

Image analysis in plant sciences: Publish then Perish

Guillaume Lobet^{1,2}

¹ Agrosphäre (IBG-3), Forschungszentrum Jülich GmbH, Jülich, Germany

² Earth and Life Institute, Université catholique de Louvain, Louvain-la-Neuve, Belgium

Email: g.lobet@fz-juelich.de

Website: www.plantmodelling.xyz

Twitter: [@guillaumelobet](https://twitter.com/guillaumelobet)

Keywords

Image analysis, open science, open data, validation, maintenance

Trends

- Image analysis techniques and tools are used by a large proportion of the plant science community.
- Existing tools address a large proportion of current needs.
- Validation and benchmarking of these tools is variable and often overlooked.
- Long term maintenance of these tools is not always ensured.

Abstract

Image analysis has become a powerful technique for most plant scientists. In recent years, dozens of image analysis tools have been published in plant science journals. These tools cover the full spectrum of plant scales, from single cells, to organs and canopies. Yet, the field of plant image analysis is still in its infancy. It still has to overcome important challenges, such as the lack of robust validation practices or the absence of long-term support. In this essay, I will (i) present the current state of the field, based on data coming from the plant-image-analysis.org database, (ii) identify the challenges faced by its community, and (iii) propose workable ways of improvement.

Images as data in plant sciences

With the advent of affordable image acquisition devices (e.g. the Phenotiki project, <http://phenotiki.com>) and the development of more specialised ones [1], images have

become a commonly used data type in plant sciences [2]. From root systems [3–6] to single leaf venation and shape [7–9], complete shoots [10–12] or electrophoresis gels, many plant scientists are likely to generate at least some images as part of their research pipeline. As a direct result, more than 140 different plant image analysis tools are now available for the community [13], and this figure is still growing.

An overview of the development of a plant image analysis tool.

Development of plant image analysis tools often follows a bottom-up dynamic. The implementation of novel phenotyping methods [14], or the acquisition of new imaging devices [15] are drivers in the development of such tools. A large proportion of these tools are developed, at least in part, within plant research groups (Key figure 1A), most of the time by a non-permanent (early career) researcher , such as PhD students or postdocs (Key figure 1B).

During the development process of a novel image analysis tool, two key goals have to be achieved: (i) does it work (**validation**, see Glossary), and (ii) does it work better than any existing tools (**benchmarking**). If both answers are positive, then the tool is usually made freely available to the scientific community, sometimes under an open-source licence (Key figure 1F). This release is often coupled with a scientific publication presenting the new tool. Nevertheless, some tools (the amount of which is unclear and hard to quantify) are never released to the community or described in a formal publication.

Unlike classical research projects in plant sciences, the presentation of an image analysis tool in a scientific paper should not be the end point of the project. As for any software solution, the publication, its official introduction to the scientific community, is only the beginning of the life of the tool. Following that introduction, the community will ideally use the tool . This implies that some level of support and maintenance is expected from the developers, hopefully for many years.

Here I will discuss two challenges faced by the plant image analysis community: the validation and the long-term maintenance of software tools. I will also propose workable ways of improvement at a minimal cost. What I will not discuss are the technical programming issues, as this has been well covered recently [16,17].

Validation and benchmarking: a never ending story

In essence, validation of an image analysis tool is a simple one-step procedure: it consists of comparing its output with real measurements (**ground-truth**, see Box 1 for discussion). Ideally, this comparison should be done using a large dataset, to ensure that the tool is working correctly over a wide range of images. In theory, the validation step should also define the intrinsic limitations of the tool, and identify which type of images cannot be analysed. This is as important as the validation itself, especially for fully automated analysis pipelines where the user input can be restricted to loading the images into an “analysis black box”, with no quality control of the analysis. In such cases, a tool that was developed for seedlings could be used to compute a mature root system, which can lead to non-predictable, potentially significant errors [18].

83 In practice, this simple validation step is hampered by the difficulty of finding
84 good quality ground truth datasets. Such dataset can be created by manually (or semi-
85 automatically) measuring experimental images, but this can be highly time consuming
86 for complex objects (such as entire shoot or mature root systems). Another strategy is
87 to use artificial images (as similar as possible to real ones) created using modelling
88 tools (Key figure 1C). In the past, only a limited number (< 50) of images were used (fig
89 . 1D), often for a limited number of species and developmental stages, which is likely to
90 lead to incomplete validations. In addition, these images were never shared with the
91 community, except for a few examples (Key figure 1F, [15]). So, instead of expanding
92 existing ground-truth datasets with new images, each new tool validation is a one-shot
93 exercise that neither benefits from previous development nor is used in future ones. For
94 a handful of published tools, comparison with previously published tools was used as
95 validation (Key figure 1C). Such validation is, by nature, incorrect as it does not evaluate
96 the ability of the tools to predict the ground-truth. It might even lead to a propagation of
97 errors from one tool to the next. More concerning is that 29% of the papers references
98 on the plant-image-analysis.org database did not contain any type of validation (Key
99 figure 1C).

