Check for updates

SOFTWARE TOOL ARTICLE

REVISED MosaicExplorerJ: Interactive stitching of terabyte-size tiled datasets from lightsheet microscopy [version 2; peer review: 2 approved]

Sébastien Tosi [1], Lídia Bardia[1], Maria Jose Barallobre[2], Arrate Muñoz-Barrutia[3,4], María Luisa Soto-Montenegro[4,5], Julien Colombelli [1]

[1]Institute for Research in Biomedicine - IRB Barcelona, the Barcelona Institute for Science and Technology - BIST, Barcelona, Spain
[2]Department of Developmental Biology, Instituto de Biología Molecular de Barcelona, CSIC, Barcelona, Spain
[3]Departamento de Bioingeniería e Ingeniería Aeroespacial, Universidad Carlos III de Madrid, Leganés, Spain
[4]Instituto de Investigación Sanitaria Gregorio Marañón, Madrid, Spain
[5]CIBER de Salud Mental (CIBERSAM), Madrid, Spain

Abstract

We introduce MosaicExplorerJ, an ImageJ macro to stitch 3D tiles from terabyte-size microscopy datasets organized on a regular 2D grid. As opposed to existing software, stitching does not require any prior information on the actual positions of the tiles, or conversion of raw TIFF images to a multi-resolution format for interactive exploration and fast processing. MosaicExplorerJ was specifically designed to process lightsheet microscopy datasets from optically cleared samples. It can handle multiple fluorescence channels, dual-sided lightsheet illumination and dual-sided camera detection.

Keywords

3D stitching, Lightsheet microscopy, Tiled scan, Mosaic, ImageJ

This article is included in the NEUBIAS - the Bioimage Analysts Network gateway.

Open Peer Review

Reviewer Status ✓ ✓

	Invited Reviewers	
	1	**2**
version 2 (revision) 04 Feb 2021	✓ report	✓ report
	↑	↑
version 1 10 Nov 2020	? report	? report

1. **Stephan Preibisch** [iD], Berlin Institute for Medical Systems Biology, Berlin, Germany HHMI Janelia, Ashburn, USA

2. **Francesco Pampaloni** [iD], Goethe University Frankfurt, Frankfurt am Main, Germany

Any reports and responses or comments on the article can be found at the end of the article.

Corresponding author: Sébastien Tosi (sebastien.tosi@irbbarcelona.org)

Author roles: Tosi S: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Project Administration, Software, Supervision, Validation, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Bardia L**: Resources, Validation, Writing – Original Draft Preparation, Writing – Review & Editing; **Barallobre MJ**: Resources, Validation, Writing – Review & Editing; **Muñoz-Barrutia A**: Resources, Validation, Writing – Review & Editing; **Soto-Montenegro ML**: Resources, Validation, Writing – Review & Editing; **Colombelli J**: Data Curation, Funding Acquisition, Investigation, Project Administration, Resources, Supervision, Validation, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing

Introduction

A number of open source tools are available to stitch mosaics from optical microscopy 3D tiled scans[1–4] but they systematically implement automated algorithms which results might depend on the starting positions of the tiles and potentially converge to a suboptimal solution. This is especially likely if the initial positions are far from the optimal positions, or if the data suffers from unexpected artifacts. Even worse, this situation can be difficult to detect in practice since these tools often bring no or scarce support to check the results and finely correct for the observable residual errors manually. Additionally, some software limits the size of the datasets[1,2], or requires the data to be redundantly converted to ad-hoc formats[3]. Finally, until now only BigStitcher[4] could handle dual-sided illumination and dual-sided camera detection, two useful lightsheet microscopy[5] modalities that can advantageously be combined (Figure 1, Left). We developed MosaicExplorerJ to address these shortcomings and bring a complementary alternative to ImageJ BigStitcher, the reference in the field.

