TreeJ: an ImageJ plugin for interactive cell lineage reconstruction from static images

Elise Laruelle, Jean-Christophe Palauqui, Philippe Andrey, Alain Trubuil

SOFTWARE

TreeJ: an ImageJ plugin for interactive cell lineage reconstruction from static images

Elise Laruelle[1,2,3*], Jean-Christophe Palauqui[1], Philippe Andrey[1] and Alain Trubuil[2]

Abstract

Background With the emergence of deep-learning methods, tools are needed to capture and standardize image annotations made by experimentalists. In developmental biology, cell lineages are generally reconstructed from time-lapse data. However, some tissues need to be fixed to be accessible or to improve the staining. In this case, classical software do not offer the possibility of generating any lineage. Because of their rigid cell walls, plants present the advantage of keeping traces of the cell division history over successive generations in the cell patterns. To record this information despite having only a static image, dedicated tools are required.

Results We developed an interface to assist users in the building and editing of a lineage tree from a 3D labeled image. Each cell within the tree can be tagged. From the created tree, cells of a sub-tree or cells sharing the same tag can be extracted. The tree can be exported in a format compatible with dedicated software for advanced graph visualization and manipulation.

Conclusions The TreeJ plugin for ImageJ/Fiji allows the user to generate and manipulate a lineage tree structure. The tree is compatible with other software to analyze the tree organization at the graphical level and at the cell pattern level. The code source is available at https://github.com/L-EL/TreeJ.

Keywords Cell lineage, Image annotation, ImageJ software

Background

Understanding how cell organizations emerge is one of the major challenges in developmental biology. Cells at any particular stage result from several rounds of cell divisions. Reconstructing the history of successive cell division events as a cell lineage tree is thus essential for understanding cell patterning [1–3]. In situations amenable to time-lapse microscopy, various programs exist to track cells and their divisions (ALT [3], LineageTracker

[4], MorphographX [5], etc.). However, in some contexts, tissue fixation is required prior to image acquisition. In plants, because of the absence of cell migration, cell lineage can still be inferred from the resulting static images, by recursively backtracking cell division events. This requires that the patterning is stereotyped enough or that clues are available for the biologist to identify daughter cells. One of the clues is the cell geometry that can be conserved during tissue growth. An expert eye carries this knowledge that can be difficult to translate into an algorithm [6].

With the emergence of deep learning methods, neural networks will increasingly be used to predict lineages, as in other biological kingdoms [7], but more manually annotated data will be needed to train the networks. In addition, no user-friendly tool has been proposed to date in the bioimage informatics community to assist

*Correspondence:
Elise Laruelle
elise.laruelle@gmail.com
[1] Université Paris-Saclay, INRAE, AgroParisTech, Institut Jean-Pierre Bourgin (IJPB), Route de Saint Cyr, 78000 Versailles, France
[2] MaIAGE, INRAE, Université Paris-Saclay, Domaine de Vilvert, 78350 Jouy-en-josas, France
[3] Sainsbury Laboratory, Cambridge University, Bateman Street, CB2 1LR Cambridge, UK

Laruelle *et al. Plant Methods*   (2023) 19:128

Page 2 of 10

biologists in reconstructing cell lineages from static images [6].

Here, we present a graphical interface integrated in ImageJ/Fiji [8–10] to reconstruct a lineage tree from a static 3D image stack. The interface design is intuitive and the experimenter can easily link and annotate cells in a recursive way.

Implementation

The aim of this work was to develop a tool for building interactively a lineage tree from a labeled image of cells in a developing tissue or organism. To conceive this tool, we choose ImageJ/Fiji, a user-friendly software that is widely used by experimental scientists. ImageJ/Fiji contains a plugin development functionality. We developed the TreeJ plugin in Java within this framework.

GUI

The created plugin embeds the current image in a graphical interface (Fig. 1A). The TreeJ interface is composed of three parts. The 2D image, or the current slice of the 3D image stack, is shown in the middle to allow direct point-and-click interactions for selecting the cells to link. The right-panel displays buttons to construct, label, and extract trees and sub-trees. The left-panel frame displays the ongoing tree and provides input/output buttons. Depending on the user question, different outputs can be generated.

Input

TreeJ is compatible with a large spectrum of labeled images obtained by any region segmentation software. The interface can be launched on any 2D or 3D label image from 8 to 32 bits. The labeling can be discontinuous (i.e., missing labels), as frequently encountered as a result of post-processing operations applied to segmented images. The only assumption on the input image is that the segmentation labels are strictly positive; if present, the label 0 should be the background of the segmented structure (limitation explained in the 'Tree file' paragraph below).

Tree underlying structure

Internally, TreeJ constructs binary tree structures from the user input directives. Java classes have been used to organize the tree(s) with node objects storing cell information. A node is composed of a cell label (from 0 to the higher label number $label_{max}$), a user annotation (initialized with an empty string) and the labels of the two daughter cells (initialized to none for the non-mother cells present in the input segmentation, i.e., leaves of the tree).

Upon plugin opening, cells (i.e. labels) present in the image generate leaves (nodes without linked daughter cells). Then, new nodes can be created by the user assignment of cell links (Fig. 1B). A new cell label is assigned to each newly created mother cell. This label is automatically defined as $label_{max} + 1$ and attached to the created node. The new node stores also the labels of the linked cells (i.e., the daughter cells). This process progressively builds a tree (or several disjoint trees), which is displayed in the left panel of the interface (Fig. 1C).

Tree file

TreeJ allows to save and load a reconstructed tree in an home made format called treeV (file extension .treeV). A treeV file is an ASCII file, containing two arrays: one encoding the tree and the other encoding the annotations of each cell. For an embryo containing *n* cells, the initial structure (i.e., without lineage) of the treeV will be:

array index	0	1	2	...	n-2	n-1
cell label	1	2	3	...	n-1	n
mother cell label	0	0	0	...	0	0
cell tag	T_1	T_2	T_3	...	T_{n-1}	T_n

where the two gray lines are the treeV data and T_i is the tag of the *i*-th cell.

The tree(s) is stored as a list with as many elements as the highest label number, typically corresponding to the number of input cells (*n* in the table above) added to the number of reconstructed mother cells (Fig. 1D). By matching the indexes of the list and the labels of the cell, the tree(s) structure can be encoded in only one array. Each position in the array corresponds to a cell. The value stored in the array at any position gives the label of the corresponding mother cell.

