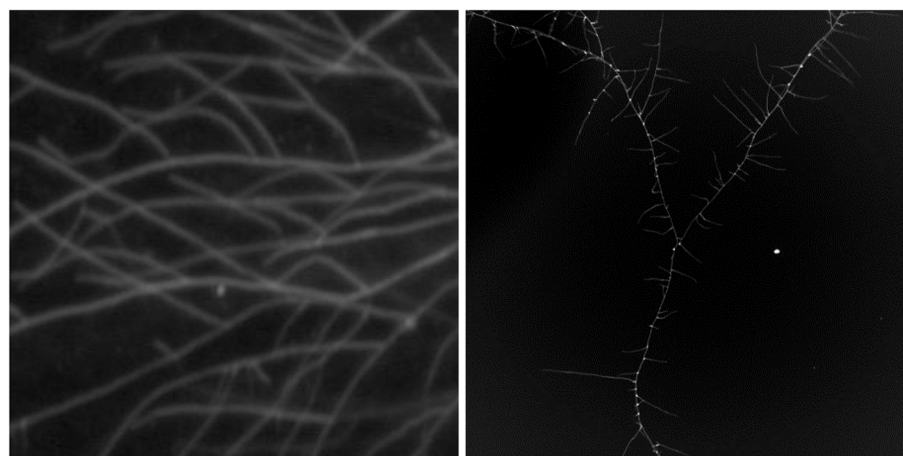
images sent to us by the authors. This image set was developed to validate the software tool Fungal Feature Tracker (Vidal-Diez de Ulzurrun et al., 2019). An example is shown in Fig. 2b.

### 2.2. Automated scale estimation

To extract a standard unit measure from pixel information, it is necessary to estimate the scale of the image, i.e., the real-world surface area to which a pixel corresponds. In the image set provided by Cardini et al. (2020), the images had a scale bar with text inserted by the microscope software (Cardini et al., 2020). Since the scale bar has a very different color from the sample, its presence can cause a strange gray-scale image histogram, which may introduce segmentation problems. Therefore, removing the scale bar before filament detection was imperative. Performing a separate segmentation of this scale bar and its text has two benefits: 1) it removes the risk of the scale bar disrupting the thresholding, and 2) it allows to fully automatically estimate the scale of the image with higher precision than manual methods like the “Measure” tool in ImageJ/Fiji (Schindelin et al., 2012). We implemented an ad-hoc algorithm for 1) segmenting the scale bar and corresponding text, 2) interpreting the text using Matlab’s embedded OCR library, and 3) replacing the segmented pixels with a color similar to the background.

### 2.3. Detection of mycelium

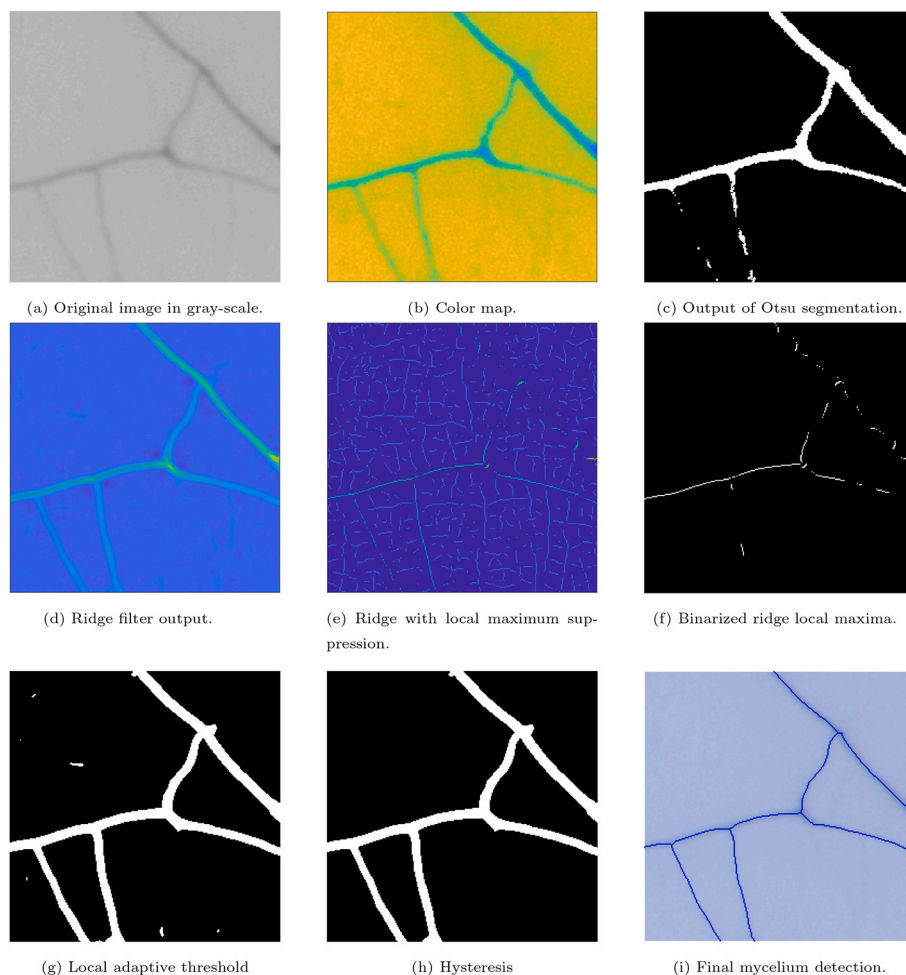
In the considered data sets, all the RGB images had unimodal histograms on all three channels, making zeroth order segmentation challenging, e.g., using Otsu’s method (Otsu, 1979) (Fig. 3c). Therefore, we opted for ridge detection following the example of (Fricker et al., 2017; Lopez-Molina et al., 2015; Obara et al., 2012; Pain et al., 2019; Vidal-



**Fig. 2.** Examples of image utilized for validating the mycelium detection algorithm. (a) Example image from the data set GUF1-1 (reproduced with permission from Jan M. Baetens). (b) Example image from the validation set of FFT (reproduced with permission from Guillermo Vidal-Diez de Ulzurrun).

**Fig. 2.** Examples of image utilized for validating the mycelium detection algorithm.





**Fig. 3.** Workflow of our proposed mycelium detection algorithm. a) It starts originating a gray-scale image. b) The colormap shows that the filaments down on the left-hand side have an intensity similar to the background. c) Provide an example of why global threshold binarization is unsuitable in this context. d) The steerable ridge filtering output enhances the filaments with respect to the background. e) The ridge filter output with local maximum suppression shows several fragmented lines; however, those in the center of the filaments have higher numerical values than those in the rest of the image. f) The output from binarizing the image in (e) using Otsu's method shows fragmented lines. g) Shows the output of applying local adaptive thresholding to (d), the filaments are detected, but some artifacts are also detected. h) A hysteresis between (f) and (g) results in a binary image representing the filaments without the artifacts. i) The final skeleton output superimposed on the original image.

Diez de Ulzurrun et al., 2015; Vidal-Diez de Ulzurrun et al., 2019). In a recent review on filter-based ridge detection by (Shokouh et al., 2021), the authors found that steerable filters of the type proposed in (Freeman and Adelson, 1991) and (Jacob and Unser, 2004) performed better than other techniques used on natural images (Shokouh et al., 2021). In the following section, we describe the procedure we implemented to detect the mycelium, and Fig. 3 shows the resulting output of the proposed steps.

The gray-scale image (Fig. 3a) is filtered with a steerable 2<sup>nd</sup> order Gaussian filter (Pang, 2022) with the standard deviation  $\sigma$ , which must be set depending on the image scale, and a vector of angles  $\Theta$  in the range from 0° to 360°, with a selected step size (Fig. 3d). Following the filtering, the segmentation is performed in three steps: 1) a non-maximum suppression is applied to the filter output (Fig. 3e), and the local maxima of interest are segmented using Otsu's method (Otsu, 1979) (Fig. 3f), 2) a local adaptive thresholding using Bradley's method is applied (Bradley and Roth, 2007) (Fig. 3g), and 3) a hysteresis of step 1 and step 2 is performed (Fig. 3h). The segmented output is then skeletonized using the medial axis transform and pruning branches with a length corresponding to less than 22  $\mu\text{m}$  (The MathWorks Inc, 2022a) (Fig. 3i).

The skeletonization output can be used to estimate the length of the

mycelium. We obtain millimeters represented by each pixel ( $S$ ), as described in Section 2.2. Then, we compute the total mycelium length by quantifying the number of 4-connected pixels  $n_4$  and the number of exclusively 8-connected pixels  $n_8$ . The two sets are separated using the MATLAB function *bwareaopen*. Thus, we can compute the mycelium length ( $L$ ) using eq. 1.

$$L = n_4 \cdot S + n_8 \cdot S \cdot \sqrt{2} \quad (1)$$

#### 2.4. Separation of filaments

In this section, we present step-by-step our approach for generating a network representation that is more accurate with respect to the conventional morphological graph from a sample image by 1) assessing the local geometry of the binary skeleton around BPs (Section 2.4.1), 2) extracting a preliminary morphological graph, similar to the ones built in (Aguilar-Trigueros et al., 2022; Dikec et al., 2020; Dirnberger et al., 2015; Fricker et al., 2017; Obara et al., 2012; Pain et al., 2019; Sachs et al., 2019; Vidal-Diez de Ulzurrun et al., 2015; Vidal-Diez de Ulzurrun et al., 2019) (Section 2.4.2), 3) the BP pairs that are likely to represent a filament overlap (i.e., between them two filaments are crossing each other) are identified based on vicinity and local approximations of the filament directions (2.4.3), and 4) finally each filament is assigned with

an ID and a filament graph representation is built (Section 2.4.4). We refer define a filament overlap as the area where two filaments cross each other.

#### 2.4.1. Branch point assessment

From the binary skeleton matrix generated in 2.3, the branch points can be identified using the Matlab native function `bwmorph`. This function returns the following: i) points representing the center of a T-crossing or Y-crossing (i.e., branch points with three arms), ii) points representing the center of an X-crossing or  $+$ -crossing (i.e., branch points with four arms), and iii) some false positive branch points (i.e., branch points with one or two arms). For each candidate branch point (CBP), a patch of the binary skeleton surrounding the branch point is provided (Fig. 4a). The dimension of the patch must be set to provide a local representation of the filament geometry, and, in our case, a patch side ( $l$ ) corresponding to 0.022 mm provided adequate representations. From this patch, components not connected to the branch point are first set to zero. Then, the  $3 \times 3$  pixels square surrounding the branch point is set to zero (Fig. 4b), and the remaining connected components in the patch are finally assigned an integer label (from 1 to max 4) using the Matlab native function `bwlabel` (Fig. 4).

Each arm may include other branch points, which may be a false or true positive. In the latter case (e.g. Fig. 5), it is important that the arms of the other branch point (Fig. 5b) are disregarded, and this can be achieved by identifying other branch points overlapping with one of the arms and breaking the connected skeleton lines around the other branch point. If the arm is split into multiple connected components, only the one connected to the original branch point is considered (Fig. 5c). A rare yet important edge case occurs when the Euler characteristic  $\chi$  of the image patch is less than one, i.e., there is an encircling (e.g., Fig. 5). An encircling contains three or more branch points, which would all be falsely removed if not handled. Thus, if  $\chi < 1$ , the following operations are performed: 1) the points compiling the encircling are identified (Fig. 5b), 2) the two branch points that are part of the encircling and connected to the CBP are identified, and 3) the part of the encircling that does not represent the shortest geodesic path between a BP and the CBP is removed (5c).