Methods
Implementation

Whereas stitching the tiles of confocal microscopy datasets chiefly consists in compensating for the scanning head to sample stage tilt (assuming a perfectly orthogonal XYZ translation system), stitching lightsheet microscopy datasets is compounded by the fact that the lightsheet is not necessarily perfectly collinear to the object plane of the detection objective. This can lead to 1) a distortion of the aspect ratio of the images (often negligible), 2) an apparent axial displacement of the tiles while moving across the mosaic. This second effect can be simply compensated by axially offsetting the 3D tiles accordingly (Figure 1, Middle), but additional lightsheet non-uniformity (or lateral misalignment) can lead to differences in the features visible in the regions of tile overlap; potentially weakening correlation based stitching algorithms. To address these issues, MosaicExplorerJ assists the user in visually aligning the tiles along the possible degrees of freedom set by a predefined physical model by following a step-by-step procedure to compensate for mismatches highlighted in the regions of tiles overlap.

Operation

ImageJ/Fiji should be installed and MosaicExplorerJ run from ImageJ macro editor. If no other data is available, the software can be tested with the sample data provided (Extended Data S1[6]). First, the mosaics from both cameras (and illumination sides) should be aligned independently before being aligned together. This first operation can be as simple as joining two matching features between two adjacent tiles (Figure 2B, Extended Data S3-V1[6]), while compensating lightsheet tilt (Figures 2C, 2D) and the axial wobbling of the motors forming the mosaics might require to adjust the axial shift of a selected number of tiles in the top row/column of the mosaic (Extended Data S3-V2[6]). Dual-sided camera datasets alignment includes an extra calibration step to compensate for discrepancy between the magnifications of both detection objective lenses (Figure 1, Right, Extended Data S3-V3[6]), while dual-sided
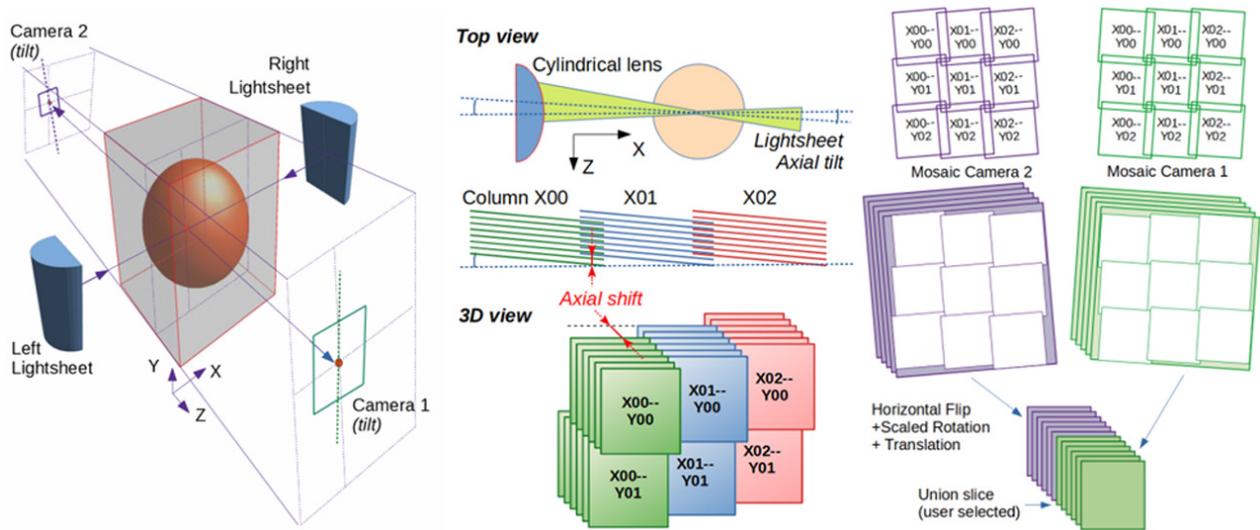
Figure 1. Left: Illuminating the sample and collecting the light from both sides enables to image a sample by lightsheet microscopy in the best conditions. **Middle**: Overlapping tiles (green, blue, red) from a 3D mosaic are shifted axially to compensate for the tilt of the lightsheet (here exaggerated around Y axis). **Right**: Reconstructed mosaics from both cameras are aligned before stacking their best section.
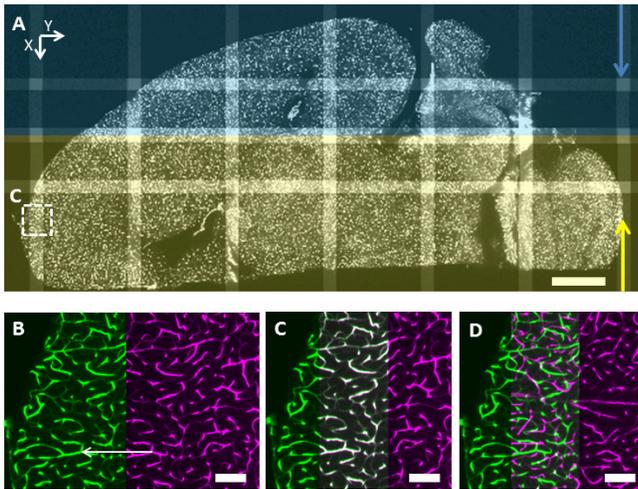
Figure 2. A: Two mosaics (2x8 tiles each with a full overlap in the central column along Y) from a dual-sided illumination dataset aligned with MosaicExplorerJ, the arrows show the directions of the light sheets (the view is turned and cropped to fit the figure). Scale bar: 1 mm. **B**: Zoomed region from the red square prior to adjusting the XY positions of the tiles but after axial shift (Z) correction. The user joins two matching features (white arrow) from two adjacent tiles. **C**: The XY positions of the tiles are adjusted based on the previous user input and for correct alignment matching features are highlighted in white in the region of overlap. **D**: Same view but prior to axial shift correction to compensate for lightsheet tilt: no matching features are apparent in the region of overlap. Scale bar: 100 μm.