100 Whenever a tool is introduced as an improvement over an existing one, a
101 benchmarking analysis should be presented with the article. This is important, because
102 any new tools should perform no worse than previous ones. In this case, better
103 performance could mean many things: a higher accuracy, faster algorithms, an
104 increased number of extracted variables or an improved user interface. Today, the main
105 limitation when benchmarking accuracy (or speed) of a new tool, is that, as I stated

earlier, there are almost no standard datasets available (not only for the images, but also the corresponding ground-truth). In today's situation, performing a benchmarking analysis means using all the previous tools on a new dataset to compare them with the new tool. Again, the absence of shared image datasets hampers this important step and even discourages developers to perform any benchmarking (in-depth benchmarking are rarely shown).

Here, editors and reviewers, as gate-keepers of the publishing process, have the leverage to improve the situation, by systematically requiring proper validation and benchmarking for any new image analysis tool.

Large, shared datasets for better tools

Here I would like to propose a handful of practices that have the potential to improve the validation of plant image analysis tools and benchmarking. As stated above, the main limitation is the lack of freely available ground truth image dataset. Fortunately, several solutions exist to improve the current situation (Box 2).

Validation datasets need to be bigger and cover larger range of scales. I believe plant models have the potential to create such datasets as they can be used to generate an unlimited number of images, with their corresponding ground-truths. These models already exist for complex plant objects such as roots [19–21], shoots [22,23] or organs [24,25] but so far have been rarely used as validation dataset [18]. For images produced with modelling tools thresholds can be set, which can be used to estimate the error linked to the intrinsic structure of the analysed object. For situations where the

environmental noise from the experimental setup needs to be reproduced with accuracy, mock-up plants or computer-generated structures can be 3D printed and imaged in realistic conditions. Although it might be complicated to reproduce the exact same conditions (e.g. light reflectance of leaf is not the same as of synthetic material), it can provide a good approximation.

Sharing images is another way to increase the size of validation datasets. In an ideal world, any phenotyping dataset, not only the ones used for a specific validation, should be accessible to the community. It would allow future tools to be tested on a wide variety of plants, plant growth stages and imaging conditions. Data sharing is even more critical with the advent of machine learning in plant image analysis [26,27]. The accuracy of these algorithms indeed depends on the quality and amount of the data used for training. They would therefore highly profit from shared, ground-truth dataset. In addition to the images that were successfully analysed using the new tool, I believe it is also useful to share the image that cannot (and should not) be analysed. This is important for future users to fully understand the limitations of the tool in order to use it correctly.

Sharing images alone is not enough for an efficient recycling. For each image additional data need to be attached. Firstly, the experimental setup needs to be described [28]. Secondly, the image acquisition protocol needs to be carefully described, as subtle changes (e.g. image compression) have been shown to lead to significant differences in the image analysis [29]. Finally, and probably the most important point regarding the development of plant image analysis tools, the extracted data needs to be associated with the images. This data has to be attached in a parsable

format (e.g. the Root System Markup Language [30]) so it can be reused for further analysis. Non-parsable formats such as Portable Document Format (PDF) should be avoided at all cost since they can hardly be reused in analysis pipelines. Such datasets can be hosted on either generic platforms (see table 1 for examples) or specialised repositories such as DIRT [6], ClearedLeavesDB [31] or the Leaf Phenotyping Challenge [32]. To centralise and reference such datasets, a new section was recently created on the plant-image-analysis.org website (<http://www.plant-image-analysis.org/dataset>). At the time of publication, this new section referenced 25 datasets, totalling more than 1,700,000 images.