Some BPs may escape because they overlap with the patch edges (Fig. 5). In these cases, the position of the missing BP can be approximated as the point with the second-greatest geodesic distance further from the other BP. Instead, if two BPs need to be approximated, they are placed at half the geodesic distance from the furthest point of the encircling.

In Matlab, the image coordinate system has the origin at the top left corner of the image, with the x-axis pointing to the right and the y-axis pointing downwards (The MathWorks Inc., 2022a). To extract the ridge filter response values for the arms of the intersection, we translate the arm coordinates to the global image coordinate system and obtain the median ridge filter response for each arm. Approximating the direction

of each arm as a 2D vector allows for the estimation of the angles, both between arms of the same intersection point and between arms of diverse intersection points. To obtain a 2D vector estimating the direction of each arm, the coordinates of each arm (Fig. 4c & 5c) must be transformed from the patch coordinate system into a Cartesian coordinate system with the origin at the center of the patch. Thus, the affine transformation in eq. 2 is used to transform the image patch coordinates  $(x, y)$  to the Cartesian coordinates  $(X, Y)$  with the origin in the point  $(x_0, y_0)$ , which in this case corresponds to the center of the image patch.

$$\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} -1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \quad (2)$$

After transformation, a linear least squares regression can be applied to estimate the line inclination and provide the arm direction. Then, the predicted values for the x-coordinates are computed, and the point resulting the furthest from the connection point is identified. Finally, the transformed coordinates of the connection point are subtracted from the coordinates for the furthest point on the estimation line to obtain the arm vector. If the branch point has three intersection arms (most common case), we assume that the three-way junction consists of one main filament, to which two arms belong, and one subordinate filament or the branch-out arm, to which the third arm belongs. In Fig. 4c, the branch-out arm is arm 2, marked in green. We may define the subordinate filament as the arm with the weakest median ridge filter response value. This strategy works well, as shown in Table 1. However, it is very sensitive to the presence of dirt or accumulations of staining ink. An alternative is to compute the angle between each pair of director vectors of the arms (Fig. 6). This strategy also works well in most cases, as shown in 1). However, in intersections with a Y-shaped morphology (i.e., the angles are very similar), since the direction approximation is not exact and depends to some degree on local aspects of the mycelium detection, the result becomes a matter of chance.

To improve the results, we merged the two strategies. If the difference between the angles (Fig. 6) exceeds a certain threshold ( $\Theta_{switch}$ ), the second strategy is utilized; otherwise, we assume the arms form a Y-shape, and the arm with the weakest median ridge filter response is set as the branch-out arm. This combined strategy yields an improved result, as shown in Table 1. Having defined the branch-out arm, we can define the branch-out point as connecting the branch-out arm to the corresponding branch point. We define the branch-out vector as the vector that approximates the direction of the branch-out arm and the two main filament vectors as the two vectors that approximate the direction of the two other arms. Fig. 7 shows six examples of local vector approximations of BPs with three different types of filament overlaps. How these BPs are subsequently identified as part of a filament overlap is detailed in Section 2.4.3.

#### 2.4.2. Building the morphological graph without overlaps

To abstract the fungal network morphological graph, we first define

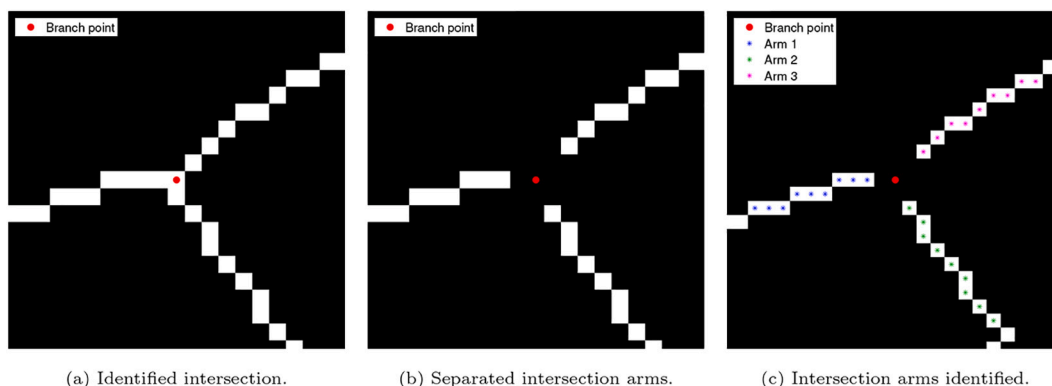


Fig. 4. Extraction of the intersection arms in the simplest case.

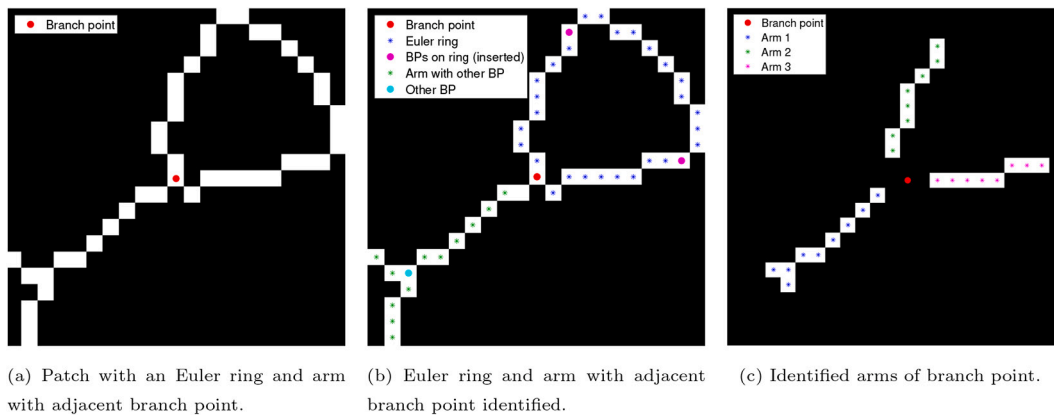


Fig. 5. Extraction of the intersection arms in a patch with Euler ring and adjacent branch point.

Table 1

Number of correctly identified and matched branches and crossings assessed on a complex image (see Fig. 9a in Section 2.5.2).

|           | Ideal matching function | Min ridge | Max angle | Combined |
|-----------|-------------------------|-----------|-----------|----------|
| Branches  | 156                     | 123       | 130       | 132      |
| Crossings | 18                      | 17        | 15        | 17       |

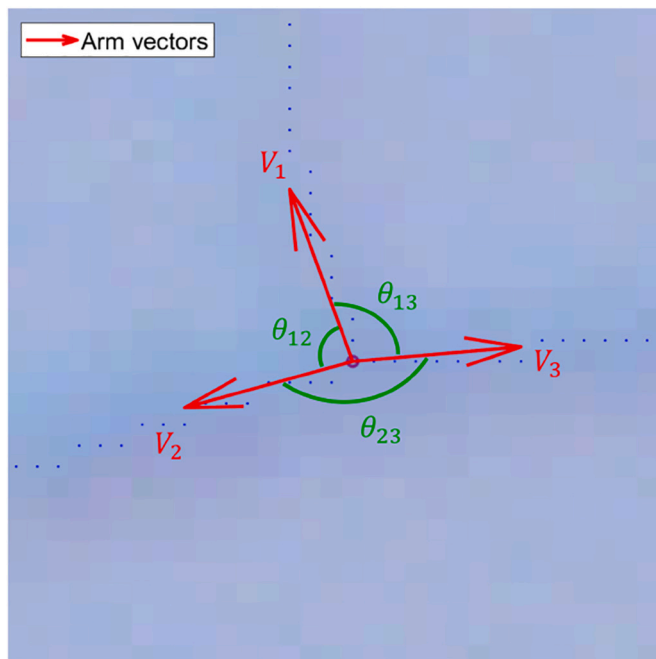


Fig. 6. Angles between arm vectors.

an undirected morphological graph with the BPs (Section 2.4.1) as nodes and the geodesic distances between the BPs in the binary skeleton (Section 2.3) as edges. We built this undirected graph by a) removing all the valid branch points (as well as assuring 8-connections are broken) from the binary skeleton, b) inserting one of the branch points at the time, and c) identifying the tip points (corresponding to hyphal apices) connected to the branch point and the quasi-Euclidean geodesic distances from the branch point to the tip points. The branch points connected to the tip point are set as connected to the current branch point, with edge weights corresponding to the geodesic distances. The resulting undirected graph representation is used to identify mutually closest branch points, which is necessary to identify pairs of branch points that

may constitute a filament overlap.

### 2.4.3. Identifying overlap pairs

As stated in Section 2.4.1, a BP may have three or four arms. If the number of arms is four, the structure generally represents an overlap of two (quasi-) perpendicular filaments. Hence, an ad-hoc solution is applied, where the points on opposite sides of the BP are identified as belonging to the same filament. The same method is applied for pairs of BPs separated by a distance ( $D \leq 2\sqrt{2}$ ).

Instead, in cases like the examples shown in Fig. 1 and Fig. 7, the overlap's resulting skeleton representation consists of two BPs with three arms each. In particular, in the example shown in Fig. 7c, two filaments cross each other at a fairly acute angle, causing the resulting binary skeleton output to be characterized by two Y-intersections separated by a relatively long line. A similar morphology is observed in Fig. 7a and b. However, the separating lines are shorter since the overlap angle is less acute. Therefore, to match the pairs of branch points making up an overlap, we propose a technique considering the geodesic distance between the branch points and the angle between the filaments. Specifically, we define the set of candidate branch point pairs  $A_{cand}$  and add to this set all those pairs of points that are each other's mutually closest neighbor in the morphological graph. An example of such a pair  $BP_1$  and  $BP_2$  is shown in Fig. 8a. For each pair of BPs, the vector  $\vec{V}_1 = BP_1 - BP_2$  is defined (Fig. 8b). We determine the angle  $\theta_1$  between  $\vec{V}_1$  and the branch out vector of  $BP_2$ ,  $\vec{V}_{B2}$  with the dot product. If  $\theta_1 \geq 90^\circ$ , the pairing is considered plausible. Otherwise, it is removed from the set candidate pairs.

Fig. 7a shows an example in which two filaments cross each other at an almost perpendicular angle. In this case, the angles between the branch-out vector and main filament vectors tend to be perpendicular, and the resulting geodesic is small. By contrast, if the filaments cross at a more acute angle, the angles between the branch-out vector and main filament vectors result in acute or obtuse, and the geodesic is greater. Hence, to determine if the remaining BP pairs in  $A_{cand}$  belong to an overlap, we propose a criterion based on the angle between the filament and the distance between branch points. By subtracting one main filament vector from the other, a single vector  $\vec{V}_{Main}$  approximating the direction of the main filament is obtained, as shown in Fig. 8c. If the angle  $\theta_{Bi}$  ( $i = 1,2$ ) between  $\vec{V}_{Main}$  and the branch out vector of each branch point is very acute or obtuse, likely the candidate pair forms an overlapping, even if the geodesic distance ( $D$ ) between them is large. On the contrary, if the  $\theta_{Bi}$  are close to a right angle, the distance between the branch points should be short for the candidate pair to remain in the set of branch point pairs likely to make up an overlapping. Considering this, we defined the criterion in (3), where  $D_T$  is a distance threshold,  $\theta_T$  is an angle threshold, and  $c$  is the resulting Boolean determining if the BP pair

remains in the  $A_{cand}$  set. For the remaining BP pairs in  $A_{cand}$ , the related connection points are removed from the set of connection points and added to the set of passage points, i.e., pairs of coordinates indicating points where filaments overlap.