illumination datasets can be stitched by following a similar procedure (Extended Data S3-V4[6]). Intensity correction can be achieved from either external correction masks or interactively adjusted from XY separable shading correction masks (Extended Data S3-V5[6]). Finally, the user can save the overall mosaic alignment for further inspection, or export the stitched dataset as a TIFF series.

Testing
MosaicExplorerJ has been extensively tested by stitching several large lightsheet microscopy datasets acquired from diverse optically cleared biological samples (see *Data availability*).

Use cases
All datasets were successfully aligned, each time in under 30 minutes. The results were checked visually by scrolling through the slices and ensuring that the alignment was correct in the regions of overlap between the tiles. The dataset Brain2_izq_2x8Mosaic_LeftSide_300GB was also aligned by BigStitcher, leading to similar results both visually and quantitatively (Extended Data Table S1[6]). It took about 2h30 to process this dataset with BigStitcher (from 3D TIF tiles to computed alignment), excluding the conversion from TIFF series (the original format of this dataset) to 3D TIFF tiles. This time

is expected to scale at least linearly with increasing dataset size for BigStitcher while alignment time in MosaicExplorerJ is relatively constant, and mostly conditioned by the degrees of freedom of the 3D tiles. After alignment, the exportation of the stitched images to TIFF series could be achieved in a comparable time with both tools.

Conclusion
Even though it only supports a subset of its features, MosaicExplorerJ brings a complementary alternative to BigStitcher and presents a number of advantages (Extended Data S2[6] and Table S2[6]). No fiducials or detectable feature points are required, which makes the tool robust, versatile, and compatible with large optically cleared samples for which introducing fiducials is virtually impossible. The processing is fast, and terabyte-size datasets can be explored on the fly without conversion to an intermediate format, even on laptop computers with limited memory. Finally, all alignment steps are performed visually, which brings direct control and feedback both on the imperfections of the datasets and on the quality of the results, minimizing the risk of leaving coarse errors unnoticed.

Data availability
Underlying data
A complete description of the datasets used to test MosaicExplorerJ, including sample preparation and imaging can be found in Extended Data S1[6]. The full datasets are too large (1.5TB) to feasibly host on a data repository; however, datasets can all be accessed publicly on the IRB Barcelona Google Drive at: https://bit.ly/37iocrP.

Extended data
Zenodo: MosaicExplorerJ F1000Research article extended data. 10.5281/zenodo.4455761[6].

Data are available under the terms of the Creative Commons Attribution 4.0 International license (CC-BY 4.0).