For the root(s) of the constructed tree(s), the value 0 is assigned to indicate the absence of mother cell. This implies that TreeJ implicitly assumes that no cell is labeled 0 in the input segmentation. Otherwise, mismatch assignment would result.

The annotations are saved in the treeV file as an array of character strings indexed by cell labels. By default, the annotation of each cell is an empty string. TreeV files can be loaded by TreeJ to continue, edit or visualize a previous lineage reconstruction. To transfer the lineage in another software, Newick format is also available as saving format.

Results: TreeJ functionalities

Experimentalist's observations are sometimes untraceable by an algorithm and need to be saved and structured to be analyzed and shared. In plant epithelia, the lineage
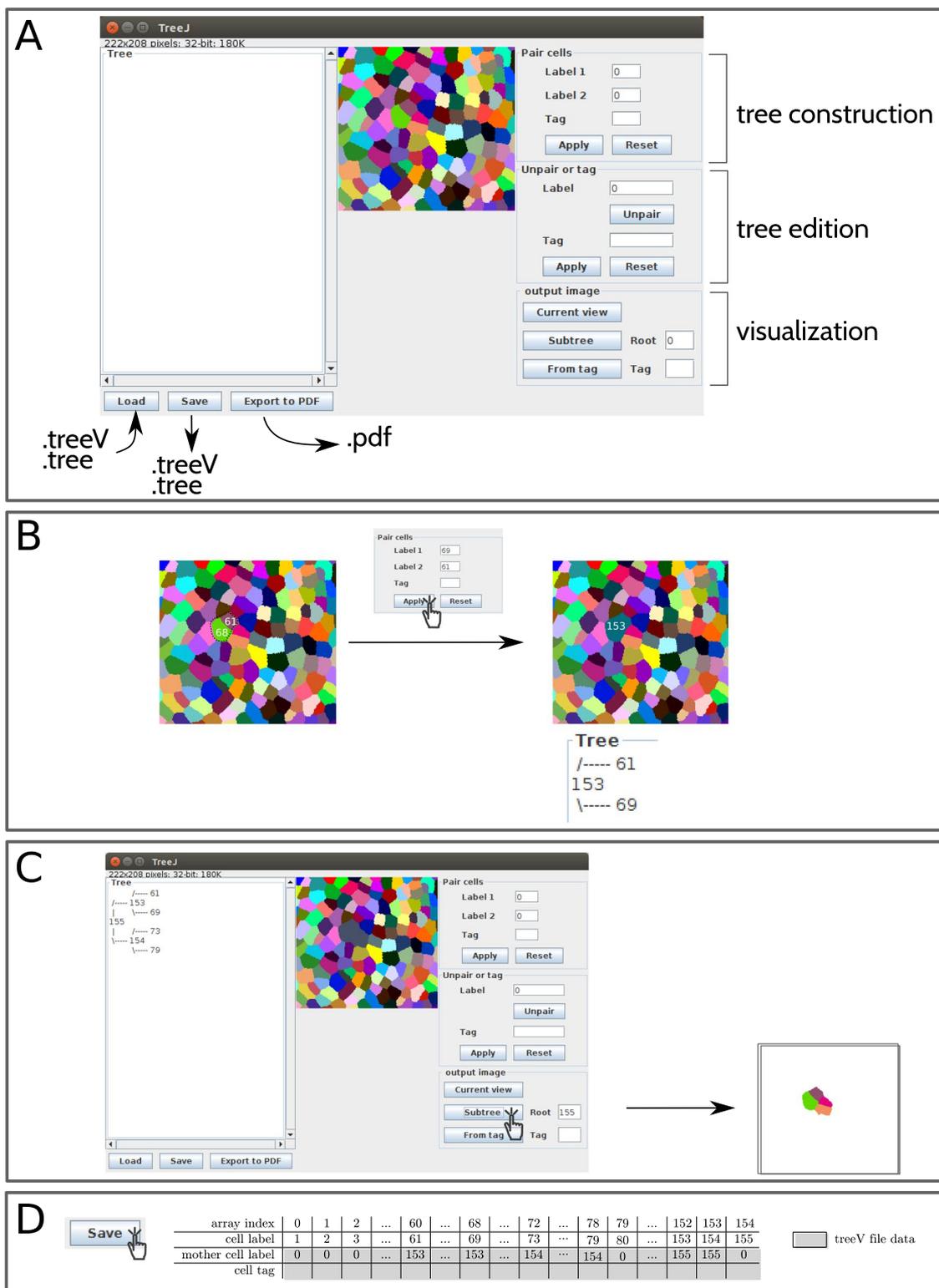
Fig. 1 TreeJ interface and main user interactions. **A** TreeJ interface showing the surface of an *A. thaliana* shoot apical meristem. **B** Binding of sister cells, the elementary step during the reconstruction of a cell lineage tree. **C** Pulling out a set of sister cells from their common ancestor node. **D** Saving the tree built in C using the treeV format. The treeV format contains 2 arrays (shown in gray) storing tree structure and annotations, respectively. In this example, the second array is empty, because the tree is not annotated

is one of these data that can be intuitively deduced from the cell pattern by the biologist. Tree data structures are available to represent and exploit such observations. TreeJ provides the interface between these structures and the experimentalist to generate, correct, annotate and visualize cell lineages.

Lineage reconstruction

The user reconstructs the lineage tree by recursively pairing sister cells (Fig. 1B). To link two cells, the user must either click the pair of cells on the image or enter manually their label values in the dedicated fields. Validating the pairing results in a new node being added to the tree, the merging of the pair of cells, and the automatic update of the display.

Errors can easily be corrected at any depth in the tree. The unlink button undoes erroneous pairings by splitting back any selected mother cell.