Algorithm (1) to create a set of lists specifying which filaments should be merged.

**Algorithm 1.** Identify and Merge Traversing Filaments.

---

```

Input :  $U \in \mathbb{N}^N, V \in \mathbb{N}^N$ 
Output: Traversals
1 while  $U \neq \emptyset$  do
2    $currentID \leftarrow U(1)$ 
3    $traversal \leftarrow U(1)$ 
4   while  $currentID \in U \vee currentID \in V$  do
5      $X \leftarrow currentID \in U$ 
6      $Y \leftarrow currentID \in V$ 
7     if  $X \neq \emptyset$  then
8        $correspID \leftarrow V(X(1))$ 
9        $U(X(1)), V(X(1)) \leftarrow \emptyset$ 
10       $traversal \leftarrow traversal \cup correspID$ 
11    else if  $Y \neq \emptyset$  then
12       $correspID = U(Y(1))$ 
13       $U(Y(1)), V(Y(1)) \leftarrow \emptyset$ 
14       $traversal \leftarrow traversal \cup correspID$ 
15     $currentID \leftarrow correspID$ 
16   $traversals \leftarrow traversals \cup traversal$ 

```

---

$$c = (D \leq D_T) \vee (|90 - \theta_{Bi}| \leq \theta_T) \quad (3)$$

Furthermore, since hyphae grow from the apex, a configuration as the one reported in Fig. 7c is unlikely to derive from two branches, otherwise one of them would have had to branch backward with a very obtuse angle with respect to the mother hypha. Previous experimental observations of other types of filamentous fungi have shown that the branch angles tend to remain within a certain range (Ledoux et al., 2022). Therefore, if the angle threshold  $\theta_T$  in 3 is large enough, this is an unlikely source of false positives.

#### 2.4.4. Rebuilding the filament graph

Given the binary skeleton structure (Section 2.3), the set of branch connection points (Section 2.4.1), and the set of passage points (Section 2.4.3), the skeleton structure can be divided into individual filaments. All the branch connection points and passage points are then set to zero, obtaining a binary skeleton with multiple thin, connected components. We then apply the Matlab native function *bwlabel* to assign an ID to each connected component.

Consequently, the filaments connected through a pair of passage points must have their IDs reassigned so that all components of the same filament have the same ID. In this context, we loop over pairs of passage points and extract a  $3 \times 3$  patch from the matrix of labeled components around each passage point. To each patch are associated two different ID numbers: 1) the ID of the main filament and 2) the ID number filament to be merged. For a given pair of passage points, the main filament is the same; hence, the two unique IDs in the two  $3 \times 3$  patches are the IDs of the filaments that should be merged.

A filament may overlap multiple other filaments. Therefore, we create a set of *traversals* where each *traversal* is a list of filament ID numbers for filaments belonging together. Given an  $N \times 2$  vector where each row is a pair of ID numbers that belong together, we utilize

Once all the traversals have been completed, the filaments that are the same are set to have the same ID number. If we consider the branch connection points as links between different filaments, an undirected graph representation of the filamentous structure can be created. Starting with the highest ridge prominence filament, a breadth-first search where every filament connected to the current filament is added as connected in the undirected graph. Thus, an undirected graph is created with the filaments as nodes and the branch points and anastomoses as edges. Furthermore, a more accurate representation of the morphological graph can be obtained considering the remaining BPs and tips as nodes and the filaments connecting them as edges.

#### 2.4.5. Generating the morphological graph with overlaps

The common way to model a graph representation of a 2D mycelium image is to consider branches/anastomoses as nodes of degree 3, tips/apices as nodes of degree 1, and the filaments between them as edges with weights corresponding to the length of the filament between the nodes (Aguilar-Trigueros et al., 2022; Dikec et al., 2020; Vidal-Diez de Ulzurrun et al., 2015). Generating such a representation considering overlaps would be more accurate in extracting the number of nodes and distances between branch points and defining the network structure. Such a graph can be generated given the ID filament map (Section 2.4.4) and the coordinates of the identified branch points, passage points, and tip points. First, a map containing all branch points and a map with all passage points are generated. Then, we iterate over the set of labeled filaments, and for each filament, the nodes on the filament are identified. Each node can be an end node or a middle branch node.

Each filament has two endpoints, and each endpoint can be either a tip or a branch point. These endpoints are identified in two steps: 1) the morphological endpoint operation is applied, and if one or two identified points are present in the set of identified tip points, they are added



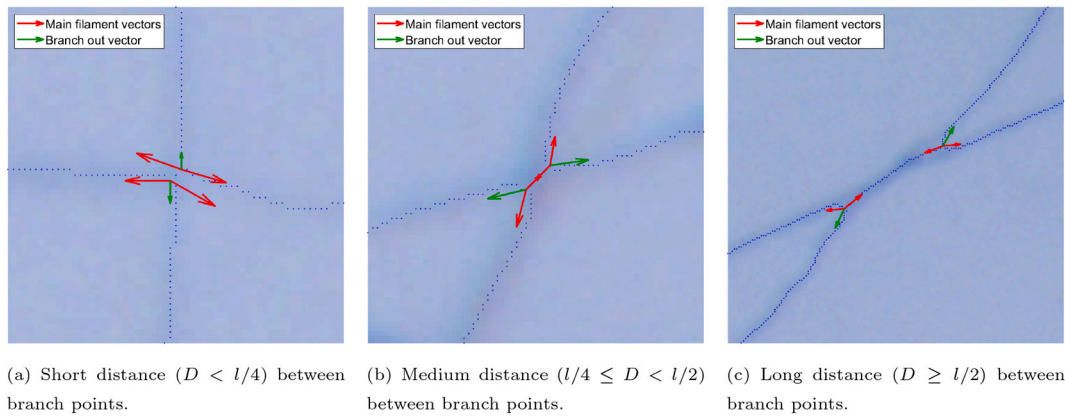


Fig. 7. Examples of filament overlaps.

to the set of end nodes; 2) if none or only one of the filament endpoints are in the set of tips, then the remaining end nodes are connected to other filaments and, by definition, it is connected as a secondary filament. Consequently, the distances from all endpoints to all connection points are calculated, and the connection points with a distance  $\leq \sqrt{2}$  to an endpoint are identified, and the connection points' corresponding branch points are added to the set of end nodes. On the contrary, the middle branch nodes are, by definition, branch points where another filament connects to the current filament and can thus be identified with a simple "and" operation between the filament and the map of branch points. It is important to highlight that all branch points are middle branch nodes in one filament and end nodes in another filament, and all tip points are end nodes in one filament. To repair any gaps in the filament, the passage points attached to the filament are identified, and a straight line between these pairs of passage points is created.

To identify the topological order of the filament's nodes and compute the weights of the edges between them, one of the two end-points are selected randomly and the geodesic distance to all the other nodes present on the filament is computed using the MATLAB function *bwdistgeodesic* with the option "quasi-Euclidean". Then, the edges in the subgraph representing the current filament are generated iteratively. First, an edge from the starting node to the closest node is created, and then from the closest node to the second closest node, and so on. This generates a table of edges with coordinates of the source node, coordinates of the target node, and weight, allowing for the creation of a graph. In the typical case, all nodes have a degree of 1 or 3. However, due to spurious filament detections or other errors up streams, self-loops are deleted, and nodes with degree 2 are replaced with an edge between the node's neighbors.

## 2.5. Validation

### 2.5.1. Validation of mycelium detection

The automated scale bar reading was tested on the images of in-vivo and in-vitro *R. irregularis* samples from Cardini et al. (2020) and compared to manual extractions of the scale bar. The aim was then to detect highly irregular lines composed of a set of pixels by quantifying the number of hits, misses, and false positives. Since the human ground truth annotation can never be perfect, it is important that a) some discrepancy between pixel detection and manual annotation is tolerated, and b) there is a one-to-one correspondence between pixels in detection output and ground truth so that each pixel is counted only once. To this end, we adopted the strategy originally proposed by Liu and Haralick (2002), where an optimal matching problem is formulated. Martin et al. (2004) proposed a sparse version of the method presented in Liu and Haralick (2002) and this was then adopted for the evaluation of detection algorithms for hyphal filaments in several works (Fricker et al., 2017; Lopez-Molina et al., 2015; Vidal-Diez de Ulzurrun et al., 2019; Wang et al., 2019).

In this study, the hand-labeled ground truth images were loaded, binarized, and skeletonized to assure adequate comparability. Then, we computed a rectangular cost matrix, where a) each row represents a ridge pixel detected by the algorithm, b) each column a ridge pixel in the ground truth, and c) the element value is represented by the Euclidean distance between each pair of points. The assignment problem is then solved using the MATLAB function *matchpairs*, with half the tolerance as cost of non-assignment (The MathWorks Inc, 2022b). Consequently, we defined a) the number of assigned pairs as the number of true positive (TP) ridge pixels, b) the number of unassigned columns as the number of false negative (FN) ridge pixels, and c) the number of unassigned rows as

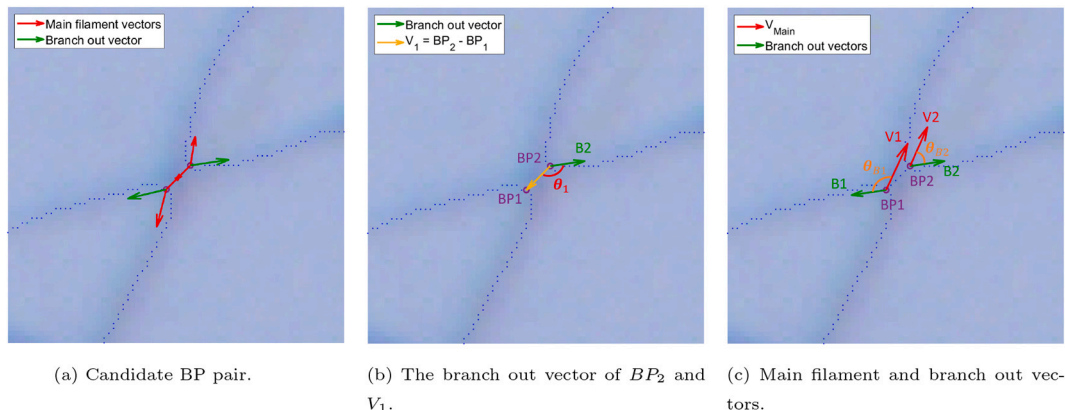


Fig. 8. Auxiliary vectors for determining if a pair of BPs make up a filament overlap.

the number of false positive (FP) ridge pixels. Finally, we assumed as true negatives (TN) the pixels not present in any of these categories.

For testing the detection performance, we tested a) the precision (4), the recall (5), the  $F_1$ -measure (6), and Mathew Correlation Coefficient (MCC) (7) of the pixel detection following (Vidal-Diez de Ulzurrun et al., 2019; Wang et al., 2019), and b) length measurement test following (Cardini et al., 2020). The test (a) was applied to the images in the GUF1 dataset with corresponding ground truth provided by the authors in (Lopez-Molina et al., 2015; Vidal-Diez de Ulzurrun et al., 2019) and to the image set provided by the authors in (Cardini et al., 2020) with ground truth annotations performed by us, utilizing the Matlab Image Labeler tool (The MathWorks Inc., 2022a). Test (b) was applied to the image set provided in Cardini et al. (2020), and manual measurements were obtained through manual tracing using the Matlab Image Labeler tool (The MathWorks Inc., 2022a).