Software availability
Source code and documentation are available from: https://github.com/SebastienTs/MosaicExplorerJ

Archived source code at time of publication: http://www.doi.org/10.5281/zenodo.4453506[7].

License: GNU General Public License v3.0.

References

1. Emmenlauer M, Ronneberger O, Ponti A, *et al.*: **XuvTools: free, fast and reliable stitching of large 3D datasets.** *J Microsc.* 2009; **233**(1): 42–60.
   PubMed Abstract | **Publisher Full Text**

2. **https://imagej.net/Grid/Collection_Stitching_Plugin**.

3. Bria A, Iannello G: **TeraStitcher - a tool for fast automatic 3D-stitching of teravoxel-sized microscopy images.** *BMC Bioinformatics.* 2012; **13**: 316.
   PubMed Abstract | **Publisher Full Text** | **Free Full Text**

4. Hörl D, Rojas Rusak F, Preusser F, *et al.*: **BigStitcher: reconstructing high-resolution image datasets of cleared and expanded samples.** *Nat Methods.* 2019; **16**(9): 870–874.
   PubMed Abstract | **Publisher Full Text**

5. Huisken J, Swoger J, Del Bene F, *et al.*: **Optical sectioning deep inside live embryos by selective plane illumination microscopy.** *Science.* 2004; **305**(5686): 1007–1009.
   PubMed Abstract | **Publisher Full Text**

6. Tosi S, Bardia L, Colombelli J, *et al.*: **MosaicExplorerJ F1000Research article extended data (Version 2.0).** *Zenodo.* 2021.
   http://www.doi.org/10.5281/zenodo.4455761

7. Tosi S: **SebastienTs/MosaicExplorerJ: Revised Release (Version 1.5).** *Zenodo.* 2021.
   http://www.doi.org/10.5281/zenodo.4453506

Open Peer Review

Current Peer Review Status: ✔ ✔

Version 2

Reviewer Report 11 March 2021

https://doi.org/10.5256/f1000research.54151.r78912

✔ **Stephan Preibisch** [iD]
[1] Berlin Institute for Medical Systems Biology, Berlin, Germany
[2] HHMI Janelia, Ashburn, VA, USA

Thanks so much to the authors for addressing my concerns, I think their work is a useful addition for lightsheet reconstruction.

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Image analysis, image registration, lightsheet

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Reviewer Report 01 March 2021

https://doi.org/10.5256/f1000research.54151.r78911

✔ **Francesco Pampaloni** [iD]
Buchmann Institute for Molecular Life Sciences, Goethe University Frankfurt, Frankfurt am Main, Germany

The authors added links to five video tutorials in the documentation available on GitHub. The videos greatly support the first user of the MosaicExplorerJ macro in understanding the different steps to align and merge the tiles and to perform the registration of dual-sided LS illumination and dual-sided detection data sets. The changes and the addition in the documentation made by

the authors in the reviewed version significantly reduce the time the user has to invest to learn how to apply the macro to her/his own data sets. Especially after following the explanatory videos, all the functions of the macro can be straightforwardly used. Overall, a user-friendly and very useful macro for the light sheet microscopy community.

Detected bugs:
- After pressing Cancel, it is not possible to re-open the MosaicExplorerJ GUI with Alt.

- The "Register OvIX & YCor -> OvIY &XCor" function works correctly only if the selection line is drawn from east -> west and not vice versa.

- The "Register OvY XCor only" function works correctly only if the selection line is drawn from the bottom to the top (or south -> north), but not vice versa.

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Light Sheet Microscopy, 3D cell biology, stem cell and tumor organoids.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

- -

Version 1

Reviewer Report 11 December 2020

https://doi.org/10.5256/f1000research.29948.r75056

? **Francesco Pampaloni** iD

Buchmann Institute for Molecular Life Sciences, Goethe University Frankfurt, Frankfurt am Main, Germany

Overview

The MosaicExplorerJ ImageJ macro-code described in the manuscript is a useful tool to stitch very large data sets recorded with light sheet microscopy. Indeed, the amount of data acquired in a typical session of light-sheet imaging of optically cleared organs (e.g. the mouse brain), reaches easily the TB range with up-to-date equipment. Therefore, user-friendly and straightforward data navigation and stitching software is needed by the scientific community. MosaicExplorerJ, at variance with automatic stitching algorithms such as BigStitcher, it is a fully manual tool, requiring the user manually perform all the alignment steps in order to compensate the tiles mismatch, e.g. the tiles axial shift arising from a tilted light sheet. Moreover, also at variance with BigStitcher,

multi-tiff image stacks can be opened directly without conversion.
I recommend the manuscript for idexing with minor changes and additions as described in the following remarks.

