Correspondence

BioImageIT: Open-source framework for integration of image data management with analysis

Check for updates

To the Editor—Advances in bioimaging over the past 20 years have been accompanied by developments in computational approaches for image reconstruction, analysis, classification and interpretation. Bioimaging has a broad range of applications addressing a variety of biological models at diverse scales of life; thus, descriptions of novel computational approaches are often focused on target case studies. Consequently, the conception and the development of a unified solution, able to tackle any scenario in biological imaging, are major challenges. Several types of architecture and tools have been proposed to surmount these technological difficulties. Although moving in the right direction, the existing software platforms (such as Fiji[1], Icy[2] and CellProfiler[3]), developed in various programming languages, are not all interoperable. Additional code development efforts are needed to gather various heterogeneous image-processing components in ad hoc workflows.

At the same time, although data storage on dedicated database servers, such as OMERO[4] and others (for example, see https://imagerie.cochin.inserm.fr/sis4web/login.php), is becoming widespread, interaction with processing and analysis tools could benefit from further improvement. A more integrated approach to data organization, visualization and analysis is required to avoid the tedious task of manual management or scripting. In addition, managing often massive datasets requires dedicated expertise in computer science to scale up storage and computational resources. Now, with the emergence of artificial intelligence, such as deep learning, in bioimaging (for example, ImJoy[5]), the automation of processing tasks and the implementation of analysis pipelines that include image visualization (such as napari[6]), it is necessary to consider all stages of the data life cycle and new human–machine interactions. It is worth noting that data handling must now meet high quality criteria that will ensure identification, accessibility and interoperability of data with

their processing, storage and analysis. These 'FAIR' principles[7] impose new procedures and ethical obligations on scientists whose research relies on biological imaging, leading to a paradigm shift in the production of knowledge through image interpretation.

To meet all these requirements, we developed BioImageIT, a unique open-source system integrating imaging data management and analysis, and an operational solution for handling large datasets in line with open science requirements (Fig. 1a). In BioImageIT, data are automatically annotated and processed in a single framework. Although the extent of its flexible design remains to be fully exploited, BioImageIT allows the integration of any existing data-management and image-processing software. For instance, data could be hosted on an OMERO[4] database using Bio-Formats[8] and processed by deep neural networks using TensorFlow[9]. Unlike previous frameworks (for example, Galaxy Project[10]), BioImageIT addresses the following end-user issues.

BioImageIT reconciles data management with analysis in a common interactive framework
Most open-source bioimaging software is developed separately and specialized for either data management or data analysis. Therefore, users must write ad hoc scripts or apply manual operations to process the data. In contrast, BioImageIT allows import and tagging of data. Each operation automatically generates metadata that facilitate keeping track of analysis steps and are compliant with the recommended FAIR principles[7].

BioImageIT is interoperable and reusable
Processing tools are pre-packaged and stored in public repositories. Thus, users can reuse them and create data analysis workflows with software developed in any language.

BioImageIT is developer friendly
BioImageIT makes it easy for data scientists to distribute new tools embedded in a package.

Only a basic configuration file (wrapper) is required for identification in BioImageIT.

BioImageIT is user focused
BioImageIT consists of three layers: back-end plugins, a python application programming interface (API) and a graphical user interface (GUI). Users can choose the most appropriate level of interaction. Biologists may prefer to be assisted step by step, in which case a GUI is appropriate. Data analysts familiar with writing scripts may use the python API. Data scientists can adopt the packaging back-end to provide a stand-alone demonstration of their new processing tool.