In case (a), to compensate for small displacements of the ground truth, for all three datasets, we used the tolerance used by Wang et al. (2019), (i.e.,  $6\sqrt{2}$  pixels) was adopted as tolerance value. This type of statistical evaluation measures suffers from the issue that they do not reward closeness or penalize distance from the actual ridge line, which in the context of edge detection is problematic (Lopez-Molina et al., 2013). Our visual inspections suggested that the automatic ridge localization was typically more accurate than the manual annotations and that errors were caused by a lack of detection or detection of background objects. Consequently, we believe that the adopted measures are suitable for the application.

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

$$F_1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} = \frac{2TP}{2TP + FP + FN} \quad (6)$$

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (7)$$

In case (b), the goal was to assess the strength of agreement between the gold standard method (manual tracing) and the proposed automated method measuring the mycelium length. Lin's Concordance Coefficient ( $\rho_c$ ) (McBride, 2005) is a commonly adopted method to assess agreement strength. Since the relation between our measurement and the manual one can be regarded as a statistical population of continuous random variables, following the guidelines by McBride (2005), we considered the one-sided 95% confidence limit of  $\rho_c$  which was computed using the software provided in (Mathew, 2018).  $CI_{low95\%}(\rho_c) < 0.9$  as poor,  $0.9 < CI_{low95\%}(\rho_c) < 0.95$  as moderate,  $0.95 < CI_{low95\%}(\rho_c) < 0.99$  as substantial, and  $CI_{low95\%}(\rho_c) > 0.99$  as almost perfect,

considering that we only had 20 samples whereas it is recommended to include at least 25 samples (McBride, 2005).

### 2.5.2. Validation of separation of filaments

The software's capability to correctly separate the filaments was assessed in two steps: 1) identification of branches and overlaps, and 2) correctness of intersection arm matching. In case (1), we annotated two selected images (Fig. 9) from (Cardini et al., 2020) using the Image Labeler App in MATLAB (The MathWorks Inc., 2022a). We applied rectangular annotations of two categories: branch and overlap. Then, concordance between the automatic software and manual annotations was assessed, and precision, recall, and  $F_1$ -measure were computed. In case (2), we annotated the images using the Image Labeler App in MATLAB, where we labeled the areas surrounding every intersection as one of three categories: 1) main filament, 2) branch filament, or 3) passing filament (i.e., part of an overlap). Since assessing the correctness of the intersection arm matching makes sense only in the case of a correctly identified branch or overlap (i.e., true positives), this step was applied only to such cases. Consequently, we quantified the percentage of correct assignments.

## 3. Results

The following section presents our results. In detail, Sections (3.1-3.3) summarize the results of our quantitative validation of the different processing steps of the mycelium analysis tool, Section 3.4 reports a

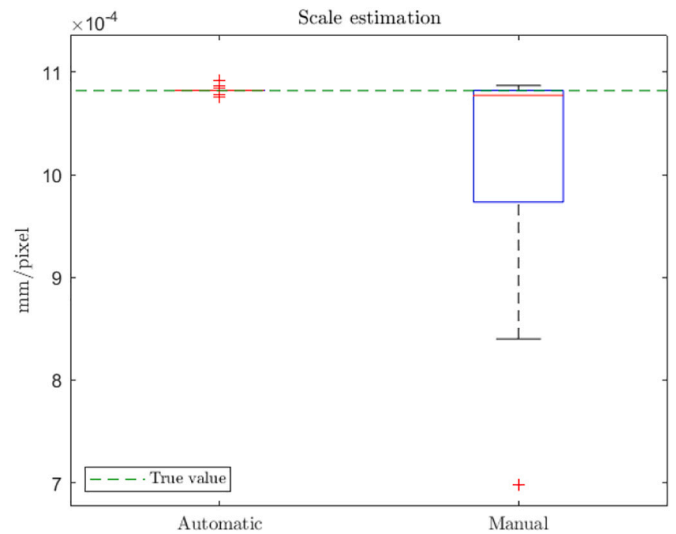


Fig. 10. Comparison between automatic and semi-automatic estimates mm/pixel relations.

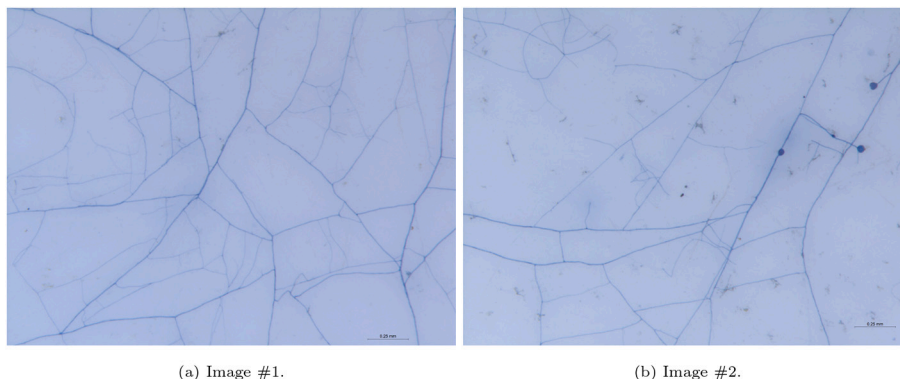


Fig. 9. Selected images for validation of branch and overlap detection.

comparison between the morphological graphs obtained with and without overlaps, and Section 3.5 reports some qualitative results deriving from the aforementioned validation procedures.

### 3.1. Scale bar reading

We found that the automatic method provides a highly precise estimation of the pixel-millimeter ratio, with only a couple of cases where one or two fewer or additional pixels were counted to the scale bar. Manually tracing the scale bar may lead to a less exact estimate. Fig. 10 shows boxplots of the mm/pixel ratios obtained considering 1) an automated segmentation and reading of the scale bar and 2) manual tracing. The results show that manual tracing is very accurate in the median case. However, the lower quartile represents an underestimation of 9%, and the lower whisker represents an underestimation of 21%.

### 3.2. Mycelium detection performance

Following the methodology described in Section 2.5.1, we computed the average precision, recall,  $F_1$ -measure, and MMC for the three different sets of fungal images. Results are shown in Table 2. In all cases, the 2<sup>nd</sup>-order Gaussian filter was applied with the set of angles  $\theta$  having a step size of 15°. The datasets contained diverse organisms and were acquired with diverse resolutions. Therefore, the standard deviation  $\sigma$  and the minimum branch length were varied. In the image set from (Cardini et al., 2020), all images were acquired at the same scale and represented the same organism. Therefore, the  $\sigma$  and minimum branch length parameters were the same for all images (i.e.,  $\sigma = 4$  pixels or 4.4  $\mu\text{m}$  and minimum branch length = 22  $\mu\text{m}$ ). For the image set from (Vidal-Diez de Ulzurrun et al., 2019), the image scales vary between samples, and the  $\sigma$ -parameter was varied ( $\sigma = [2.5, 3.5]$  pixels, step size = 0.5) and the minimum branch length was set to 3 pixels for all samples. In some images, the border of the petri dish was visible, and the effect of this border was removed with an ad-hoc masking operation. Finally, the image set from (Lopez-Molina et al., 2015) contained a variety of scales and species, and therefore, the  $\sigma$ -parameter was varied ( $\sigma = [1, 6]$  pixels, step size = 0.5). Instead, the minimum branch length was always set to 5 pixels. After visual inspection, these parameters' values were selected based on perceived output adequacy.

Fig. 11 shows the concordance between manual tracing-based length measures and length measures obtained by applying the detection algorithm. Fig. 11a shows that the length is vastly over-estimated for some images, and consequently, the concordance is poor. From a closer inspection, all these images had important chunks of dirt and accumulations of staining ink. Therefore, we applied a filter to remove all components not connected to a reference. The resulting comparison between manual tracing and automatic measurement is reported in Fig. 11b, and the resulting estimate of Lin's Concordance Coefficient ( $\hat{\rho}_c$ ) is 0.96, with a 95% lower confidence interval of 0.91, which may be interpreted as a moderate level of agreement (McBride, 2005).

### 3.3. Separation of filaments

The separation was tested following the method described in Section 2.5.2 on image #1 (Fig. 9a) and image #2 (Fig. 9b). Table 3 summarizes the branch/overlap identification outcomes obtained directly from the mycelium binary skeleton matrix from a relatively clean image containing 160 branches or anastomoses and 20 overlapping points. In eq.

**Table 2**  
Detection performance on diverse fungal image sets.

| Image set reference                  | Precision | Recall | $F_1$ | MMC  |
|--------------------------------------|-----------|--------|-------|------|
| Cardini et al. (2020)                | 0.82      | 0.92   | 0.88  | 0.88 |
| Vidal-Diez de Ulzurrun et al. (2019) | 0.96      | 0.92   | 0.94  | 0.94 |
| Lopez-Molina et al. (2015)           | 0.88      | 0.89   | 0.88  | 0.88 |

(3), the distance threshold  $D_T$  was set to 13  $\mu\text{m}$ , the angle threshold  $\theta_T$  to 45°, and  $\Theta_{switch}$  described in Section 2.4.1 was set to 30°.

Spurious filament detection substantially affects the number of perceived branches. It is also a frequent occurrence that filaments at the edge of the image are not detected due to boundary effects, causing the omission of branches. The evaluation described in Section 2.5.2 was performed while ignoring parts of the output with dirt or boundary effects to conduct an isolated performance assessment of the process after skeletonization. This assessment was applied to two selected images, image #1 (same as in Table 3) and image #2, which was considered more challenging. The results are reported in Table 4.

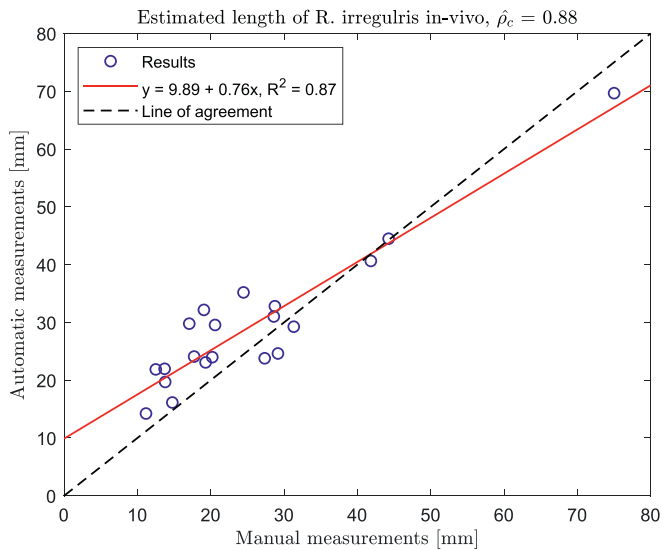
Besides correctly assigning whether an intersection point is a branch, an anastomosis point, or a filament overlapping, it is important to correctly connect the arms, especially in filament overlappings. In this context, 139 branches and 35 overlappings were annotated, and the concordance is reported in Table 5. Two wrongly matched overlappings are studied in detail in Figs. (12a, 12d).