Remarks

The dataset deposited by the authors on the IRB Barcelona server allows readily testing the macro. The macro runs from the macro editor. The dataset is quickly loaded into the main window. In order to properly upload the tiles, the different files have to be named sequentially following the explained rule. After opening the dataset, the alignment and stitching are performed by manually setting the shifting parameters (xy translation and axial alignment along z) shown in the control panel.

In general, a first-time user can achieve the stitching of the mosaic 3D dataset as described in the manuscript by following the steps outlined in the documentation available on GitHub. However, the learning curve is quite steep and some effort and time is necessary by the user in order to understand the working of the macro and to achieve the result. In order to allow for a smoother learning curve, I strongly recommend the authors to upload videos illustrating the stitching procedure for two datasets at increasing difficulty.

Particularly, the following step described in the documentation (p. 3, end) is not clearly explained, and should be better illustrated with a further figure and in a video:

"For linear correction, it is recommended to use the top left tile as reference and gradually tweak ZxCor from the tile to the right, until some matching features are apparent in the overlap region. Then ZyCor can be adjusted, this time by considering the tile below. The adjustment of the Z offsets can be performed by typing values in the Panel or by following the procedure described below for free correction."

I also recommend to include the documentation in the manuscript extended material as well, and not exclusively in GitHub, as the macro "instructions" are an essential part of the work.

Is the rationale for developing the new software tool clearly explained?
Yes

Is the description of the software tool technically sound?
Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Partly

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Yes

Are the conclusions about the tool and its performance adequately supported by the

findings presented in the article?

Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Light Sheet Microscopy, 3D cell biology, stem cell and tumor organoids.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 21 Jan 2021

Sebastien Tosi, Institute for Research in Biomedicine - IRB Barcelona, the Barcelona Institute for Science and Technology - BIST, Barcelona, Spain

Thank you for reviewing the article and for pointing out these weaknesses of the documentation. We have tried to substantially improve the documentation accordingly and we have created five video tutorials to help illustrate all the steps described in the documentation. Please see our detailed answers below.

> In general, a first-time user can achieve the stitching of the mosaic 3D dataset as described in the manuscript by following the steps outlined in the documentation available on GitHub. However, the learning curve is quite steep and some effort and time is necessary by the user in order to understand the working of the macro and to achieve the result. In order to allow for a smoother learning curve, I strongly recommend the authors to upload videos illustrating the stitching procedure for two datasets at increasing difficulty.

We have included five video tutorials that illustrate the main operations and complement the documentation. These videos are available from the GitHub repository and from the new Extended Data **S3**.

> Particularly, the following step described in the documentation (p. 3, end) is not clearly explained, and should be better illustrated with a further figure and in a video: "For linear correction, it is recommended to use the top left tile as reference and gradually tweak ZxCor from the tile to the right, until some matching features are apparent in the overlap region. Then ZyCor can be adjusted, this time by considering the tile below. The adjustment of the Z offsets can be performed by typing values in the Panel or by following the procedure described below for free correction."

We have completely rewritten this part of the documentation and the new video tutorial **S3-V2** is fully dedicated to explaining this critical step.

> I also recommend to include the documentation in the manuscript extended material as well, and not exclusively in GitHub, as the macro "instructions" are an essential part of the work.

> We created a new Extended Data **S3** with links to the documentation of the software and to the five new video tutorials. Both are also available from the GitHub repository.
>
> **Competing Interests:** No competing interests were disclosed.

Reviewer Report 07 December 2020

?

Stephan Preibisch (iD)
1 Berlin Institute for Medical Systems Biology, Berlin, Germany
2 HHMI Janelia, Ashburn, VA, USA

The authors present MosaicExplorerJ, an ImageJ macro for manual alignment of tiled lightsheet acquisitions.