Ultimately, image and data sharing is a responsibility that rests on the shoulders of the whole plant science community, rather than just the scientists producing the data. Publishers, editors and reviewers are the drivers that have the power to enforce data sharing at the time of publication. As an example, researchers who signed the Peer Reviewers' Openness Initiative pledge not to review manuscripts unless the underlying data is made public (when possible) [33]. Several publishers (such as BioMed Central, eLife or the Public Library of Science) already include a data sharing obligation in their publication guidelines, but they remain marginal in the publishing landscape. Such enforcement has been done in the past for genomic data and lead to the creation of a huge resource for the scientific community (e.g. EBI-ArrayExpress host data from 1.5 million essays [34]). I believe the same should be done for image data.

172

173 Lack of incentive threatens long term maintenance.

174 Long-term maintenance of a software tool is not straightforward within academia. For
175 one thing, maintenance in itself is hardly rewarded by traditional means, as it does not
176 lead to a formal publication. In theory, tools that are well maintained are more likely to
177 be used by the community, which might generate citations for the associated paper.
178 Updates, when substantial, might lead to the writing of a new paper, but this is rarely
179 observed. As such, the maintenance is usually solely based on the goodwill of the
180 researcher who developed the tool. Long-term maintenance is further jeopardised by
181 the fact that most developers are non-permanent researchers (PhD students or
182 postdocs, Key figure 1A). After a few years, it is likely that the developer is no longer in
183 the group that initially needed the tool. The maintenance then becomes harder to
184 sustain with the increasing lack of incentive.

185 In situations where the tool is robust and well documented, it might survive a few
186 years on its own. However, as operating systems get major updates and languages
187 become deprecated, its backward compatibility might not be ensured. After a few years,
188 without maintenance and updates, there is a fair chance that it will stop working. In a
189 survey asking if the tool was still maintained, 60% of the tools' corresponding authors
190 either did not respond or answered negatively (Key figure 1H). As a direct
191 consequence, in some fields, development of new tools is needed due to a premature
192 obsolescence of existing ones.

The lack of incentive for long term maintenance is an issue common to all scientific software tools [35] (Box 3). Recently, to tackle this issue, a web platform (www.depsy.org) was created to quantify researchers' participations in software tool development and maintenance [36]. Although Depsy is currently limited to Python and R tools, it opens a window on better valorisation of scientific programming at large by providing a quantification of researchers' involvement. Other initiatives exist to increase the incentives, such as the Software Sustainability Institute, that aims at "cultivate better, more sustainable, research software to enable world-class research" (<https://www.software.ac.uk>). Ultimately, what is needed is an explicit valorisation of software tool maintenance and development by grant and promotion committees.

External hosting platform for longer lifespan

Even without better incentives, the lifespan of plant image analysis tools can be potentially increased using existing solutions. Here, two different issues need to be addressed: (i) the long-term availability and (ii) the long-term maintenance.

Today, many solutions exist to freely distribute stable versions of codes and software solutions. Here, I would like to strongly advertise in favour of external, open, hosting platforms (table 1). Hosting a tool on a dedicated webpage (e.g. on a personal or research group website) is, in theory, a good solution. However, there is a risk that the original link to the webpage changes over time (e.g. update of university website, change of website content manager or change of group name), leading to a disconnection between the original publication and the tool. As an example, out of the 97 academic tools from the plant-image-analysis.org database, 15% of the links

referenced in the papers were no longer active (Key figure 1G). The external platforms presented in table 1 are less likely to suddenly disappear or change in the near future, as they gather millions of users worldwide. In addition, these platforms provide other useful capabilities, such as backup, version control, and collaboration management, all of which ensure a better sustainability of the code. Since recently, Github (Table 1) also offers the possibility to permanently store code versions on Zenodo (Table 1), and therefore to associate them with a Digital Object Identifier (DOI). Other tools, such as Docker have been developed to ensure a better, easier and more stable dissemination of software tools [37].

The hosting platforms listed in table 1 strongly encourage users to share their code with the community. Different types of open-source licences are available (e.g. MIT, Apache Licence 2.0, GNU GPLv3) to accommodate specific needs and restrictions. Sharing the source code at the time of publication means that the whole community can help improve, test, and maintain the new tools. However, this is not currently the case in the plant science community and such a collaboration culture needs to be pushed forward. Pushing a step further, in addition to having the code stored on an external platform, tools can be deployed through Cyberinfrastructure projects, such as CyVerse (formerly iPlant) [38]. Although Cyberinfrastructure projects face specific challenges [17], they have the potential to ensure a longer lifespan for the tools they host, as they are, in theory, independent of operating systems and supported by external funding and grants. The DIRT project [6] is a successful example of an image analysis pipeline that is deployed as a free online service. It is now used by 213 scientists worldwide (with an average growing rate of 1 new user every 3 days) and the

development team is growing from 1 researcher to 1 software engineer, 1 post-doc and 2 to 3 PhD students (A. Bucksch, personal communication).