Fig. 12 shows a selection of faulty assessments obtained through the algorithm. In Fig. 12a, the direction vectors related to the BP on the left-hand side are assigned based on the ridge prominence values (the steerable filter response), which differ marginally in this case. However, since the wrong filament is assigned at the branch-out filament, the result is a faulty matching. In Fig. 12b, a thinner filament appears to pass under a thicker point where a thicker hypha branches. The middle BP is matched with the BP on the right-hand side, although the BP on the left-hand side is more suitable. This type of overlapping multi-structures was out of scope in this study. Fig. 12c shows another example of a multi-structure, which is more extended. In this case, one filament goes from the top left corner to the bottom right corner, and another enters the Figure below the top left corner and exits above the bottom right corner. The overlaying filament is connected to two other filaments through branching or anastomosis. Consequently, the two BPs with branch-out vectors related to the underlying filament are separated by the other two BPs. In Fig. 12d, two filaments overlap each other, and the two resulting BPs are close enough to each other to satisfy the criterion in (3) based on the distance. The three vectors are, in both cases, separated by roughly equal angles, and there is no clear difference in ridge prominence. In this particular case, the branch-out arm label is assigned to diverse filaments, and consequently, the match gets wrong.

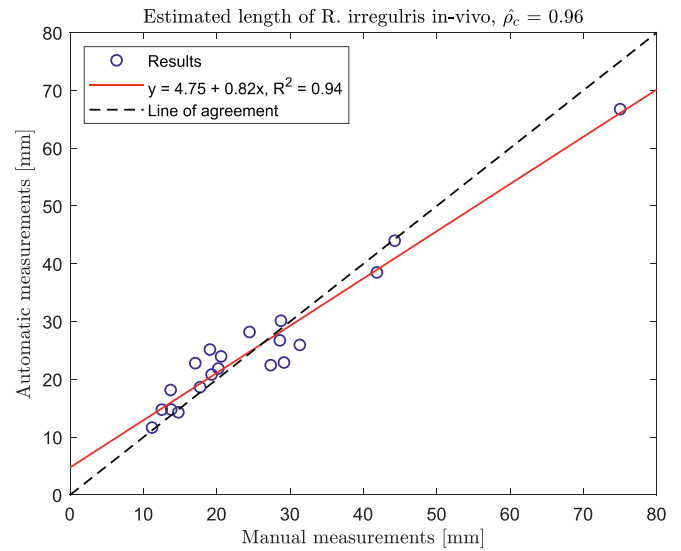
### 3.4. Comparison between the morphological graph with and without overlaps

We selected six images from the set of images in Cardini et al. (2020) that were relatively clean with few strange objects. We generated the morphological graph with overlaps as described in Section 2.4.5 and the morphological graph without overlaps as detailed in Section 2.4.2. Fig. 13 shows an example of these two types of representation.

In the graph shown in Fig. 13a, nodes may have degrees 1–4: nodes with degree 1 are tips, nodes with degree 2 are branch nodes sharing two edges with a neighbor, nodes with degree 3 represent regular branch nodes and nodes with degree 4 derive from quasi-perpendicular overlaps. The branch nodes could be either real biological branch nodes or be part of an overlap. By contrast, in the graph shown in Fig. 13b, all nodes have degree 1 or 3. Dikec et al. (2020) estimated the number of actual biological nodes using the formula  $N = N^g(1 - r)$  where  $N$  is the number of biological nodes,  $N^g$  is the number of geometrical nodes.  $r$  was then estimated with  $r = \frac{N_b^g}{N^g}$ , where  $N_b^g$  was the number of geometrical branch nodes within the distance  $n$  from each other. If we consider branch nodes in the morphological graph with overlaps to represent a direct estimation of the set  $N$ , we can compute  $r = 1 - \frac{N}{N^g}$ . Table 6 shows a) the number of geometric branch nodes  $N^g$  in the morphological graph without overlaps and b) the number of branch nodes in the morphological graph with overlaps, which is an estimate of the number of biological nodes. Consequently, the mean  $r$  corresponds to 20% (also



(a) Without removal of non-connected components.



(b) With removal of non-connected components.

Fig. 11. Comparison of automatic and manual estimates total mycelium length in *R. irregularis* samples from (Cardini et al., 2020).

Table 3

Identification of branches and crossing, including where mycelium detection errors are present.

|                 | True Positive | False Positive | False Negative | Precision | Recall | F1   |
|-----------------|---------------|----------------|----------------|-----------|--------|------|
| Branch img #1   | 155           | 33             | 5              | 0.83      | 0.97   | 0.89 |
| Crossing img #1 | 17            | 1              | 3              | 0.94      | 0.85   | 0.89 |

20% in (Dikec et al., 2020)), and the standard deviation  $\sigma_r = 6\%$  (1.6% in (Dikec et al., 2020)).

Another difference observed between the two morphological graphs is that the edges become fewer and longer when the geometrical nodes are removed. An average reduction of 15% in the number of edges is found among the six tested images. Fig. 14 shows the histograms of the stacked edge lengths from the two types of graph representation. In Fig. 14a, there are numerous very short edges, and the number of edges in each bin declines rapidly. In Fig. 14b, the longer edges make up a larger part of the histogram, and the decline in frequency is smoother.

### 3.5. Qualitative results

Considering the above outcomes, several interesting qualitative results can be derived (Fig. 15). From the binary skeleton matrix obtained from the procedure described in Section 2.3, we show in Section 3.2 that the total mycelium length in the sample can be accurately estimated. A common measure of mycelium density in a 2D sample is  $mm/mm^2$  (Avio et al., 2006; Cardini et al., 2020). A sliding window moving average filtering operation on the binary skeleton matrix can be performed to obtain the local mycelium density distribution over the sample. The output of this operation is highly dependent on the window size: a window that is too small will only reveal an increased density around the filaments, and a window that is too large will just reveal a blob in the middle of the image. We found that, for our samples of *R. irregularis*, square windows of sizes between  $0.19mm^2$  and  $0.42mm^2$  yield meaningful outputs. In Fig. 15a, we apply a window with a size corresponding to  $0.29mm^2$ . Given a set of thin binary filaments, each with a unique ID, eq. (1) can be applied to each filament to get its length (Fig. 15b). To visualize the network structure, we can consider two diverse representations (Fig. 15c & 15d). In Fig. 15c, the separated hyphae have been

Table 4

Identification of branches and crossing, excluding where mycelium detection errors are present.

|                  | True Positives | False Positives | False Negatives | Precision | Recall | F1   |
|------------------|----------------|-----------------|-----------------|-----------|--------|------|
| Branches img #1  | 155            | 1               | 1               | 0.99      | 0.99   | 0.99 |
| Crossings img #1 | 17             | 1               | 1               | 0.94      | 0.94   | 0.94 |
| Branches img #2  | 65             | 3               | 0               | 0.96      | 1      | 0.98 |
| Crossings img #2 | 18             | 1               | 2               | 0.95      | 0.90   | 0.92 |

colored based on their relative ridge prominence. In Fig. 15d, each isolated filament is instead represented as a node in the graph, and the node is placed over the filament. The node color and size are set based on the relative ridge prominence and node degree, respectively.

## 4. Discussions

### 4.1. Filament detection and length estimation

This paper presents a method to detect the filamentous structure of mycorrhizal networks in 2D images. We assess the algorithm's performance in terms of precision, recall,  $F_1$ -score, and MMC of the detected skeleton representation, and we assess the algorithm's viability for automatically measuring mycelium length. To estimate the mycelium length, it is necessary to know the scale of the image, and any error in the scale estimation is directly multiplied by the detection error. The proposed strategy of thresholding the scale bar relies on the assumption that the scale bar is distinctly the darkest object in the image: if this assumption does not hold, the result is fatal. An improvement might consist of using a scale bar with a specific RGB channel and applying the segmentation to that channel to guarantee correct functionality.

Table 5

Matching accuracy.

|          | Correct | Incorrect | Accuracy |
|----------|---------|-----------|----------|
| Branch   | 132     | 7         | 95%      |
| Crossing | 31      | 4         | 89%      |



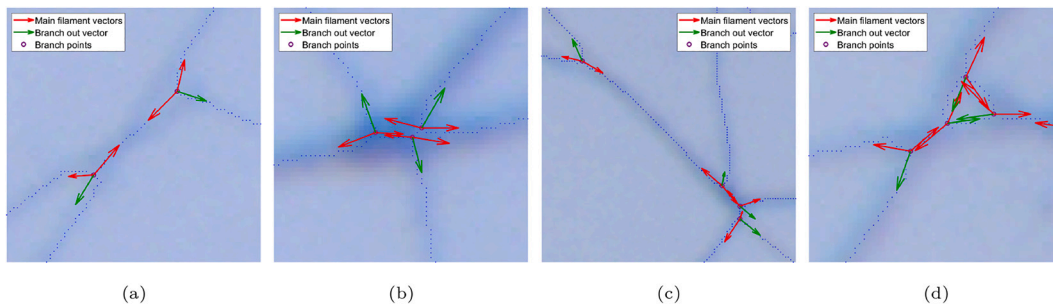


Fig. 12. Examples of situations too complex for the algorithm.

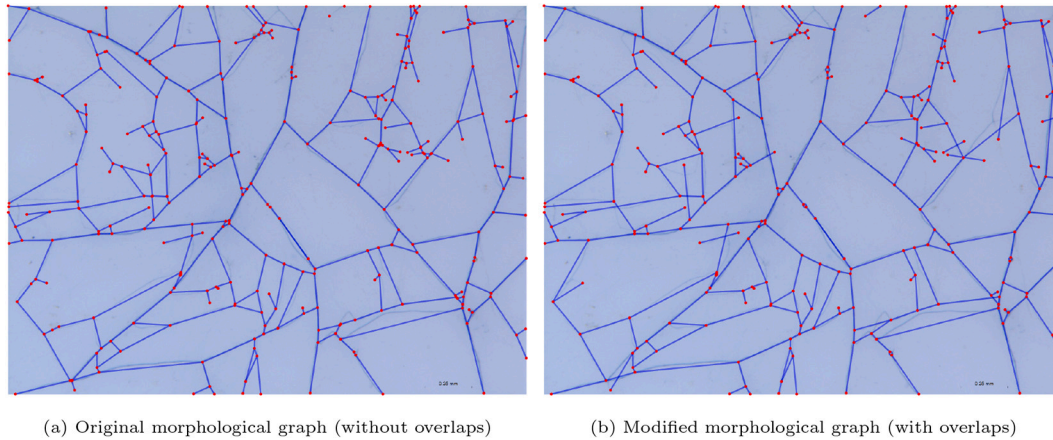


Fig. 13. Example of morphological graphs: original version (i.e., with no overlaps) (a), and modified version (i.e., with overlaps) (b).

Table 6

Difference in number of branch nodes in the morphological graphs without ( $N^{\#}$ ) and with ( $N$ ) overlaps.

| Image | $N^{\#}$ | $N$ | $r$  |
|-------|----------|-----|------|
| #1    | 218      | 186 | 0.15 |
| #2    | 181      | 134 | 0.26 |
| #3    | 88       | 75  | 0.15 |
| #4    | 80       | 67  | 0.16 |
| #5    | 80       | 58  | 0.28 |
| #6    | 185      | 148 | 0.2  |

The algorithms are evaluated for thin objects. The presence of thicker objects may cause severe problems already in the skeletonization step, as pointed out by (Dirnberger et al., 2015). As previously discussed in Sections (1 and 2.3), the skeletonization using the medial axis transform tends to yield short spurious branches. By setting a branch length threshold, there is the risk that some short branches are wrongly removed. In these cases, such short branches will not substantially affect the length estimate, and while it may lead to some branches being ignored, the branches will not be ignored anymore once they have grown.