Annotating cells and sub-trees

TreeJ allows the introduction of biologically important information such as cell identity or localization in specific anatomical domains. Each node of the cell lineage tree can be tagged with an arbitrary text identifier. This annotation can be done when pairing cells, or later on. The same tag can be used for different cells (mother and daughters), thus supporting the annotation of whole domains (see the displayed tree in Fig. 2A).
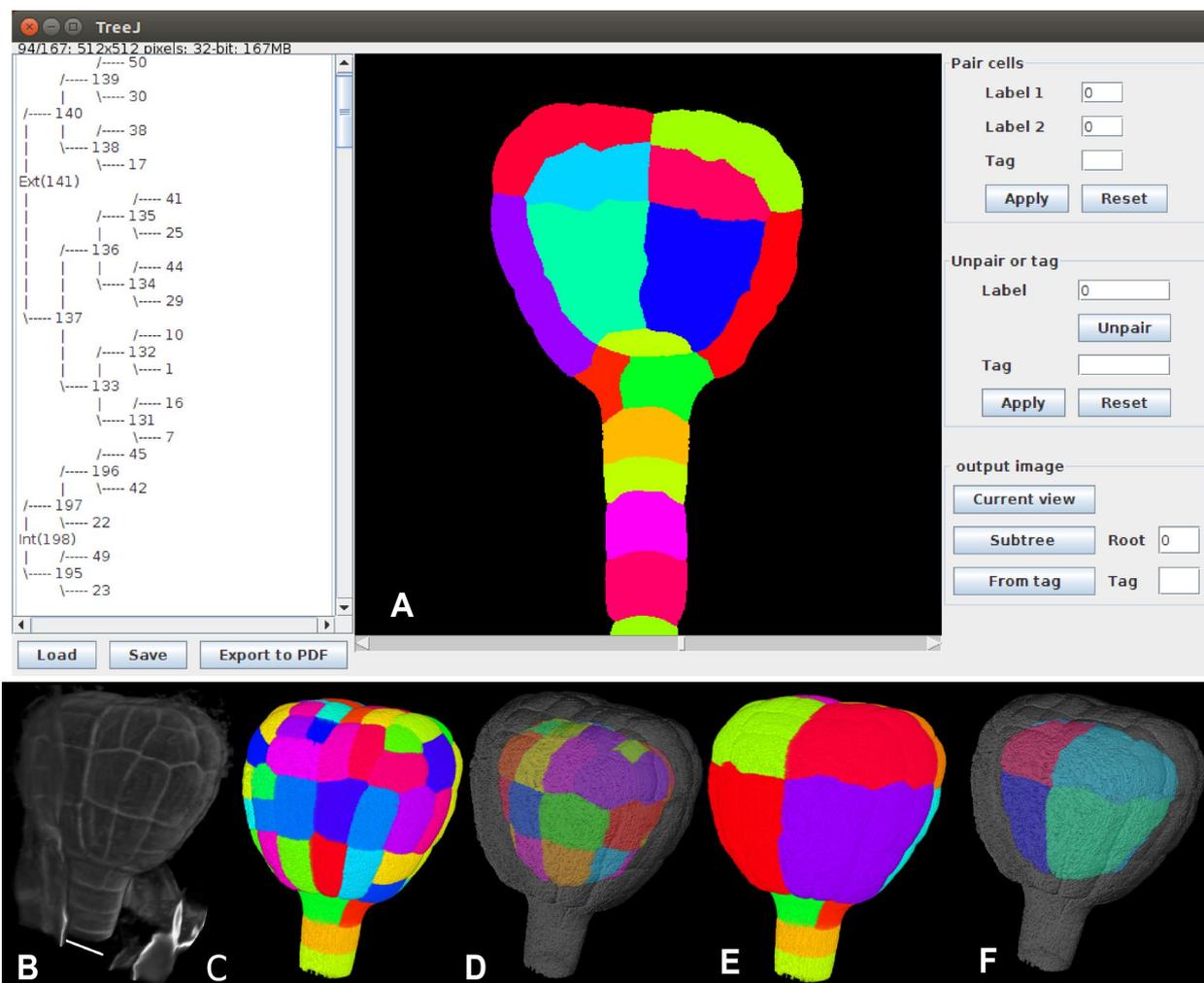
Fig. 2 Cell lineage reconstruction and annotation using TreeJ: illustration with a plant embryo. **A** TreeJ interface showing an *A. thaliana* 122-cells embryo tracked back over four cell generations. **B** 3D embryo raw image (scale bar: 10 μm). **C** 3D segmentation of image shown in **B**. **D** Output image of tagged internal tree nodes. **E** Output image after the back tracking, external parts. **F** Output image after the back tracking, inner parts. 3D renderings were made using the ImageJ 3D viewer [11]

Laruelle *et al. Plant Methods* (2023) 19:128

Page 5 of 10

Output

At any time during the reconstruction of a lineage tree, the user can create an independent copy of the modified label image, which can thus be visualized and analyzed with any tool available in ImageJ (see examples developed in the applications below). TreeJ also allows to create images containing a subset of cells from a sub-tree specified by its mother cell label or a subset of cells sharing the same tag (Fig. 1C and Fig. 2D). This provides further opportunities for validating the reconstructed lineages and to analyze quantitatively selected anatomical domains.

The reconstructed tree graphs can be saved under PDF, treeV, or Newick file formats. TreeV, a home made format, can be taken as input by TreeJ, thus allowing lineage reconstruction and corrections over several sessions. Newick is a generic tree format recognized by other tools for further tree processing.

Applications

Lineage reconstruction of an *Arabidopsis thaliana* embryo

The *Arabidopsis thaliana* early embryo development is a stereotypical process with regard to geometry and patterning [12–14]. Starting from a unique cell, the embryo establishes the main tissues and organs in less than ten cell generations. Tissue patterning is an important aspect of embryo development, which appears very early in the process. At the third generation, an apical-basal axis is established and, at the fourth cell generation, the protoderm is formed from a periclinal division with an asymmetrical volume-ratio. After the 4th cell generation, in the 16-cells embryo, divisions start to be variable.

Because of the thickness of the seed coat, embryos must be fixed to study their cellular organization (note that it is possible to follow embryo development using live imaging [13], but the resolution is not accurate enough to properly segment cells in 3D). The use of fixed developmental times complicates the study of the cell pattern dynamics, but the lineage can still be inferred because of the cell walls that almost fix the cell organization. As an example, in order to follow the evolution of the volume asymmetry between protodermal and internal domains downstream the fourth generation, we considered an embryo of 122 cells (7th generation). Cell walls were stained with Propidium Iodide [15] (Fig. 2B). The acquired 3D image was segmented with the *Morphological Segmentation* ImageJ plugin [16](Fig. 2C). The segmentation was manually corrected when errors were present.