Concluding Remarks and Future Perspectives

There is no doubt that plant image analysis software is central in the plant biologist's toolbox, as reflected in the number of tools published in the recent years. However, the field of plant image analysis is still in its infancy and suffers from two main weaknesses: a lack of strong validation (and benchmarking) and a lack of long-term maintenance. However, I believe that simple steps can be taken to improve the current situation. These steps include creating and sharing better validation datasets as well as sharing and hosting codes on external, long term, platforms. I also believe such measures should be enforced by publishers and reviewers as they are the current gatekeepers of scientific publications.

Outstanding questions

- Can we (users, developers, reviewers and publishers) adopt publication standards to ensure a better validation and benchmarking of plant image analysis tools?
 - Can we, as a community, ensure long-term maintenance of plant image analysis tools, despite the lack of structural incentive to do so?
 - Can grant and promotion committees explicitly consider maintenance and improvement of existing tools in their evaluation?
-

Acknowledgements

The author would like to warmly thank Alexander Bucksch (University of Georgia, USA), Hannah Dee (Aberystwyth University, UK), Iko Koevoets (University of Amsterdam, The Netherlands) and Ruben Rellán-Alvarez (Langebio Cinvestav, Mexico) for their constructive comments and discussions on early version of the manuscript.

References

- 1 Vázquez-Arellano, M. *et al.* (2016) 3-D Imaging Systems for Agricultural Applications—A Review. *Sensors* 16, 618
 - 2 Fahlgren, N. *et al.* (2015) Lights, camera, action: high-throughput plant phenotyping is ready for a close-up. *Curr. Opin. Plant Biol.* 24, 93–99
 - 3 van Dusschoten, D. *et al.* (2016) Quantitative 3D Analysis of Plant Roots growing in Soil using Magnetic Resonance Imaging. *Plant Physiol.*
 - 4 Symonova, O. *et al.* (2015) DynamicRoots: A Software Platform for the Reconstruction and Analysis of Growing Plant Roots. *PLoS One* 10, e0127657–15
 - 5 Cai, J. *et al.* (2015) RootGraph: a graphic optimization tool for automated image analysis of plant roots. *J. Exp. Bot.* at <http://jxb.oxfordjournals.org/content/early/2015/07/28/jxb.erv359.short>
 - 6 Das, A. *et al.* (2015) Digital imaging of root traits (DIRT): a high-throughput computing and collaboration platform for field-based root phenomics. *Plant Methods* 11, 1–12
-

- 280 7 Bühler, J. *et al.* (2015) phenoVein—A Tool for Leaf Vein Segmentation and
 281 Analysis. *Plant Physiol.* 169, 2359–2370
 - 282 8 Biot, E. *et al.* (2016) Multiscale quantification of morphodynamics: MorphoLeaf,
 283 software for 2-D shape analysis. *Development*
 - 284 9 Wu, T.-C. *et al.* (2016) LobeFinder: A Convex Hull-Based Method for Quantitative
 285 Boundary Analyses of Lobed Plant Cells. *Plant Physiol.* 171, 2331–2342
 - 286 10 Knecht, A.C. *et al.* (2016) Image Harvest: an open-source platform for high-
 287 throughput plant image processing and analysis. *J. Exp. Bot.*
 - 288 11 Pound, M.P. *et al.* (2014) Automated Recovery of Three-Dimensional Models of
 289 Plant Shoots from Multiple Color Images. *Plant Physiol.* 166, 1688–1698
 - 290 12 McCormick, R.F. *et al.* (2016) 3D Sorghum Reconstructions from Depth Images
 291 Identify QTL Regulating Shoot Architecture. *Plant Physiol.* 172, 823–834
 - 292 13 Lobet, G. *et al.* (2013) An online database for plant image analysis software tools.
 293 *Plant Methods* 9, 1–7
 - 294 14 Mathieu, L. *et al.* (2015) Rhizoponics: a novel hydroponic rhizotron for root system
 295 analyses on mature *Arabidopsis thaliana* plants. *Plant Methods* 11, 1–7
 - 296 15 Fahlgren, N. *et al.* (2015) A Versatile Phenotyping System and Analytics Platform
 297 Reveals Diverse Temporal Responses to Water Availability in *Setaria*. *Mol. Plant* 8,
 298 1520–1535
 - 299 16 Wilson, G. *et al.* (2014) Best Practices for Scientific Computing. *PLoS Biol.* 12,
 300 e1001745
 - 301 17 Bucksch, A. *et al.* (2017) Overcoming the Law of the Hidden in Cyberinfrastructures.
 302 *Trends Plant Sci* 22, 117–123
-

