Reviewers' Comments:

First of all, we want to sincerely thank all the reviewers for their constructive and helpful criticism. We hope that we were able to address all of your concerns and suggestions with our substantial revision of the manuscript. Apart from considerable changes in the manuscript including simulations of light propagation using raytracing to model aberrations, we extended the BigStitcher software with modules that allow for non-rigid registration and virtual re-blocking to refine alignment, as well as a powerful method for quality estimation based on Fourier Ring Correlation.

*Reviewer #1:*

*Remarks to the Author:*

*Hörl et al. present a new software tool for automatically aligning large-scale 3D image tiles that were acquired using light-sheet microscopy including several refinement steps for optimal alignment. In addition to stitching tiles that were imaged from a single orientation, multiple pre-registered orientations can be fused as well, rendering the software suitable for very large specimens. The software is built upon several successful preexisting tools like the BigDataViewer, Bead-based Registration, ImageStitching and the MultiView Deconvolution algorithm by the same authors and implemented as a Fiji plugin, which will potentially make it useful for a large audience. The plugin is well-documented, includes sample data and should thus be easy to use for new users. The paper is nicely written, clearly structured and easy to follow. The figures are well designed and properly emphasize the possibilities of the framework. I have a few minor suggestions listed below but generally recommend accepting the manuscript.*

We thank the reviewer for this positive evaluation of our work.

*- The paper lacks a bit of discussion about the limitations of the framework. Maybe it would be good to extend the conclusions part a bit in this direction (only if there are any substantial limitations, of course).*

We added a section “Limitations” to the online methods/supplemental material. There is unfortunately not enough space for an in-depth discussion of these aspects in the main text itself. We therefore commented more on existing limitations in the respective paragraphs.

*- The new tool is based on several existing software packages and algorithms. It would thus be good to highlight, which of the parts are actually new algorithmic contributions and which parts were reused more or less unchanged from previous implementations.*

We highlighted the new developments of BigStitcher more in the text and additionally added significant functionality as outlined in the introduction (non-rigid, quality estimation, splitting).

*- Line 345 (main manuscript) and Supplementary Note on link dropping: here you refer to a distance threshold that is used to limit the correspondence assignments and a user-defined threshold for the average error. How does the user have to adjust these parameters? Is there any intuition on how to set them properly?*

We added the following text to the documentation, which hopefully makes it easier to understand and adjust:

The global optimization with iterative link dropping has two error thresholds:

* **relative error threshold:** the optimization will be repeated until the largest error of any tile is smaller than the average error of all tiles times the threshold. Lowering this threshold emphasizes consistent quality of the alignment.
* **absolute error threshold:** the optimization will be repeated until the average error of the tiles falls below this value, which should be set the average magnitude of displacements (in pixels) that is still "acceptable" after alignment.

**Both** stopping conditions have to be met for the optimization to finish.

Setting the thresholds can be a bit tricky, because both too high and too low values might lead to undesired results. Intuitively, one would set the 'absolute error' as low as is still acceptable and the 'relative error' rather low to ensure consistent quality. Setting the thresholds too low, however, will result in links being dropped until only a spanning tree of the link graph remains, in which case there is an "error-less" solution for placing the tiles, but it might be sub-optimal due to the emphasis on the remaining links.

Since the processing time for the global optimization is comparatively short and results are immediately displayed in BigDataViewer, manual optimization of these meta-parameters should be feasible, either starting with high values and lowering them or starting with low values and increasing them until the best and most consistent alignment is produced. We found, however, that the default values of 3.5px absolute error and 2.5 relative error work well in many cases. To give the user a bit more flexibility without confusing them with setting the values directly, we recently added "STRICT" (default thresholds) and "RELAXED" (default thresholds \*2) presets.

The documentation for the global optimization can be found at<https://imagej.net/BigStitcher_Global_optimization#Optimization_strategy_and_convergence_criteria>

*- The same applies to the regularization parameters \lambda\_1 - \lambda\_3 in Supplementary Note 7: how are they set properly? Any intuition or a rule of thumb how to specify them?*

We added a usage guide to the documentation, which can be found at <https://imagej.net/BigStitcher_BrightnessContrastAdjustment>

*- Figure 1: For panels (l, m) magenta is hardly visible. If this is in fact intended to show the perfect overlap, maybe mention that differences are hardly visible?*

We adjusted the images and the figure legend accordingly.

*- Generally, use consistent capitalization of the headlines (e.g., in the online methods there’s a mixture of title and sentence case)*

We fixed this inconsistency.