We used TreeJ to reconstruct lineage trees (Fig. 2A). Recursively, sister cells were inferred depending on the compatibility of their cell shapes, but also depending on the expert knowledge of embryo development (Fig. 3). Once the embryo had been tracked back to the 16-cells stage, we annotated the mother cells corresponding to the protoderm (Ext) and the inner tissues (Int) (Fig. 2A).

In the literature, it has been described that the cell divisions are almost synchronized during the early embryo development [17]. With the improvements in imaging methods, the time resolution has shown a delay between the cell divisions. This observation is well seen in older embryo lineage trees [12, 13]. Similarly, using TreeJ on a 122-cells embryo, the reconstructed lineage presented a cell generational gap. Unbalanced trees were observed in the reconstructed lineage (Fig. 2A and Fig. 3). The
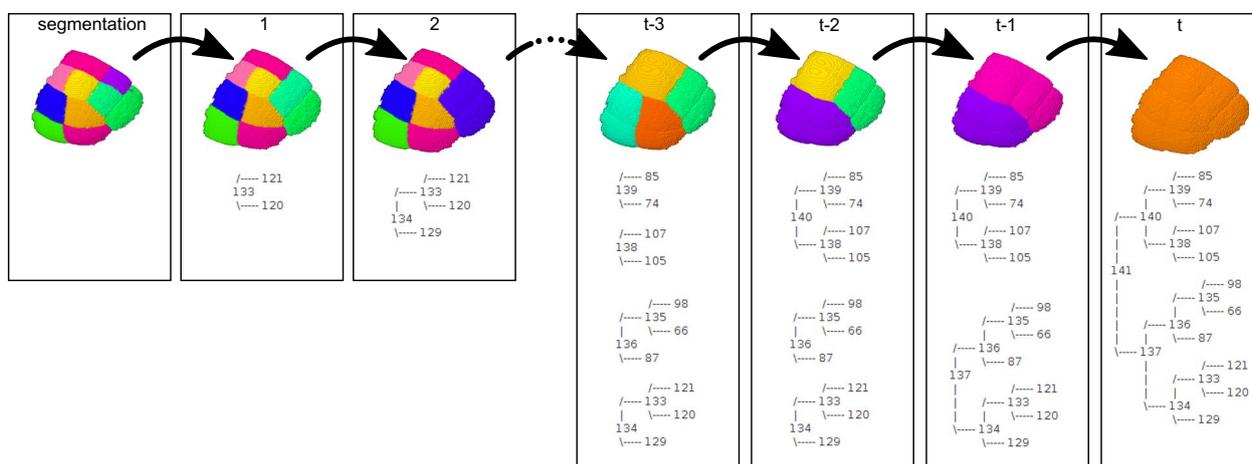
Fig. 3 Cell lineage reconstruction in a protodermal domain. Combination of 3D visualizations of TreeJ outputs generated with the ImageJ 3D viewer [11], and of trees shown in the TreeJ frame during the backtracking process. Outputs obtained by recursively tracking cell divisions back to the progenitor cell of the considered domain at the 16-cells embryo stage. Backtracking containing t steps

Laruelle *et al. Plant Methods*   (2023) 19:128

Page 6 of 10

protoderm layer had at least one additional cell generation compared with internal tissues.

Using the tag image extraction functionality of TreeJ, we separated in two distinct images the cells descending from the ancestors of either the epidermal or the inner tissues (Fig. 2D). With the tools contained in ImageJ, we then analyzed the difference of volume and cell number between the two tissues. We extracted cell volumes in each image with the Analyze Regions 3D functionality of the MorphoLibJ package. We observed that more cells were present in the protoderm (73 cells) than in the inner tissue (49 cells). This result was consistent with the reported generational shift [13]. In addition, the average volume for internal cells was $468.8 \pm 198.7\ \mu m^3$ (average $\pm$ s.d.) and $404.3 \pm 140.3\ \mu m^3$ for external cells. These volume distributions were also in agreement with previous reports [12].

Using the current view image extraction functionality of TreeJ, we also obtained the volumes of the rewound cells at the fourth generation. We split the rewound cells from the four different quarters of embryo at generation 4 (with ImageJ tools for labeled images from [18]). We then computed the average volume ratio between protodermal cells and internal cells in the apical and in the basal domains of each quarter of the rewound 122-cells embryo. The average volume ratios in the apical and in the basal domains were 0.70 ± 0.06 and 0.83 ± 0.04, respectively. Thus, the volume asymmetry between the protoderm and the internal tissue [12, 14] is still present after more than two cell generations.

Reconstructed dynamics of *Arabidopsis thaliana* embryo development

Using TreeJ over several embryos, fixed at different stages of their development, allows to reconstruct an approximation of the embryogenesis dynamics [19]. We annotated the internal and protodermal mother cells of 36 embryos between 33 and 257 cells. Using a python script, annotations from TreeJ were used in combination with the segmented images to compute the distributions of cell volumes (Fig. 4A) and the volume of every tissue per embryo (Fig. 4B). Ordering embryos by their numbers of cells allowed us to reconstruct a temporal evolution of these quantities.

A globally decreasing trend in the average cellular volume in the two domains was observed (Fig. 4A). This tendency is consistent with the relatively faster pace of cell proliferation compared to cell growth during early embryogenesis [14]. In spite of fluctuations in mean cell volume between successive times, due to variability between cells and between embryos, there was a marked difference in the average volume between cells in the protoderm and cells in the inner tissues. The higher volume
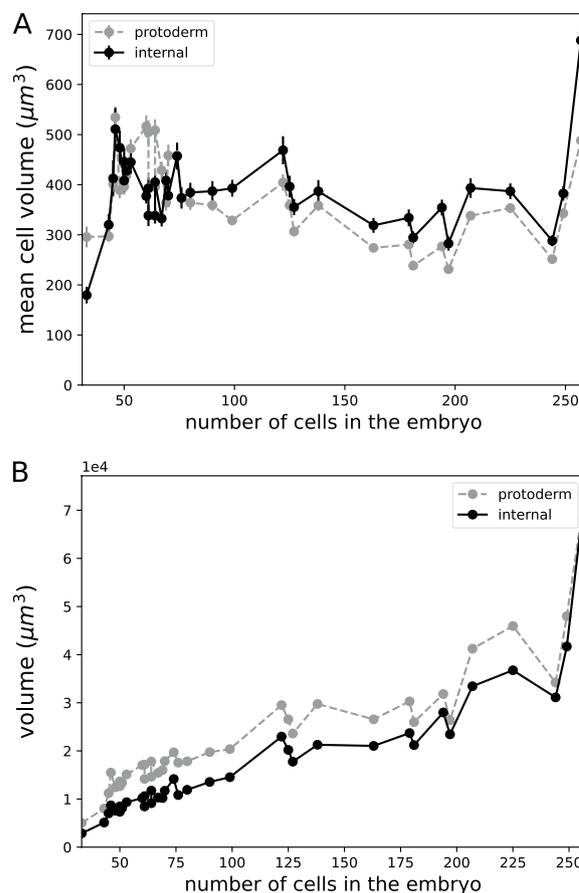
Fig. 4 Dynamics of cell and domain growth during *Arabidopsis thaliana* embryo development. Measurements over 36 embryos (from [19])(dots) ordered by their total number of cells. **A** Growth dynamics for internal and for external (protoderm) cells. Error bars: s.e.m. **B** Growth dynamics of the internal and the external (protoderm) domain