- 303 18 Lobet, G. *et al.* (2017), Using a structural root system model to evaluate and
 - 304 improve the accuracy of root image analysis pipeline. *Front Plant Sci* 8, 447
 - 305 19 Pagès, L. *et al.* (2004) RootTyp: a generic model to depict and analyse the root
 - 306 system architecture. *Plant Soil* 258, 103–119
 - 307 20 Pagès, L. *et al.* (2013) Calibration and evaluation of ArchiSimple, a simple model of
 - 308 root system architecture. *Ecol. Modell.* 290, 76–84
 - 309 21 Leitner, D. *et al.* (2010) A dynamic root system growth model based on L-Systems.
 - 310 *Plant Soil* 332, 117–192
 - 311 22 Fournier, C. and Andrieu, B. (1998) A 3D Architectural and Process-based Model of
 - 312 Maize Development. *Ann. Bot.* 81, 233–250
 - 313 23 Boudon, F. *et al.* (2012) L-py: an L-system simulation framework for modeling plant
 - 314 architecture development based on a dynamic language. *Front. Plant Sci.* 3, 1–20
 - 315 24 Žádníková, P. *et al.* (2016) A Model of Differential Growth-Guided Apical Hook
 - 316 Formation in Plants. *Plant Cell* DOI: 10.1105/tpc.15.00569
 - 317 25 Fozard, J.A. *et al.* (2016) Hybrid vertex-midline modelling of elongated plant organs.
 - 318 *Interface Focus* 6, 20160043–20160014
 - 319 26 Tsaftaris, S.A. *et al.* (2016) Machine Learning for Plant Phenotyping Needs Image
 - 320 Processing. *Trends Plant Sci.* DOI: 10.1016/j.tplants.2016.10.002
 - 321 27 Pound, M.P. *et al.* 01-Jan-(2016) , Deep Machine Learning provides state-of-the-art
 - 322 performance in image-based plant phenotyping. , *bioRxiv*, 053033
 - 323 28 Ówiek-Kupczyńska, H. *et al.* (2016) Measures for interoperability of phenotypic data:
 - 324 minimum information requirements and formatting. *Plant Methods* 12, 44
 - 325 29 Minervini, M. *et al.* (2015) The significance of image compression in plant
-

phenotyping applications. *Funct. Plant Biol.*

30 Lobet, G. *et al.* (2015) Root System Markup Language: Toward a Unified Root
Architecture Description Language. *Plant Physiol.* 167, 617–627

31 Das, A. *et al.* (2014) ClearedLeavesDB: an online database of cleared plant leaf
images. *Plant Methods* 10, 1–8

32 Scharr, H. *et al.* (2015) Leaf segmentation in plant phenotyping: a collation study.
Mach. Vis. Appl. 27, 585–606

33 Morey, R.D. *et al.* (2016) The Peer Reviewers' Openness Initiative: incentivizing
open research practices through peer review. *R Soc Open Sci* 3, 150547

34 Kolesnikov, N. *et al.* (2015) ArrayExpress update—simplifying data submissions.
Nucleic Acids Res. 43, D1113–D1116

35 Howison, J. and Herbsleb, J.D. (2011) , Scientific software production: incentives
and collaboration. , in *Proceedings of the ACM 2011 conference on Computer
supported cooperative work*, pp. 513–522

36 Singh Chawla, D. (2016) The unsung heroes of scientific software. *Nature* 529,
115–116

37 Cito, J. and Gall, H.C. (2016) , Using Docker Containers to Improve Reproducibility
in Software Engineering Research. , in *Proceedings of the 38th International
Conference on Software Engineering Companion*, Austin, Texas, pp. 906–907

38 Merchant, N. *et al.* (2016) The iPlant Collaborative: Cyberinfrastructure for Enabling
Data to Discovery for the Life Sciences. *PLoS Biol.* 14, e1002342

39 Joppa, L.N. *et al.* (2013) Computational science. Troubling trends in scientific
software use. *Science* 340, 814–815

349 40 Baxter, S.M. *et al.* (2006) Scientific software development is not an oxymoron. *PLoS*
350 *Comput. Biol.* 2, e87