In summary, BioImageIT is a generic framework for managing, analyzing and ensuring traceability except at this stage for patient-sensitive data. Unlike previous platforms, it addresses the needs of end users and provides a flexible solution to link annotated data with processing tools. It facilitates interactions between experimental and data scientists. Because BioImageIT is built upon existing technologies, it may be considered as a computational overlay providing a user-friendly interface for existing software to end users, without hindering the addition of new analytical methods by experienced developers. Finally, the ability of BioImageIT to integrate with data interaction tools allows highly specialized examples to be deployed for a specific application domain, as illustrated here for lattice light sheet microscopy data processing (Fig. 1b–i; see other case studies in Supplementary Figs. 1 and 2). These examples demonstrate how BioImageIT creates sophisticated image-analysis workflows for datasets obtained using advanced microscopic techniques. BioImageIT is being deployed on ten imaging platforms covering a broad perimeter of the France-BioImaging national infrastructure (https://bioimageit.github.io/#/about). With some of them, we are currently developing a BioImageIT python API to run analysis pipelines from their own database GUI. To facilitate or generalize this use, including with OMERO[4], a Java or a REST API
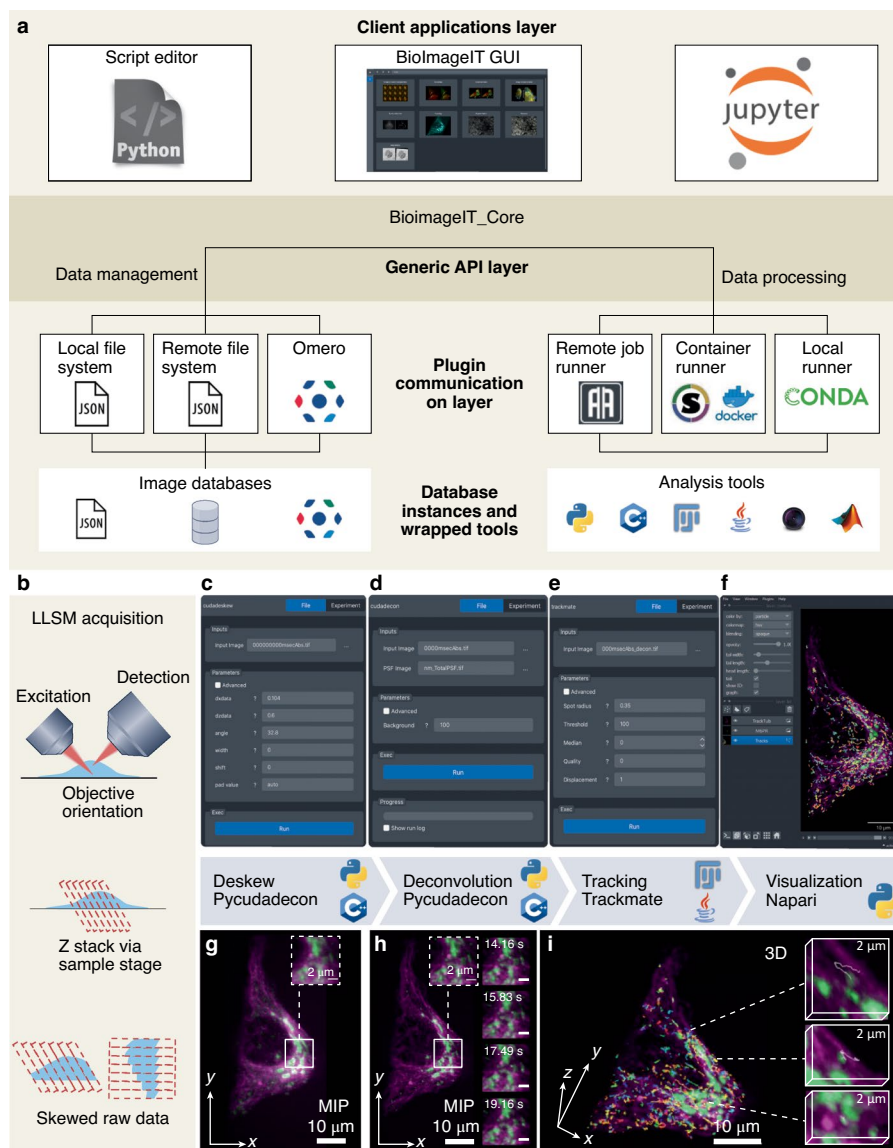
Fig. 1 | BioImageIT overview. a, Schematic view of BioImageIT architecture. The BioImageIT core comprises data-management and data-processing functionalities. Users can access plugins through a script editor, the BioImageIT graphical interface or the Jupyter platform. Data-management functionalities exploit local files, remote files or the OMERO database. Data processing can perform computations in remote jobs, containers or local runners. Image analysis is provided by plugins, which can be written in different programming languages. Developers can implement their own plugins in BioImageIT and design their own graphical interface. **b**–**i**, Example of lattice light shee microscopy workflow for 3D reconstruction and tracking of a live genome-edited HeLa cell (CD-M6PR-eGFP) stained with Tubulin Tracker Deep Red for Microtubules[11]. **b**, Because of the geometry of lattice light sheet microscopy (LLSM) scanning, raw 3D images are skewed. **c,g**, First, realignment (deskew) of raw stacks is performed using the software package pycudadecon. **d,h**, Richardson–Lucy deconvolution is performed using pycudadecon. **e**, CD-M6PR-eGFP vesicles are tracked[11] using Trackmate. **f,i**, Deconvolved stacks and tracks are rendered using napari. The full workflow, including pycudadecon, Trackmate and napari, was gathered in BioImageIT.