I think it can be useful as it provides simple, editable source code. I personally found it quite hard to use, but more practice might make it more efficient. But more explanations of the parameters and keyboard shortcuts in the application would certainly be helpful (press F1 or similar).

Most importantly, the paper should clearly state the limitations of the software and correct wrong statements about the comparison to other tools, in particular BigStitcher from my point of view.

Not mentioned limitations (or implicitly assumed ones):
- all files need to have the same dimensions (also in z)

- only 3D translation models are supported

- no chromatic aberration correction

- maximal xy output size is 2^31 pixels, the limitation of ImageJ (around 47000x47000 pixels)

Incorrect comparisons that should be corrected:
1) "As opposed to existing software, stitching does not require any prior information on the actual positions of the tiles, sample fiducials, or conversion of raw TIFF images, and the stitched images can be explored instantly."
- That is the same in BigStitcher, conversion to HDF5 is not necessary, TIFFs can be virtually loaded and displayed instantaneously

2) "but they all implement automated algorithms … Even worse, this situation can be difficult to detect in practice since these tools bring no or scarce support to check the results and correct for errors manually."
- BigStitcher provides user friendly methods for manual & interactive alignment (3D

translation): https://imagej.net/BigStitcher_manual_translation

3) "Finally, none of these solutions handles both dual-side illumination and dual-side camera detection, two useful lightsheet microscopy"

- BigStitcher supports dual-side illumination and dual-side camera detection (e.g. Supplementary Figure 17)

4) S2-Comparison with BigStitcher:

- 3D TIFF tiles can be natively used through virtual loading, live display in BDV even. HDF5 conversion is optional (yet very useful)

- 2D TIFF slices are also supported, but it's a quite hidden feature]

- BigStitcher supports Dual-camera support

- Physical model constrained grid alignment is also possible, see link above, even interactive & manual

- Automatic color highlighting of tiles mismatch can be done in BigStitcher

- BigStitcher additionally supports per-tile intensity correction using Affine 1D models

Is the rationale for developing the new software tool clearly explained?

Partly

Is the description of the software tool technically sound?

Partly

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Image analysis, image registration, lightsheet

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Author Response 21 Jan 2021

Sebastien Tosi, Institute for Research in Biomedicine - IRB Barcelona, the Barcelona Institute for Science and Technology - BIST, Barcelona, Spain

First of all, we thank the reviewer for his thorough comments and we would like to apologize for overlooking some of BigStitcher's features that we were genuinely unaware of. Also, as rightly pointed out by the reviewer, some limitations of MosaicExplorerJ were not discussed in the article. We have remedied this, as described point by point in our answer, and we hope that the features and the operating workflow of the two applications are now reflected more accurately in the fully updated Extended Data **S2**.

> More explanations of the parameters and keyboard shortcuts in the application would certainly be helpful (press F1 or similar).

We have tried our best to improve this aspect in MosaicExplorerJ (v1.5, now on GitHub): a help window can now be displayed by pressing F1, contextual information is provided in ImageJ status bar (and strategically in some of the dialog boxes), and the manual operation to perform axial adjustment of the tiles has been simplified. Finally, five video tutorials (bit.ly/2KY0BDN) were created to illustrate the main operations and to complement the documentation.

> Not mentioned limitations (or implicitly assumed ones) of MosaicExplorerJ:

1. All files need to have the same dimensions (also in z)Actually, this is not strictly required as missing tiles or tiles missing Z slices (for 2D TIFF series mode) will not trigger an error but the corresponding images will just be displayed black. However, to answer this comment, we now state in the Abstract of the article (and in Extended Data **S2**) that the tiles should be organized on a regular 2D grid.

2. Only 3D translation models are supported

Yes, only these operations are supported for a single camera dataset but 2D similarity (rotation, scaling, translation) is performed (after 3D translations) when registering the mosaics from two opposing cameras (as described in **Figure 1**).

3. No chromatic aberration correction

We now explicitly state in Extended Data **Table S2** that chromatic aberrations correction is not supported by MosaicExplorerJ (while it is supported by BigStitcher).