351 41 Crouch, S. *et al.* (2013) The Software Sustainability Institute: Changing Research
352 Software Attitudes and Practices. *Comput. Sci. Eng.* 15, 74–80

353

Key Figure 1. Overview of the development of plant image analysis tools. Data are coming from the papers referenced on the plant-image-analysis.org database (C,D,E,F,G) and from a survey send to the corresponding authors of the papers presenting the different tools (A,B,H). It should be noted here that these data are restricted to published tools, probably ignoring a large portion of tools that were never made public. All data and codes used for this figure are available at <https://doi.org/10.5281/zenodo.168216>

Table 1. Example of repositories for sharing large data sets and codes

Name	Primary use	URL	Price *	Size limitation	Possibility to have a DOI
ZENODO	data	www.zenodo.org	free	50GB / dataset	yes
FIGSHARE	data	www.figshare.org	free	5GB/file	yes
DRYAD	data	www.datadryad.org	120\$/ submission	20GB (expandable)	yes
ClearedLeavesDB	leaf data	clearedleavesdb.org	free	200GB/user	no
DIRT	root data	dirt.iplantcollaborative.org	free	200GB/user	no
Github	code	www.github.com	free	1GB	yes
Bitbucket	code	www.bitbucket.org	free	1GB	no
Sourceforge	code	www.sourceforge.net	free	(?)	no
CyVerse	Data	www.cyverse.org	free	1TB/dataset	yes

(*) Price are given for public datasets / codes. Price for private dataset / codes might differ

Box 1. Ground-truth is a complex concept

Image analysis aims at estimating properties (e.g. size, shape, number) of objects contained within images. The measurement made by the analysis algorithm must therefore obtain results as identical as possible as the true nature of the represented object (the ground-truth). For instance, for a sphere, the diameter estimated from a picture of the said sphere should be identical as its true diameter.

This is where the complexity of the ground-truth arises: how do we defined the true diameter of the sphere? Different options are possible: (i) we can measure the physical object (e.g. with a caliper), (ii) we can manually measure the object in the image (e.g with a line drawing tool) or (iii) we can create a computer model of the sphere, with a known diameter, and create an image from it. Unfortunately, none of these methods are perfect. Measuring the physical object is prone to human errors. Manually measuring the image is prone to both human errors and image distortion during the acquisition process. Modelled images are usually noiseless, and therefore likely to be unrepresentative of the experimental images.

Thus, defining the ground-truth of an image is not straightforward, especially for complex objects. The methods used for ground-truthing should therefore always be described, to avoid misinterpretations

Box 2. Better validation and longer maintenance: some guidelines

Better validation

Larger datasets

- Use modelling tools to generate images
- Use 3D-printed models imaged in conditions as realistic as possible

Share datasets on a public repository:

- Images that work, but also some that do not work with the tools
- Information about the experimental conditions
- Information about the image acquisition
- Ground-truth data in a parsable file type

Longer lifespan

- Make the tool freely available for the community
- Open the code and store it on a public repository
- Create a stable version of the tool at the time of publication.
- Embed the tool in a Cyberinfrastructure Project, such as CyVerse
- Provide ample documentation on how to install and use the tool

Box 3: Short software lifecycle: a common theme in scientific computing

Lack of sustainability, reproducibility and long-term maintenance is not a challenge specific to plant image analysis tools. The scientific computing community at large is affected by the issue [39-40]. Several projects have emerged to face the problem, such as the Software Sustainability Institute in the United Kingdom [41] or the Software Carpentry Foundation in the United States (<https://software-carpentry.org/>). The aim of both organisations is to improve computing skills in the scientific community at large and promote scientific software.

Several factors have been identified as critical to increase the software lifecycle [40-41]:

- The **technical debt** of the developing team: many scientific software developers are not trained computer scientists. This might lead to technical limitations in the tool development that might be difficult to overcome and lead to the arrest of the project.
- **Lack of long-term project management**: computing projects are often seen as side projects and are not given enough importance in the lab. Proper management, as for any other type of research, is needed.
- **Lack of proper validations and quality controls**
- **Lack of sufficient documentation**

Glossary

Validation: The validation step aims at determining how well a specific tool is able to describe the analysed object.

Benchmarking: Benchmarking a new tool consist of comparing its performance with existing tools. Performance can take different meanings, such as the accuracy of the measurements, the speed of the analysis or the ease of use.

Ground-truth: The ground-truth is the true nature of the object represented in the image. See Box 1 for details.