will be developed. With this ongoing phase, using imaging core facilities as pillars, our primary goal is to build a BioImageIT community.

Reporting summary

Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

The data generated during this study are available in Figshare with the identifier https://doi.org/10.6084/m9.figshare.19375598 (ref. [11]).

Code availability

BioImageIT source code used in this publication is open source and published under the BSD 4-Clause "Original" or "Old" License. The source code is available through GitHub: https://github.com/bioimageit. The latest stable releases used in this publication, and the current versions, which include bug fixes and updates, are available on the Zenodo repository (ref. [12]). Details of how to use the software are described on the project's website:

Correspondence

https://bioimageit.github.io/. The Omero plugin is licensed under the GNU General Public License v2.0.

Sylvain Prigent [1,2,3],
Cesar Augusto Valades-Cruz [1,2,3],
Ludovic Leconte[1,2,3], **Léo Maury** [1,2],
Jean Salamero [1,2] ✉ and
Charles Kervrann [1,2] ✉

[1]SERPICO Project Team, Inria Centre Rennes-Bretagne Atlantique, Rennes, France. [2]SERPICO Project Team, UMR144 CNRS Institut Curie, PSL Research University, Paris, France. [3]These authors contributed equally: Sylvain Prigent, Cesar Augusto Valades-Cruz, Ludovic Leconte.
✉e-mail: jean.salamero@curie.fr; charles.kervrann@inria.fr

References

1. Schindelin, J. et al. *Nat. Methods* **9**, 676–682 (2012).
2. de Chaumont, F. et al. *Nat. Methods* **9**, 690–696 (2012).
3. Stirling, D. R., Carpenter, A. E. & Cimini, B. A. *Bioinformatics* **37**, 3992–3994 (2021).
4. Allan, C. et al. *Nat. Methods* **9**, 245–253 (2012).
5. Ouyang, W., Mueller, F., Hjelmare, M., Lundberg, E. & Zimmer, C. *Nat. Methods* **16**, 1199–1200 (2019).
6. Sofroniew, N. et al. napari/napari: 0.4.12rc2, https://doi.org/10.5281/ZENODO.5587893 (2021).
7. Wilkinson, M. D. et al. *Sci. Data* **3**, 160018 (2016).
8. Linkert, M. et al. *J. Cell Biol.* **189**, 777–782 (2010).
9. Abadi, M. et al. Preprint at *arXiv* https://arxiv.org/abs/1603.04467 (2016).
10. Goecks, J., Nekrutenko, A., Taylor, J. & Galaxy Team. *Genome Biol.* **11**, R86 (2010).
11. Prigent, S., Valades-Cruz, C. A., Leconte, L., Salamero, J. & Kervrann, C. *BioImageIT: Demo Datasets* https://doi.org/10.6084/m9.figshare.19375598 (2022).
12. Prigent, S. et al. *BioImageIT* https://doi.org/10.5281/zenodo.6361881 (2022).