4. Maximal xy output size is 2^31 pixels, the limitation of ImageJ (around 47000x47000 pixels)

We now explicitly state in Extended Data **Table S2** that mosaics stitched by MosaicExplorerJ are limited to about 47.000 x 47.000 pixels (and eight channels) while BigStitcher sets no limit. We haven't included this information in the text of the main article as we believe that it

is not really limiting in practice.

Incorrect comparisons that should be corrected:

1) "As opposed to existing software, stitching does not require any prior information on the actual positions of the tiles, sample fiducials, or conversion of raw TIFF images, and the stitched images can be explored instantly."
That is the same in BigStitcher, conversion to HDF5 is not necessary, TIFFs can be virtually loaded and displayed instantaneously.

We apologize for this misleading statement and we were unaware that tiles could be loaded as a set of 2D TIFF series in BigStitcher. Still, we believe that HDF5 conversion is implicitly required for efficient operations when handling TB datasets, both for interactive 3D viewing and to enable a practical processing time (especially when several rounds of tile placement / parameters optimizations have to be performed to get to satisfying results). We updated this paragraph of the text to reflect this idea, and we stated in Extended Data **Table S2** that while not strictly required HDF5 conversion is desirable to speed up operations.

2) "but they all implement automated algorithms ... Even worse, this situation can be difficult to detect in practice since these tools bring no or scarce support to check the results and correct for errors manually."
BigStitcher provides user friendly methods for manual & interactive alignment (3D translation): https://imagej.net/BigStitcher_manual_translation

We were aware of this feature but, as far as we can tell, tiles are first automatically laid down to a regular grid (with fixed overlap), and then only individual tiles (or whole groups) can be interactively moved. This feature is certainly useful to coarsely set the initial positions of the tiles but not so much to finely adjust the mosaic (that would ultimately require to finely move tiles on a tile by tile basis). In turn, thanks to the predefined physical correction models, as few as two pairs of tiles need to be manually adjusted in MosaicExplorerJ to stitch a whole mosaic (with no further processing). We tried to reflect this more accurately in this paragraph and in Extended Data **Table S2**.

3) "Finally, none of these solutions handles both dual-side illumination and dual-side camera detection, two useful lightsheet microscopy"
BigStitcher supports dual-side illumination and dual-side camera detection
(e.g. Supplementary Figure 17)

We apologize for this plainly wrong statement (we did not know that BigStitcher supported dual-sided cameras as the option is not listed in the image loader). We modified the text to exclude BigStitcher from the set of software not supporting the combination of these two modalities and we reflected these features in details in Extended Data **S2** and Extended Data **Table S2**.

4) S2-Comparison with BigStitcher:
3D TIFF tiles can be natively used through virtual loading, live display in BDV even. HDF5 conversion is optional (yet very useful)

2D TIFF slices are also supported, but it's a quite hidden feature

Extended Data **Table S2** has been updated to reflect this.

BigStitcher supports Dual-camera support

Extended Data **Table S2** has been updated to reflect this.

Physical model constrained grid alignment is also possible, see link above, even interactive & manual

For the mosaic stitching part only (excluding cameras registration), MosaicExplorerJ physical correction models include camera tilt (XY translations), lighshsheet tilt (linearly increasing Z translations) and motor axis axial wobble (XY separable Z translations). BigStitcher global tile layout only supports a regular grid with fixed overlap (not accounting for previously mentioned effects). In our view, this is useful to coarsely set the starting positions of the tiles prior to optimization but it is really practical to finely adjust the positions of the tiles of large mosaics with no further processing. We tried to reflect this more accurately in the text and in Extended Data **S2**.

Automatic color highlighting of tiles mismatch can be done in BigStitcher

Extended Data **Table S2** has been updated to reflect this. MosaicExplorerJ supports alternate red/green (and cyan/yellow) while BigStitcher supports random colors.

BigStitcher additionally supports per-tile intensity correction using Affine 1D models

Extended Data **Table S2** has been updated to reflect this.

Competing Interests: No competing interests were disclosed.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias

- You can publish traditional articles, null/negative results, case reports, data notes and more

- The peer review process is transparent and collaborative

- Your article is indexed in PubMed after passing peer review

- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000Research