in the internal tissues can hypothetically be explained by a lower number of cells, because of the cell generation delay. The higher total volume of the protoderm compared to the volume of inner tissue (Fig. 4B) is consistent with this interpretation.

These results illustrate how TreeJ can help to analyze and reveal global trends in the dynamics of different domains from fixed observations, thanks to the manual annotations that can be performed in addition to the reconstruction of cell lineages.

Other application of TreeJ

TreeJ facilitates the reconstruction of trees by linking cells. The reason to link cells together can be due to a lineage relationship, but it can also be due to any other types of relation, as a spatial relation or a functional relation. In the example below we illustrate how TreeJ can also

Laruelle *et al. Plant Methods*   (2023) 19:128

Page 7 of 10

be used for the purpose of cell annotation and grouping, independently of any lineage relationship.

Plant roots show a stereotyped radial organization all along their longitudinal axis [20, 21] (Fig. 5A,B). This organization is emerging from the root tip, where a group of stem cells in the root apical meristem continuously divides to participate in the root growth. The stem cells are organized radially to generate the tissue radial organization. Each stem cell in the root meristem is at the origin of a cell file in a tissue layer and ensures its longitudinal growth. The cells close to the root meristem are the youngest and the cell age increases with cell position along the longitudinal axis. Hence, in a fixed root, the longitudinal axis unveils the time scale. The need for tracking divisions is less relevant in the

root, but other cell relations can be pertinent and easily determined by the experimentalist.

The root is regionalized depending on several factors. The longitudinal axis can be divided in several parts [22, 23]: there is firstly a dividing part at the root meristem, then an elongation zone where cells grow and then a maturation area where cells end their development to be fully functional. Along the radial axis, the concentric tissues have different functions (Fig. 5A). From outside to inside, there are, firstly, the external cells that constitute the epidermis, forming a load-bearing barrier resisting the growth exerted by inner tissues, then cells forming the cortex, a storage tissue. The third layer of cells is the endodermis, a filtering tissue protecting the
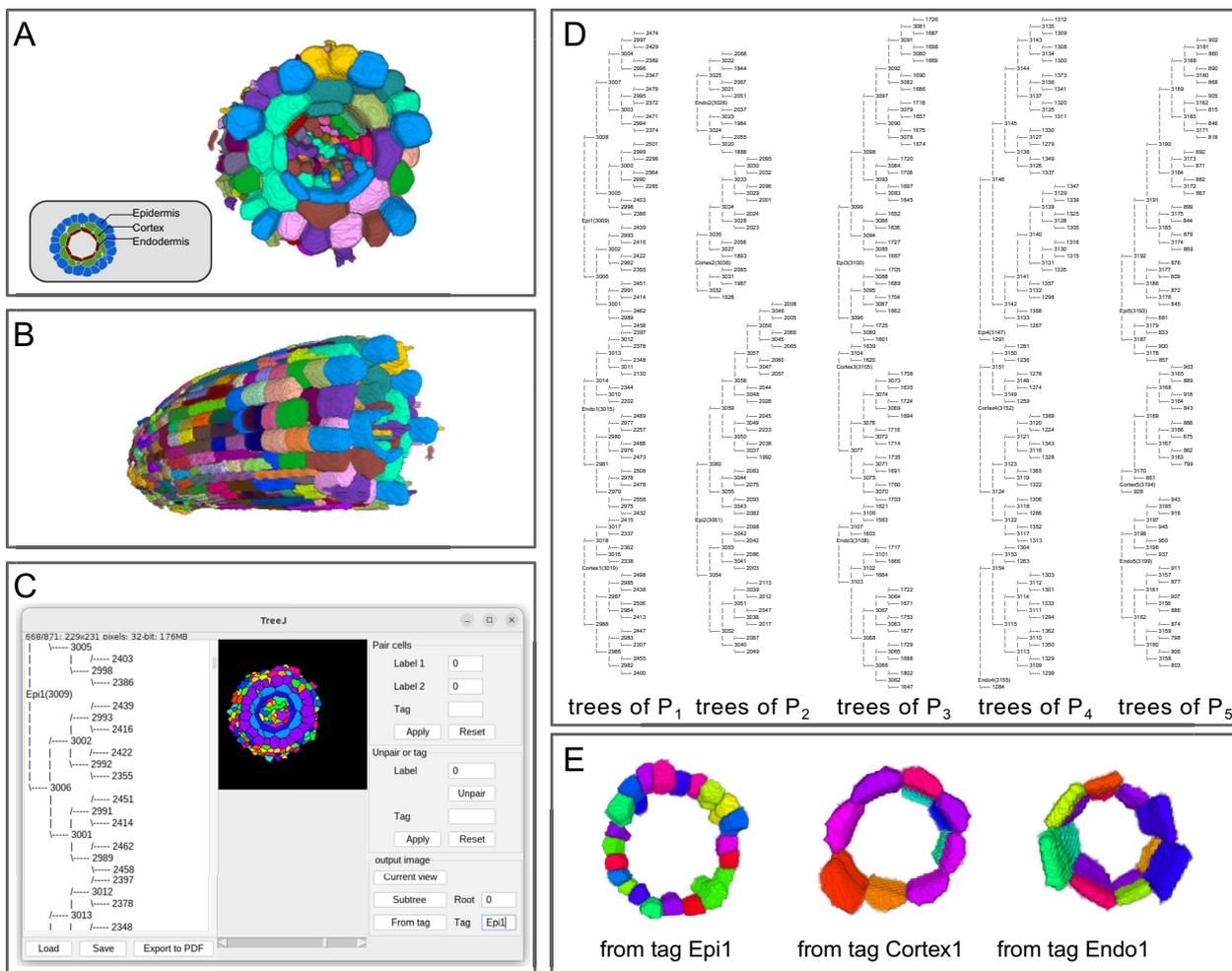
Fig. 5 Grouping and annotations of cells in an *Arabidopsis thaliana* root. **A**, **B** 3D visualizations with the ImageJ 3D viewer [11] of the segmentation obtained with the *Morphological Segmentation* ImageJ plugin [16], top and side views, respectively. **C** TreeJ interface after tree reconstruction and tagging. **D** Reconstructed trees for each position (P_x with x from 1 to 5) and each layer. Tags assigned to the grouped cells contain the layer name and the position number. **E** 3D visualizations with the ImageJ 3D viewer of the images obtained using tag-based selection of the three layers at Position 1 (top view)