As observed in Section 3.2, the mycelium detection algorithm performed similarly to the state-of-the-art algorithms on the data sets from (Lopez-Molina et al., 2015; Vidal-Diez de Ulzurrun et al., 2019), without

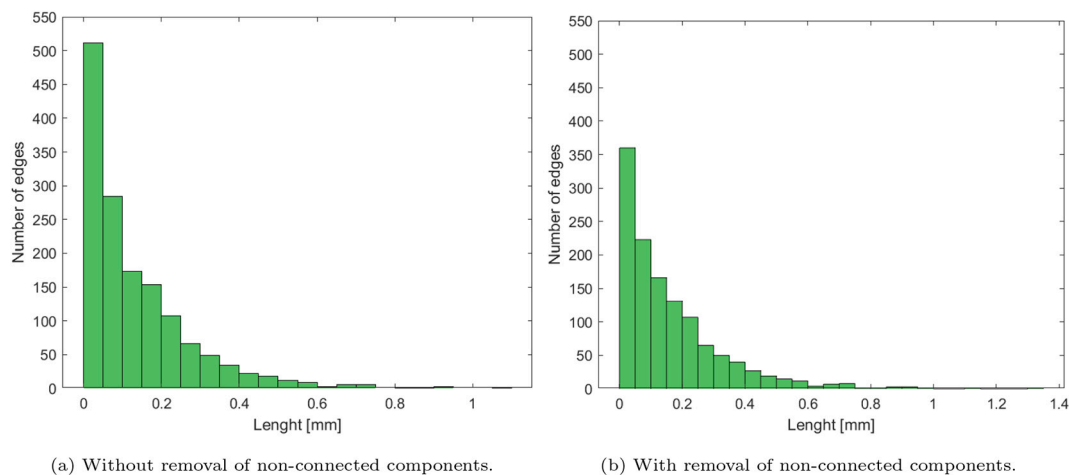
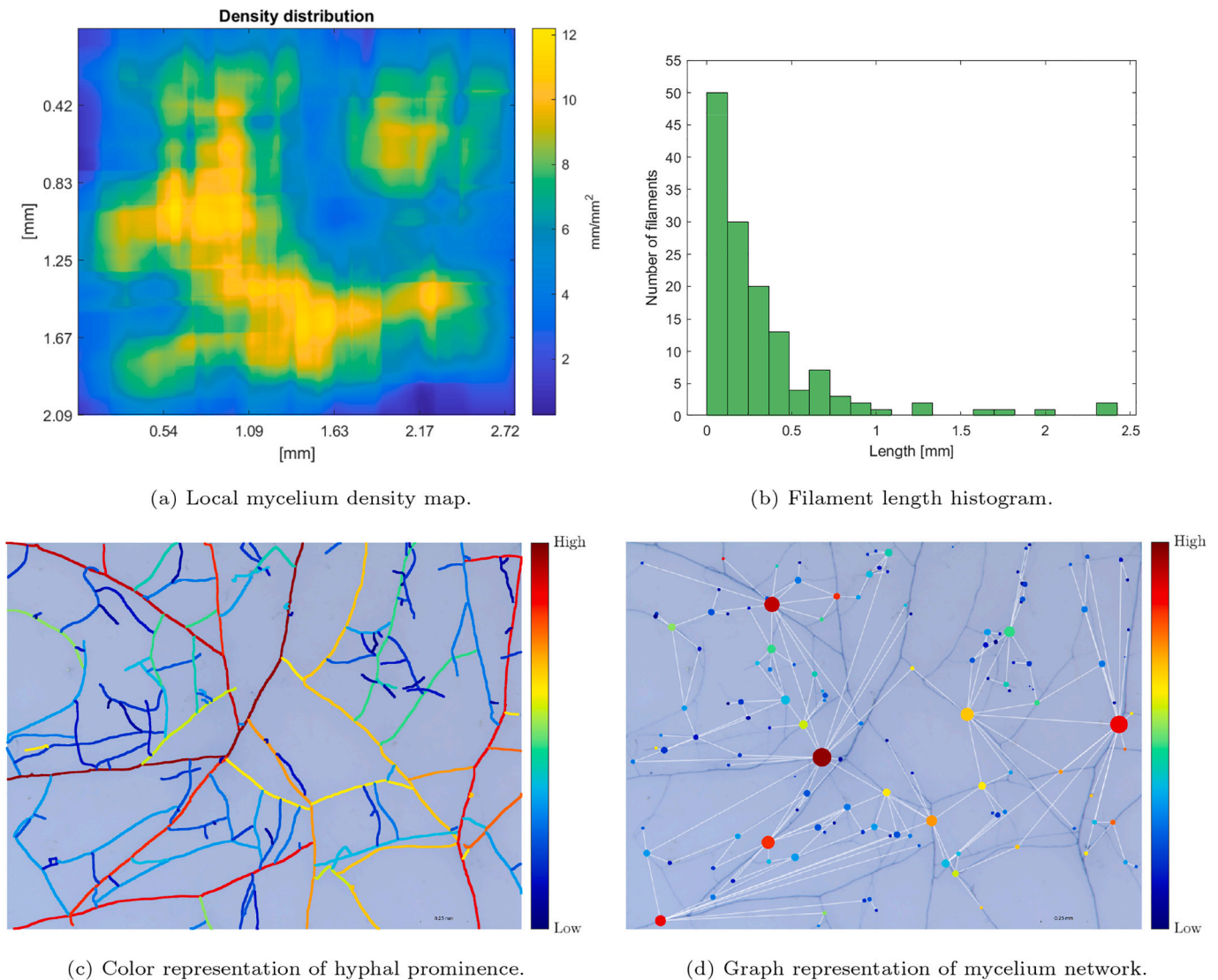


Fig. 14. Comparison of the resulting edge lengths in the two types of morphological graphs.



**Fig. 15.** Qualitative results. a) Maps the local mycelium density around different points on the pixel level. b) Length histogram of the segmented filaments in (c). c) Filaments are colored based on their ridge prominence (i.e., steerable ridge filter output). d) Graph representation of (c) with the same color coding. Edges imply that the filaments are connected, and node size depends on node degree (the larger the node, the higher the connectivity).

additional filtration (aside from border removal). An  $F_1$ -score and MMC of 0.89 on fungal images in (Lopez-Molina et al., 2015; Wang et al., 2019), and an average  $F_1$ -score of 0.92 and MMC of 0.93 were reported in (Vidal-Diez de Ulzurrun et al., 2019). These results suggest that the detection performance of our method is slightly better than plain SOAGK and slightly worse than normalized adaptive SOAGK. However, the difference is not large enough for a decisive judgment, especially since the error occurring in the human annotation is not considered.

In Vidal-Diez de Ulzurrun et al. (2019), two parameters, the binarization thresholds  $\tau_1$  and  $\tau_2$ , had to be tuned, and in Wang et al. (2019) the parameter  $\kappa$ , used for adjusting the threshold, was fine-tuned. On the contrary, with our technique, the only parameters needed to be tuned are directly related to the scale of the objects of interest, making the tuning more intuitive for our specific application. A future perspective is to find an ideal relation between these parameters and the image scale (automatically estimated from (2.2)) and the species (usually known) to obtain a fully automated procedure. In this study, this is not addressed due to the lack of diverse images with known scales.

Although an exhaustive study of the algorithm's performance in ridge detection benchmarks was not conducted, the results suggest that the proposed detection algorithm performs on par with the state-of-the-

art algorithms on fungal images with a more intuitive parameterization for the considered application. Differently, the images from (Cardini et al., 2020) are much dirtier, and to obtain a satisfactory length estimate, applying additional filtering is therefore required. The detection algorithm is not specific enough to discard uninteresting objects with color and geometry similar to those of interest. Thus, for the data set deriving from (Cardini et al., 2020), it is necessary to apply an ad-hoc criterion for removing such objects. Nevertheless, the results suggest that if the sample is sufficiently clean, the proposed method can replace manual measurements of mycelium length.

Future perspectives for a more versatile removal of these components might include adding a user interface for easily marking desired components or the development of a more specific detector. Deep learning is a paradigm that has been shown to outperform other methods in several domains. Indeed, (Xu et al., 2021) developed a Convolutional Neural Network-based segmentation model and compared its performance against other methods, including the PCT method developed in (Obara et al., 2012), finding a substantial improvement.

**Table 7**  
Identification of branches and crossing, without the matching algorithm.

|                     | True Positive | False Positive | False Negative | Precision | Recall | F1   |
|---------------------|---------------|----------------|----------------|-----------|--------|------|
| Branches<br>img #1  | 155           | 59             | 4              | 0.72      | 0.97   | 0.83 |
| Crossings<br>img #1 | 3             | 0              | 17             | 1         | 0.15   | 0.26 |

#### 4.2. Filament separation

This paper introduces a novel method to identify filaments crossing each other automatically. We first analyze each branch point and extract the branch point's arms to remove false positive branch points and to obtain a local approximation of the arms' geometry and prominence. Then, the following is done: a) a morphological graph representation is created, b) the main filament and branching filament are identified, and c) the branch points, likely to constitute a crossing based on their position and local geometry, are matched together. This two-step procedure allows new representations of mycelium samples, reducing the connectivity overestimation reported in previous works (Aguilar-Trigueros et al., 2022). Without considering this matching, the case reported in Table 3 would be the same as shown in Table 7, implying that naively extracting the morphological graph would, in this case, overestimate the connectivity by 38% instead of 20%. On the contrary, the overestimation with crossing detection would be <1% if the errors related to detection error are disregarded, whereas otherwise, it would be equal to 19%.

It is worth mentioning that edge crossings in the image sets analyzed in (Aguilar-Trigueros et al., 2022; Lasser and Katifori, 2017; Lopez-Molina et al., 2015; Vidal-Diez de Ulzurrun et al., 2019) were rare, so the added value of a fully automatic identification procedure of such edges is limited. Instead, in the images of *R. irregularis* from Cardini et al. (2020), the frequency of such edges was higher. In the first manually annotated sample, we found 160 branches and 20 crossings (89% branches, 11% crossings), and in the second image, we found 65 branches and 20 crossings (76% branches and 24% crossings). Although not perfect, the identification of crossing can improve the interpretation of the mycelium network structure in 2D mycelium images. The results presented in Section 2.4.5 suggest an average reduction in connectivity of 20% between the conventional morphological graph and the morphological graph with overlaps. This outcome is consistent with the estimate proposed in (Dikec et al., 2020), who estimated that around 20% of the branch nodes not real biological nodes.

The approach is not scale-invariant, and it is crucial that, for each structure, there is a sufficient number of pixels for estimating local geometries and ridge prominence. The scale range in which the algorithm may work well was not investigated. However, a scale at which the side of a pixel corresponded to 1.1  $\mu\text{m}$  worked well for samples of *R. irregularis* with a nominal hyphal width of 7.4–19  $\mu\text{m}$  (University of Kansas, 2023).