Laruelle *et al. Plant Methods*   (2023) 19:128

Page 8 of 10

inner tissues composed of the pericycle (giving rise to lateral roots) and of the vascular tissues.

Given the root organization along two axes, one can ask how the different layers (radial axis) behave during development (longitudinal axis). Most of the developmental studies about the root have been performed in 2D, although recently, 3D measurements have gained interest [24–26] to phenotype more precisely cell events depending on various parameters (position, function, etc). With TreeJ, we can build metadata to analyze groups of cells by their type and by their age.

We selected five equidistant positions from the meristematic zone along a 3D segmented root image (the protocol for image acquisition and segmentation was the same as for the embryo). At each position, the cells from each tissue layer (epidermis, cortex or endodermis) were grouped into a separate tree, annotated by the corresponding position and layer (Fig. 5C). Here, we took advantage of the possibility in TreeJ to store into a single representation an arbitrary number of disjoint trees (Fig. 5D). Note that, in this application, cells were linked and incorporated in an arbitrary order in the trees without considering their actual lineage relationships, since the trees were used as a way to group cells by position and by tissue. Using the tag-based image creation functionality of TreeJ, a separated image was generated for each tree to extract measurements (Fig. 5E). The mean cell volume per group (position+layer) was calculated from cell volume measurements obtained with the Analyze Regions 3D plugin of the MorphoLibJ package [16] (Fig. 6A). The statistical analysis showed a similar average cell volume increase for the epidermis and for the cortex layer (Kolmogorov-Smirnov (KS) test, $P > 0.05$). The endodermis contained smaller cells (KS-test; $P < 0.05$ from Position 3) due to asymmetric formative division giving rise to endodermis and cortical cells [27]. These observations are consistent with a recently reported 3D analysis [24].

The epidermis is composed of two types of cells: the trichoblasts (giving root hairs) and the atrichoblasts [28, 29]. Previous 2D descriptions showed a differential shape between these two cell types [30]. As trichoblasts are wider and atrichoblasts are more elongated, we used TreeJ to evaluate whether their 3D volumes differed at the root tip. We discriminated the two cell types by their cell spatial position relatively to cortex cells and manually grouped cells from each type with TreeJ. As above, the cells linked and tagged at each position could easily be separated in two distinct images and the two cell types could be analyzed separately, a process we automated over the five longitudinal positions with a python script. In spite of a slight systematic offset between the volumes of trichoblasts and atrichoblasts, with an apparent
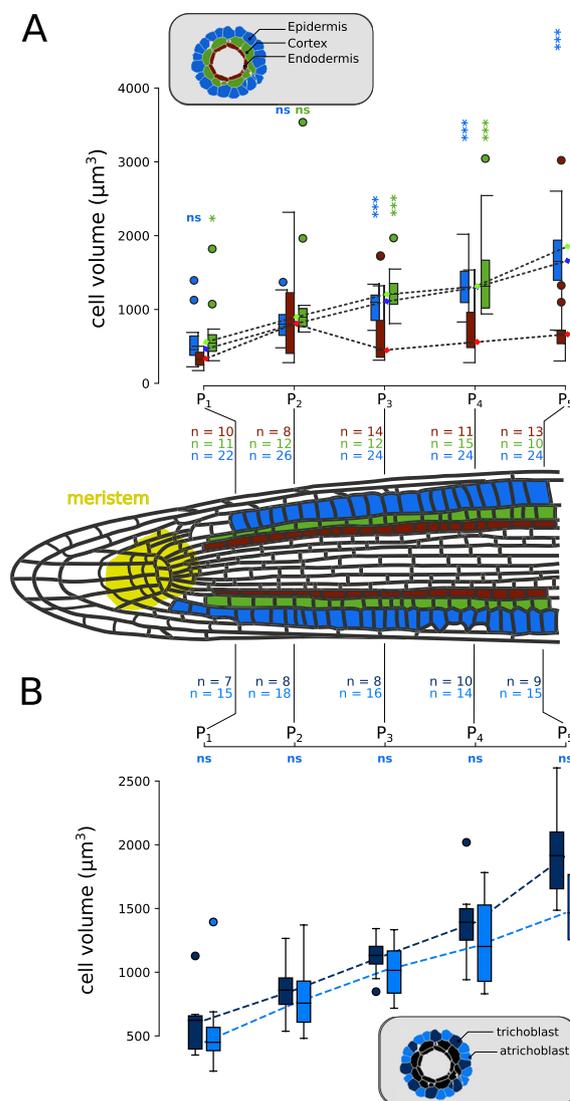
Fig. 6 Longitudinal evolution along the root axis of cell volume, per tissue (**A**) and per cell type (**B**). **A** Cell volume in the three first root layers at five equidistant positions. Kolmogorov-Smirnov tests applied between the endodermis and either the epidermis (blue) or the cortex (green): * ($P < 0.05$) and *** ($P < 0.001$). **B** Cell volume of trichoblasts and atrichoblasts at the five same positions. Kolmogorov-Smirnov tests applied between trichoblast and atrichoblast distributions showed no significant differences (ns)

increase along the longitudinal axis, we observed no significant difference (Fig. 6B). This result is consistent with recent 3D observations [24] but further analyses, including larger samples, may be required to examine a possible growth divergence at the end of the analyzed region. TreeJ will be useful for this purpose.