The matching algorithm gives some errors typically when a) the overlapping filaments have similar prominence and b) when filament crossings coincide with branching/anastomosis. In the first case, the performance may be improved by utilizing more information about the filaments, which could be obtained by, e.g., considering the grayscale or ridge values of the filament in the BP's neighborhood derived through the width estimation procedure proposed in (Obara et al., 2012). In the second case, there are some possible performance improvements to consider. For instance, in the examples shown in Figs. (12b) and (12c), the error might be solved by removing the condition of mutual closeness and permitting to match BPs separated by one or more other BPs. A more robust assessment of arm similarity needs to be performed to remove this condition, possibly demanding the extraction of more arm features. Indeed, there is a vast range of potentially complex situations, and

guaranteeing the correct assessment of overlapping structures in 2D is challenging. Currently, the tool shows higher performances for samples that are not too dense. However, defining a threshold density is difficult since, for instance, in the densest areas of Image 1 (Fig. 15a), there are no apparent errors, while in other parts there are, suggesting that challenging conditions occur randomly rather than at a certain density. Furthermore, it is important that most of the colony is within the image: for instance, in Fig. 15c, the red filament on the right-hand side is not divided since it goes very close to the border of the image and, therefore, it appears to be connected to some filaments far from it, as shown in Fig. 15d. Strategies previously proposed by other authors might potentially improve performance. For instance, Lasser and Katifori (2017) provided a GUI for manually fixing the errors. Given such an interface and the computed branch angle and difference in ridge prominence, the user might be directed towards where the algorithm is less confident and improve the procedure. As proposed in (Dikec et al., 2020; Zhang et al., 2019), tracking the filaments' growth from an early stage may facilitate keeping them apart and indicate with a higher degree of certainty that the filaments are actually separate. Tracking the filaments' growth over time may also be a way to enable automated differentiation between branch points and anastomosis points. However, this does require multiple time image acquisitions and, consequently, an increased experimental effort.

As mentioned in Section 4.1, performances may be improved by exploiting machine learning. Specifically, (Guo et al., 2021) extracted spherical patches around points of interest in the 3D volumetric images and used a deep learning-based model to classify the point of interest: if the point of interest was a cross-over point/overlap, the incoming filaments were separated. With respect to the approach proposed in (Guo et al., 2021), our approach is similar for hierarchically starting by creating an overview and then locally analyzing the points of interest. Meanwhile, (Guo et al., 2021) analyzes all potential points of interest by considering a patch with a fixed radius, which may be advantageous in 3D thanks to the volumetric information. We locally analyze the points of interest considering their respective filament angles in order to be able to identify overlaps with geometrical branch points far from each other (i.e., very acute intersection angle) in two dimensions.

## 5. Conclusions

This paper presents a highly automated analysis tool for mycelium network characterization. Specifically, we present a filament detection algorithm based on the ridge skeletonization method that also solves the problem of identifying and handling overlapping filaments in a 2D representation of a 3D object. The approach combines steerable filtering with first-order local adaptive thresholding. It processes the images in six steps: automated scale estimation, mycelium detection, branch point assessment, building of the morphological graph, identification of crossing pairs, and rebuilding of the filament graph representation. Crossings of overlapping filaments are identified by assessing the local geometry and ridge prominence around each branch point in the skeleton representation of the filamentous structure. We tested our algorithm on manually annotated real images of filamentous fungi. We show that 1) our tool can be used to measure the length of mycelium in an image, and the result concurs moderately with those obtained through manual tracing (Section 4.1), 2) filament detection is on par with the state of the art techniques in terms of precision and recall as discussed in Section 4.1, and 3) the identification and handling of overlapping filaments perform well for simple crossings as detailed in Section 4.2. These results suggest that the proposed method can provide a more accurate representation of the filament structure of mycorrhizal fungi and facilitate the analysis in laboratory conditions for a better characterization of complex fungal networks.



## CRedit authorship contribution statement

**Oscar Sten:** Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation. **Emanuela Del Dottore:** Writing – original draft, Supervision, Methodology, Conceptualization. **Nicola Pugno:** Writing – original draft, Supervision. **Barbara Mazzolai:** Writing – original draft, Supervision, Resources, Funding acquisition, Conceptualization.

## Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Barbara Mazzolai reports financial support was provided by European Research Council. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Source code and all data required to reproduce the presented results are available at: <https://gitlab.iiit.it/oscar.sten/mycelium-network-analysis-tool>

## Acknowledgments

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme Grant Agreement No. 101003304 (I-Wood). The authors would like to thank Guillermo Vidal-Diez de Ulzurrun for providing us with additional test data.

## References

- Aguilar-Trigueros, C.A., Boddy, L., Rillig, M.C., Fricker, M.D., Jan. 2022. Network traits predict ecological strategies in fungi. *ISME Commun.* 2 (1), 2. URL <https://www.nature.com/articles/s43705-021-00085-1>.
- Alaux, P., Zhang, Y., Gilbert, L., Johnson, D., Sep. 2021. Can common mycorrhizal fungal networks be managed to enhance ecosystem functionality? *PLANTS, PEOPLE, PLANET* 3 (5), 433–444. URL <https://onlinelibrary.wiley.com/doi/10.1002/ppp3.10178>.
- Avio, L., Pellegrino, E., Bonari, E., Giovannetti, M., Oct. 2006. Functional diversity of arbuscular mycorrhizal fungal isolates in relation to extraradical mycelial networks. *New Phytol.* 172 (2), 347–357. URL <https://onlinelibrary.wiley.com/doi/10.1111/j.1469-8137.2006.01839.x>.
- Babikova, Z., Gilbert, L., Bruce, T.J.A., Birkett, M., Caulfield, J.C., Woodcock, C., Pickett, J.A., Johnson, D., Jul. 2013. Underground signals carried through common mycelial networks warn neighbouring plants of aphid attack. *Ecol. Lett.* 16 (7), 835–843. URL <https://onlinelibrary.wiley.com/doi/10.1111/ele.12115>.
- Babikova, Z., Johnson, D., Bruce, T., Pickett, J., Gilbert, L., Jan. 2014. Underground allies: how and why do mycelial networks help plants defend themselves?: what are the fitness, regulatory, and practical implications of defence-related signaling between plants via common mycelial networks? *BioEssays* 36 (1), 21–26. URL <https://onlinelibrary.wiley.com/doi/10.1002/bies.201300092>.
- Barry, D.J., Chan, C., Williams, G.A., Jun. 2009. Morphological quantification of filamentous fungal development using membrane immobilization and automatic image analysis. *J. Ind. Microbiol. Biotechnol.* 36 (6), 787–800. URL <https://academic.oup.com/jimb/article/36/6/787/5993586>.
- Barry, D.J., Williams, G.A., Chan, C., May 2015. Automated analysis of filamentous microbial morphology with AnaMorf. *Biotechnol. Prog.* 31 (3), 849–852. URL <https://onlinelibrary.wiley.com/doi/10.1002/btpr.2087>.
- Bonfante, P., Genre, A., Jul. 2010. Mechanisms underlying beneficial plant–fungus interactions in mycorrhizal symbiosis. *Nat. Commun.* 1 (1), 48. URL <https://www.nature.com/articles/ncomms1046>.
- Bradley, D., Roth, G., Jan. 2007. Adaptive thresholding using the integral image. *J. Graphics Tools* 12 (2), 13–21. URL <https://www.tandfonline.com/doi/full/10.1080/2151237X.2007.10129236>.
- Brunk, M., Sputh, S., Doose, S., van de Linde, S., Terpitz, U., Dec. 2018. HyphaTracker: an ImageJ toolbox for time-resolved analysis of spore germination in filamentous fungi. *Sci. Rep.* 8 (1), 605. URL <http://www.nature.com/articles/s41598-017-19103-1>, doi:10.1038/836 s41598-017-19103-1.
- Cardini, A., Pellegrino, E., Del Dottore, E., Gamper, H.A., Mazzolai, B., Ercoli, L., May 2020. HyLength: a semi-automated digital image analysis tool for measuring the length of roots and fungal hyphae of dense mycelia. *Mycorrhiza* 30 (2–3), 229–242. URL <https://link.springer.com/10.1007/s00572-020-00956-w>.
- Chen, W., Liu, M., Zhan, Q., Tan, Y., Meijering, E., Radojevic, M., Wang, Y., Feb. 2021. Spherical-patches extraction for deep-learning-based critical points detection in 3D neuron microscopy images. *IEEE Trans. Med. Imaging* 40 (2), 527–538. URL <https://ieeexplore.ieee.org/document/9224681/>.
- De Ligne, L., Vidal-Diez de Ulzurrun, G., Baetens, J.M., Van den Bulcke, J., Van Acker, J., De Baets, B., Dec. 2019. Analysis of spatio-temporal fungal growth dynamics under different environmental conditions. *IMA Fungus* 10 (1), 7. URL <https://IMAFungus.biomedcentral.com/articles/10.1186/s43008-019-0009-3>.
- Dikec, J., Olivier, A., Bobée, C., D'Angelo, Y., Catellier, R., David, P., Filaine, F., Herbert, S., Lalanne, C., Lalucque, H., Monasse, L., Rieu, M., Ruprich-Robert, G., Véber, A., Chapeland-Leclerc, F., Herbert, E., Feb. 2020. Hyphal network whole field imaging allows for accurate estimation of anastomosis rates and branching dynamics of the filamentous fungus *Podospora anserina*. *Sci. Rep.* 10 (1), 3131. URL <https://www.nature.com/articles/s41598-020-57808-y>.
- Dirnberger, M., Kehl, T., Neumann, A., Nov. 2015. NEFI: network extraction from images. *Sci. Rep.* 5 (1), 15669. URL <https://www.nature.com/articles/srep15669>.
- Freeman, W., Adelson, E., Sep. 1991. The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell.* 13 (9), 891–906. URL <http://ieeexplore.ieee.org/document/93808/>.
- Fricker, M.D., Akita, D., Heaton, L.L., Jones, N., Obara, B., Nakagaki, T., Jun. 2017. Automated analysis of *Physarum* network structure and dynamics. *J. Phys. D. Appl. Phys.* 50 (25), 254005. URL <https://iopscience.iop.org/article/10.1088/1361-6463/aa72b9>.
- Gilbert, L., Johnson, D., 2017. Plant–plant communication through common mycorrhizal networks. In: *Advances in Botanical Research*, vol. 82. Elsevier, pp. 83–97. URL <https://linkinghub.elsevier.com/retrieve/pii/S0065229616300878>.
- Guo, C., Liu, M., Guan, T., Chen, W., Wen, H., Zeng, T., Wang, Y., 2021. Crossover structure separation with application to neuron tracing in volumetric images. *IEEE Trans. Instrum. Meas.* 70, 1–13. URL <https://ieeexplore.ieee.org/document/9399456/>.
- Jacob, M., Unser, M., Aug. 2004. Design of steerable filters for feature detection using canny-like criteria. *IEEE Trans. Pattern Anal. Mach. Intell.* 26 (8), 1007–1019. URL <http://ieeexplore.ieee.org/document/1307008/>.
- Kaufhold, J.P., Tsai, P.S., Blinder, P., Kleinfeld, D., Aug. 2012. Vectorization of optically sectioned brain microvasculature: learning aids completion of vascular graphs by connecting gaps and deleting open-ended segments. *Med. Image Anal.* 16 (6), 1241–1258. URL <https://linkinghub.elsevier.com/retrieve/pii/S1361841512000850>.
- Lasser, J., Katifori, E., Dec. 2017. NET: a new framework for the vectorization and examination of network data. *Source Code Biol. Med.* 12 (1), 4. URL <http://scfbm.biomedcentral.com/articles/10.1186/s13029-017-0064-3>.
- Ledoux, C., Chapeland-Leclerc, F., Ruprich-Robert, G., Bobée, C., Lalanne, C., Herbert, R., David, P., Jul. 2022. Prediction and experimental evidence of the optimisation of the angular branching process in the thallus growth of *Podospora anserina*. *Sci. Rep.* 12 (1), 12351. URL <https://www.nature.com/articles/s41598-022-16245-9>.
- Ledoux, C., Bobée, C., Cabet, V., David, P., Filaine, F., Hachimi, S., Lalanne, C., Ruprich-Robert, G., Herbert, R., Chapeland-Leclerc, F., Feb. 2024. Characterization of spatio-temporal dynamics of the constrained network of the filamentous fungus *Podospora anserina* using a geomatics-based approach. *PLoS One* 19 (2), e0297816. URL <https://dx.plos.org/10.1371/journal.pone.0297816>. <https://doi.org/10.1371/journal.pone.0297816>.
- Liu, G., Haralick, R.M., Oct. 2002. Optimal matching problem in detection and recognition performance evaluation. *Pattern Recogn.* 35 (10), 2125–2139. URL <https://linkinghub.elsevier.com/retrieve/pii/S0031320301002047>.
- Liu, S., Zhang, D., Song, Y., Peng, H., Cai, W., Nov. 2018. Automated 3-D neuron tracing with precise branch erasing and confidence controlled back tracking. *IEEE Trans. Med. Imaging* 37 (11), 2441–2452. URL <https://ieeexplore.ieee.org/document/8354803/>.
- Lopez-Molina, C., De Baets, B., Bustince, H., Apr. 2013. Quantitative error measures for edge detection. *Pattern Recogn.* 46 (4), 1125–1139. URL <https://linkinghub.elsevier.com/retrieve/pii/S003132031200461X>.
- Lopez-Molina, C., Vidal-Diez de Ulzurrun, G., Baetens, J., Van den Bulcke, J., De Baets, B., Nov. 2015. Unsupervised ridge detection using second order anisotropic Gaussian kernels. *Signal Process.* 116, 55–67. URL <https://linkinghub.elsevier.com/retrieve/pii/S0165168415001255>.
- Manubens-Gil, L., Zhou, Z., Chen, H., Ramanathan, A., Liu, X., Liu, Y., Bria, A., Gillette, T., Ruan, Z., Yang, J., Radojević, M., Zhao, T., Cheng, L., Qu, L., Liu, S., Bouchard, K.E., Gu, L., Cai, W., Ji, S., Roysam, B., Wang, C.-W., Yu, H., Sironi, A., Iascone, D.M., Zhou, J., Bas, E., Conde-Sousa, E., Aguiar, P., Li, X., Li, Y., Nanda, S., Wang, Y., Muresan, L., Fua, P., Ye, B., He, H.-Y., Staiger, J.F., Peter, M., Cox, D.N., Simonneau, M., Oberlander, M., Jefferis, G., Ito, K., Gonzalez-Bellido, P., Kim, J., Rubel, E., Cline, H.T., Zeng, H., Nern, A., Chiang, A.-S., Yao, J., Roskams, J., Livesey, R., Stevens, J., Liu, T., Dang, C., Guo, Y., Zhong, N., Tourassi, G., Hill, S., Hawrylycz, M., Koch, C., Meijering, E., Ascoli, G.A., Peng, H., Jun. 2023. BigNeuron: a resource to benchmark and predict performance of algorithms for automated tracing of neurons in light microscopy datasets. *Nat. Methods* 20 (6), 824–835. URL <https://www.nature.com/articles/s41592-023-01848-5>.
- Martin, D., Fowlkes, C., Malik, J., May 2004. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Trans. Pattern Anal. Mach. Intell.* 26 (5), 530–549. URL <http://ieeexplore.ieee.org/document/1273918/>.
- Martínez-Galicia, E., Fernanda Flores Enríquez, A., Puga, A., Gutiérrez-Medina, B., Oct. 2023. Analysis of the emerging physical network in young mycelia. *Fungal Genet. Biol.* 168, 103823. URL <https://linkinghub.elsevier.com/retrieve/pii/S1087184523000543>.