With these applications, we have shown that TreeJ offers the possibility to store any cell relationships contained in a static image in a unique file. This file can be

Laruelle *et al. Plant Methods*   (2023) 19:128

Page 9 of 10

used outside of ImageJ/Fiji and it provides an easy access to the experimentalist's annotations.

Conclusion

With TreeJ, cell lineages can easily be constructed, visualized, and exported from a static segmented image. The formatted tree representation shows the relative division rates and can be used as a phenotype of the system to compare between individuals or mutants. Using TreeJ adds robustness to the lineage reconstruction process from static images and provides standardized representations that facilitate communication and sharing among the community. In addition to cell lineage reconstruction, TreeJ supports further analyses related to cell genealogy. Thanks to the TreeJ image export functionality, it is possible to extract the descendants from a common precursor or to rewind cell organizations back to previous configurations. Many analyses downstream tree reconstruction can be performed within the ImageJ/Fiji ecosystem. However, to perform new measurements or analyses, the reconstructed trees can be extracted and analyzed with, for example, dedicated python scripts. TreeJ can be used on any segmented tissue, be it animal or plant.

Future developments encompass the automation of pairing [31] or annotation [32] based on previous entries with machine-learning schemes. AI models could be trained with reference data sets created from TreeJ outputs. Adding a pretrained model in an ImageJ/Fiji plugin is now within reach [33–35].

Availability of data and materials
Most of the plant materials and the raw data of the figures used in this study are available on the Github web page at https://github.com/L-EL/TreeJ/. The plugin is also available at http://imagej.net/plugins/treej. Version v1.0.0 can be found here [36]. Project name: TreeJ. Project home page: https://github.com/L-EL/TreeJ. Archived version: 10.5281/zenodo.10041643. Operating system(s): Platform independent. Programming language: Java. Other requirements: java 1.8 (ImageJ2/Fiji 1.51n or higher) and itextpdf plugin. License: GNU GPL.

Declarations

Competing interests
The authors declare that they have no competing interests.

References

1. Bao Z, Murray JI, Boyle T, Ooi SL, Sandel MJ, Waterston RH. Automated cell lineage tracing in *Caenorhabditis elegans*. Proc Natl Acad Sci. 2006;103(8):2707–12. https://doi.org/10.1073/pnas.0511111103.
2. Al-Kofahi O, Radke RJ, Goderie SK, Shen Q, Temple S, Roysam B. Automated cell lineage construction: a rapid method to analyze clonal development established with murine neural progenitor cells. Cell Cycle. 2006;5(3):327–35. https://doi.org/10.4161/cc.5.3.2426.
3. Fernandez R, Das P, Mirabet V, Moscardi E, Traas J, Verdeil J-L, Malandain G, Godin C. Imaging plant growth in 4D: robust tissue reconstruction and lineaging at cell resolution. Nat Methods. 2010;7(7):547–53. https://doi.org/10.1038/nmeth.1472.
4. Downey MJ, Jeziorska DM, Ott S, Tamai TK, Koentges G, Vance KW, Bretschneider T. Extracting fluorescent reporter time courses of cell lineages from high-throughput microscopy at low temporal resolution. PLoS ONE. 2011;6(12):1–12. https://doi.org/10.1371/journal.pone.0027886.
5. Barbier de Reuille P, Routier-Kierzkowska A-L, Kierzkowski D, Bassel GW, Schüpbach T, Tauriello G, Bajpai N, Strauss S, Weber A, Kiss A, et al. MorphoGraphX: a platform for quantifying morphogenesis in 4D. Elife. 2015;4:05864. https://doi.org/10.7554/eLife.05864.
6. Hong J, Sanz FA, Trubuil A, Isenberg T. Design and evaluation of three selection techniques for tightly packed 3D objects in cell lineage specification in botany. In: GI 2021-Graphics Interface; 2021. p. 213–223. https://doi.org/10.20380/GI2021.33
7. Lugagne J-B, Lin H, Dunlop MJ. DeLTA: automated cell segmentation, tracking, and lineage reconstruction using deep learning. PLoS Comput Biol. 2020;16(4):1–18. https://doi.org/10.1371/journal.pcbi.1007673.
8. Schindelin J, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, Preibisch S, Rueden C, Saalfeld S, Schmid B, et al. Fiji: an open-source platform for biological-image analysis. Nat Methods. 2012;9(7):676–82. https://doi.org/10.1038/nmeth.2019.
9. Rueden CT, Schindelin J, Hiner MC, DeZonia BE, Walter AE, Arena ET, Eliceiri KW. Image J2: ImageJ for the next generation of scientific image data. BMC Bioinformatics. 2017;18:1–26. https://doi.org/10.1186/s12859-017-1934-z.
10. Schneider CA, Rasband WS, Eliceiri KW. NIH image to ImageJ: 25 years of image analysis. Nat Methods. 2012;9(7):671–5. https://doi.org/10.1038/nmeth.2089.
11. Schmid B, Schindelin J, Cardona A, Longair M, Heisenberg M. A high-level 3D visualization API for Java and ImageJ. BMC Bioinformatics. 2010;11(1):1–7. https://doi.org/10.1186/1471-2105-11-274.
12. Yoshida S, de Reuille PB, Lane B, Bassel GW, Prusinkiewicz P, Smith RS, Weijers D. Genetic control of plant development by overriding a geometric division rule. Dev Cell. 2014;29(1):75–87. https://doi.org/10.1016/j.devcel.2014.02.002.
13. Gooh K, Ueda M, Aruga K, Park J, Arata H, Higashiyama T, Kurihara D. Live-cell imaging and optical manipulation of arabidopsis early embryogenesis. Dev Cell. 2015;34(2):242–51. https://doi.org/10.1016/j.devcel.2015.06.008.
14. Moukhtar J, Trubuil A, Belcram K, Legland D, Khadir Z, Urbain A, Palauqui J-C, Andrey P. Cell geometry determines symmetric and asymmetric division plane selection in arabidopsis early embryos. PLoS Comput Biol. 2019;15(2):1–27. https://doi.org/10.1371/journal.pcbi.1006771.
15. Truernit E, Bauby H, Dubreucq B, Grandjean O, Runions J, Barthélémy J, Palauqui J-C. High-resolution whole-mount imaging of three-dimensional tissue organization and gene expression enables the study of phloem development and structure in arabidopsis. Plant Cell. 2008;20(6):1494–503. https://doi.org/10.1105/tpc.107.056069.
16. Legland D, Arganda-Carreras I, Andrey P. MorpholibJ: integrated library and plugins for mathematical morphology with ImageJ. Bioinformatics. 2016;32(22):3532–4. https://doi.org/10.1093/bioinformatics/btw413.
17. Griffith ME, Mayer U, Capron A, Ngo QA, Surendrarao A, McClinton R, Jürgens G, Sundaresan V. The TORMOZ gene encodes a nucleolar protein required for regulated division planes and embryo development in Arabidopsis. Plant Cell. 2007;19(7):2246–63. https://doi.org/10.1105/tpc.106.042697.