- Mathew, R.P., 2018. MATLAB code for computing Lin's Concordance Correlation Coefficients including confidence intervals. URL [https://github.com/robertpetermathew/f\\_CCC](https://github.com/robertpetermathew/f_CCC).
- McBride, G.B., 2005. A Proposal for Strength-of-Agreement Criteria for Lin's Correlation Coefficient. URL <https://www.medcalc.org/download/pdf/McBride2005.pdf>.
- McGarigal, K., Cushman, S., Ene, E., 2023. FRAGSTATS v4: Spatial Pattern Analysis Program for Categorical Maps. URL <https://www.fragstats.org>.
- Obara, B., Grau, V., Fricker, M.D., Sep., 2012. A bioimage informatics approach to automatically extract complex fungal networks. *Bioinformatics* 28 (18), 2374–2381. URL <https://academic.oup.com/bioinformatics/article/28/18/2374/243031>.
- Otsu, N., Jan. 1979. A threshold selection method from gray-level histograms. *IEEE Trans. Syst. Man Cybern.* 9 (1), 62–66. URL <http://ieeexplore.ieee.org/document/4310076/>.
- Pain, C., Kriechbaumer, V., Kittelmann, M., Hawes, C., Fricker, M., Feb. 2019. Quantitative analysis of plant ER architecture and dynamics. *Nature Communications* 10 (1), 984. URL <https://www.nature.com/articles/s41467-019-08893-9>. <https://doi.org/10.1038/s41467-019-08893-9>.
- Pang, J., 2022. Steerable Filter. URL <https://www.mathworks.com/matlabcentral/fileexchange/44956-steerable-filter>.
- Sachs, C.C., Koepff, J., Wiechert, W., Grünberger, A., Nöh, K., Dec. 2019. mycelysio – high-throughput analysis of *Streptomyces* mycelium live cell imaging data. *BMC Bioinform.* 20 (1), 452. URL <https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-019-3004-1>.
- Sánchez-Orellana, G., Casas-Flores, S., Gutiérrez-Medina, B., Feb. 2019. Automated, continuous video microscopy tracking of hyphal growth. *Fungal Genet. Biol.* 123, 25–32. URL <https://linkinghub.elsevier.com/retrieve/pii/S1087184518302573>.
- Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., Tinevez, J.-Y., White, D.J., Hartenstein, V., Eliceiri, K., Tomancak, P., Cardona, A., Jul. 2012. Fiji: an open-source platform for biological-image analysis. *Nat. Methods* 9 (7), 676–682. URL <http://www.nature.com/articles/nmeth.2019>.
- Sharma, M., Saha, O., Sriraman, A., Hebbalaguppe, R., Vig, L., Karande, S., Jul. 2017. Crowdsourcing for chromosome segmentation and deep classification. In: In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). IEEE, Honolulu, HI, USA, pp. 786–793. URL <http://ieeexplore.ieee.org/document/8014843/>.
- Shokouh, G.-S., Magnier, B., Xu, B., Montesinos, P., Jul. 2021. Ridge detection by image filtering techniques: a review and an objective analysis. *Pattern Recog. Image Anal.* 31 (3), 551–570. URL <https://link.springer.com/10.1134/S1054661821030226>.
- Simard, S.W., Beiler, K.J., Bingham, M.A., Deslippe, J.R., Phillip, L.J., Teste, F.P., Apr. 2012. Mycorrhizal networks: mechanisms, ecology and modelling. *Fungal Biol. Rev.* 26 (1), 39–60. URL <https://linkinghub.elsevier.com/retrieve/pii/S1749461312000048>.
- Soille, P., Vogt, P., Mar. 2009. Morphological segmentation of binary patterns. *Pattern Recog. Lett.* 30 (4), 456–459. URL <https://linkinghub.elsevier.com/retrieve/pii/S0167865508003267>.
- Song, Y.Y., Zeng, R.S., Xu, J.F., Li, J., Shen, X., Yihdego, W.G., Oct. 2010. Interplant communication of tomato plants through underground common mycorrhizal networks. *PLoS One* 5 (10), e13324. URL <https://dx.plos.org/10.1371/journal.pone.0013324>.
- Sten, O., Del Dottore, E., Raffaele, G., Ronzan, M., Pugno, N.M., Mazzolai, B., 2023. Sporify: An automated tool to quantify spores in Z-stacked 3D Samples. In: Meder, F., Hunt, A., Margheri, L., Mura, A., Mazzolai, B. (Eds.), *Biomimetic and Biohybrid Systems*, Vol. 14158. Springer Nature Switzerland, Cham, pp. 178–192 series Title: *Lecture Notes in Computer Science*. URL [https://link.springer.com/10.1007/978-3-031-39504-8\\_12](https://link.springer.com/10.1007/978-3-031-39504-8_12), doi:[https://doi.org/10.1007/978-3-031-39504-8\\_12](https://doi.org/10.1007/978-3-031-39504-8_12). series Title: *Lecture Notes in Computer Science*.
- Tan, Y., Liu, M., Chen, W., Wang, X., Peng, H., Wang, Y., Apr. 2020. DeepBranch: deep neural networks for branch point detection in biomedical images. *IEEE Trans. Med. Imaging* 39 (4), 1195–1205. URL <https://ieeexplore.ieee.org/document/8861356/>.
- The MathWorks Inc, 2022a. Image Processing Toolbox. URL [https://se.mathworks.com/products/image.html?s\\_tid=srchtitle\\_image\\_1](https://se.mathworks.com/products/image.html?s_tid=srchtitle_image_1).
- The MathWorks Inc, 2022b. MATLAB (R2022a). URL <https://www.mathworks.com>.
- University of Kansas, 2023. The International Collection of (Vesicular) Arbuscular Mycorrhizal Fungi: *Rhizophagus irregularis*. URL <https://invam.ku.edu/irregularis>.
- Vidal-Diez de Ulzurrun, G., Baetens, J., Van den Bulcke, J., Lopez-Molina, C., De Windt, I., De Baets, B., Nov. 2015. Automated image-based analysis of spatio-temporal fungal dynamics. *Fungal Genet. Biol.* 84, 12–25. URL <https://linkinghub.elsevier.com/retrieve/pii/S1087184515300281>.
- Vidal-Diez de Ulzurrun, G., Huang, T.-Y., Chang, C.-W., Lin, H.-C., Hsueh, Y.-P., Oct. 2019. Fungal feature tracker (FFT): a tool for quantitatively characterizing the morphology and growth of filamentous fungi. *PLoS Comput. Biol.* 15 (10), e1007428. URL <https://dx.plos.org/10.1371/journal.pcbi.1007428>.
- Vogt, P., Riitters, K., Jan. 2017. GuidosToolbox: universal digital image object analysis. *Eur. J. Remote Sens.* 50 (1), 352–361. URL <https://www.tandfonline.com/doi/full/10.1080/22797254.2017.1330650>. <https://doi.org/10.1080/22797254.2017.1330650>.
- Wang, G., Lopez-Molina, C., Vidal-Diez de Ulzurrun, G., De Baets, B., Jul. 2019. Noise-robust line detection using normalized and adaptive second-order anisotropic Gaussian kernels. *Signal Process.* 160, 252–262. URL <https://linkinghub.elsevier.com/retrieve/pii/S0165168419300842>.
- Xu, H., Blonder, B., Jodra, M., Malhi, Y., Fricker, M., Jan. 2021. Automated and accurate segmentation of leaf venation networks via deep learning. *New Phytol.* 229 (1), 631–648. URL <https://onlinelibrary.wiley.com/doi/10.1111/nph.16923>.
- Zhang, H., Söderholm, N., Sandblad, L., Wiklund, K., Andersson, M., Jun. 2019. DSeg: a dynamic image segmentation program to extract backbone patterns for filamentous bacteria and hyphae structures. *Microsc. Microanal.* 25 (3), 711–719. URL [https://www.cambridge.org/core/product/identifier/S1431927619000308/type/journal\\_article](https://www.cambridge.org/core/product/identifier/S1431927619000308/type/journal_article).