Laruelle *et al. Plant Methods*   (2023) 19:128

Page 10 of 10

18. Schaefer E, Belcram K, Uyttewaal M, Duroc Y, Goussot M, Legland D, Laruelle E, de Tauzia-Moreau M-L, Pastuglia M, Bouchez D. The prepro-phase band of microtubules controls the robustness of division orienta-tion in plants. Science. 2017;356(6334):186–9. https://doi.org/10.1126/science.aal3016.

19. Laruelle E, Belcram K, Trubuil A, Palauqui J-C, Andrey P. Large-scale analy-sis and computer modeling reveal hidden regularities behind variability of cell division patterns in Arabidopsis thaliana embryogenesis. Elife. 2022;11:79224. https://doi.org/10.7554/eLife.79224.

20. van den Berg C, Weisbeek P, Scheres B. Cell fate and cell differentiation status in the arabidopsis root. Planta. 1998;205(4):483–91. https://doi.org/10.1007/s004250050347.

21. Benfey PN, Scheres B. Root development. Curr Biol. 2000;10(22):813–5. https://doi.org/10.1016/S0960-9822(00)00814-9.

22. Vaughn L, Baldwin K, Jia G, Verdonk J, Strohm A, Masson P. The cytoskel-eton and root growth behavior. The plant cytoskeleton. Adv Plant Biol. 2011;307–26. https://doi.org/10.1007/978-1-4419-0987-9_14.

23. Bargmann BO, Vanneste S, Krouk G, Nawy T, Efroni I, Shani E, Choe G, Friml J, Bergmann DC, Estelle M, et al. A map of cell type-specific auxin responses. Mol Syst Biol. 2013;9(1):688. https://doi.org/10.1038/msb.2013.40.

24. Blein T, Duerr J, Pasternak T, Haser T, Falk T, Liu K, Ditengou FA, Ron-neberger O, Palme K. Light dynamically regulates growth rate and cellular organisation of the arabidopsis root meristem. bioRxiv; 2018. https://doi.org/10.1101/353987

25. Fridman Y, Strauss S, Horev G, Ackerman-Lavert M, Reiner-Benaim A, Lane B, Smith R, Savaldi-Goldstein S. The root meristem is shaped by brassinos-teroid control of cell geometry. Nat Plants. 2021;7(11):1475–84. https://doi.org/10.1038/s41477-021-01014-9.

26. Pasternak T, Haser T, Falk T, Ronneberger O, Palme K, Otten L. A 3D digital atlas of the Nicotiana tabacum root tip and its use to investigate changes in the root apical meristem induced by the Agrobacterium 6b oncogene. Plant J. 2017;92(1):31–42. https://doi.org/10.1111/tpj.13631.

27. Smolarkiewicz M, Dhonukshe P. Formative cell divisions: principal deter-minants of plant morphogenesis. Plant Cell Physiol. 2013;54(3):333–42. https://doi.org/10.1093/pcp/pcs175.

28. Galway ME, Masucci JD, Lloyd AM, Walbot V, Davis RW, Schiefelbein JW. The TTG gene is required to specify epidermal cell fate and cell pattern-ing in the arabidopsis root. Dev Biol. 1994;166(2):740–54. https://doi.org/10.1006/dbio.1994.1352.

29. Dolan L, Janmaat K, Willemsen V, Linstead P, Poethig S, Roberts K, Scheres B. Cellular organisation of the arabidopsis thaliana root. Development. 1993;119(1):71–84. https://doi.org/10.1242/dev.119.1.71.

30. Löfke C, Dünser K, Kleine-Vehn J. Epidermal patterning genes impose non-cell autonomous cell size determination and have additional roles in root meristem size control. J Integr Plant Biol. 2013;55(9):864–75. https://doi.org/10.1111/jipb.12097.

31. Hong J, Maciejewski R, Trubuil A, Isenberg T. Visualizing and comparing machine learning predictions to improve Human-AI teaming on the example of cell lineage. IEEE Trans Visual Comput Graph. 2023;1–14. https://doi.org/10.1109/TVCG.2023.3302308.

32. Zhang S, Jafari O, Nagarkar P. A survey on machine learning techniques for auto labeling of video, audio, and text data; 2021. arXiv preprint http://arxiv.org/abs/2109.03784. https://doi.org/10.48550/arXiv.2109.03784

33. Falk T, Mai D, Bensch R, Çiçek Ö, Abdulkadir A, Marrakchi Y, Böhm A, Deu-bner J, Jäckel Z, Seiwald K, et al. U-net: deep learning for cell counting, detection, and morphometry. Nat Methods. 2019;16(1):67–70. https://doi.org/10.1038/s41592-018-0261-2.

34. Encarnacion-Rivera L, Foltz S, Hartzell HC, Choo H. Myosoft: an automated muscle histology analysis tool using machine learning algorithm utilizing FIJI/ImageJ software. PLoS ONE. 2020;15(3):1–22. https://doi.org/10.1371/journal.pone.0229041.

35. Gómez-de-Mariscal E, García-López-de-Haro C, Ouyang W, Donati L, Lun-dberg E, Unser M, Munoz-Barrutia A, Sage D. DeepImageJ: a user-friendly environment to run deep learning models in ImageJ. Nat Methods. 2021;18(10):1192–5. https://doi.org/10.1038/s41592-021-01262-9.

36. L-EL: L-EL/TreeJ: TreeJ. Zenodo; 2023. https://doi.org/10.5281/zenodo.10041643

Publisher's Note