*- Supplementary Figure 1: Not sure what the benefit of showing the random forest classifications is to this figure. In my opinion, the main point the figure should emphasize is the preservation of fluorescence, i.e., a close-up of a few frames of the raw signal would be sufficient (potentially increase the zoom, such that structures can be easily resolved by eye).*

Our original rationale was that consistent detection by a random forest shows that the signal is preserved sufficiently, but we agree, it is not super intuitive. We therefore now show 6 slices with increasing depth of up to 2,0cm together with quality measurements based on the Fourier Ring Correlation to illustrate how deep the signal that can typically be detected.

*- Supplementary Figure 10: For DS2 and DS4 the distribution apparently becomes more narrow. Any idea why it broadens again for the DS8 plot?*

The reason is that downsampling smoothens the data, which, in combination with the subpixel localization of the spot, increases the registration quality at 2-fold and 4-fold downsampling. However, with even larger downsampling the loss of pixel resolution outweighs the effect of smoothing hence the quality drops. If, for example, you would have perfect data with no noise, the quality would drop already at 2-fold downsampling. We added this explanation to the online methods part.

*- Supplementary Figure 15: Interest points are hardly visible. Maybe show a magnification inset or generally increase the size of the interest point markers?*

We updated the figure and used screenshots made on a “normal” instead of a 4K display. The points are better visible in the real application, even on 4K screens. Now it should be nicely visible in the PDF.

*- Supplementary Figure 17: Scale bar is missing.*

We added the scalebars to the image.

*- It seems the test data is currently hosted on a GoogleDrive. Maybe it would be good to move the data to a different location, where it is also guaranteed to have persistent links that are also available in the future.*

We additionally added the example files to the Open Science Framework as a project: <https://osf.io/bufza/> It is one of the recommended websites by Nature (<https://www.nature.com/sdata/policies/repositories>). We also added the link to the supplement.

*- Supplementary Information Typos: know shift -> known shift, sice -> since, groping -> grouping, resonable -> reasonable, stong -> strong*

Thanks for pointing out those typos, we corrected them.

*- Supplementary Videos 2 and 5 are a bit too fast/short. I would recommend lengthening the videos, decreasing the playback speed a little or adding still frames in between, which would make it more easy to follow without having to restart the video multiple times.*

We updated both videos accordingly.

*Reviewer #2:*

*Remarks to the Author:*

*The article by Hoerl et al. describes an improved methodology, BigStitcher, to reconstruct large, modern fluorescence imaging experiments. The authors claim that BigStitcher is easier and more efficient to use than existing pipelines, automatically selects the best illumination direction, efficiently fuses and deconvolves multiview data, and compensates for all major optical effects in light sheet imaging.*

*BigStitcher is a much-needed advancement in the field, as many laboratories are now generating large imaging data sets without a rigorous pipeline to assemble, visualize, and analyze these data. While BigStitcher does not deal with analysis, the principled approach detailed in this manuscript for data assembly and visualizing will aid analysis and rigor in reporting by standardizing the first two steps in a pipeline.*

*In general, this is a well-written manuscript with clear experiments, results, and justifications. However, I do not believe the authors claim of compensating for all major optical effects is merited.*

*Unfortunately, the authors cannot control how users prepare their cleared tissue samples and carry out light sheet (or other high-throughput) imaging. These choices may render it impossible for BigStitcher to compensate for all major optical effects. My comments and questions are almost entirely concerned with this claim. I have attempted to provide a set of questions with actionable items for each major comment.*

We first of all we thank the reviewer for generally positive remarks on our software. We are very sorry that we created a confusion about what we actually intended to describe with “all major optical effects”. We agree with the reviewer that this is actually not the case, I think it mostly happened because of the space limit in the abstract. What we intend to claim is that we compensate for all major geometric transformations introduced by optical effects and also overall intensity adjustments by multiplication and addition of intensity values. We changed the abstract and text accordingly. To further strengthen this point we added in-depth simulations of light propagation in tissues using raytracing, added non-rigid capabilities, and implemented strategies to at least be able to robustly detect areas with reduced image resolution. We hope that this will address the concerns of the reviewer.

*Major comments and associated question for the authors:*

*1. Recent work by the Tomer group (Migliori et al. BMC Biology 2018) demonstrated why the exciting light sheet in a flat dual-sided light sheet configuration might not fully penetrate thick cleared samples. Knowledge of this optical aberration motivates the author's implementation of algorithms to decide which illumination direction provides the highest signal-to-noise. However, it is possible that one or both exciting light sheets will completely diverge while traversing a thick cleared tissue sample before reaching the imaging area.This situation will lead to a lack of usable information at that particular imaging plane, making it impossible to computationally correct for the responsible aberration.*

We agree and we see these types of aberrations in samples (Fig. 2a). We added simulations of light propagation in tissue using raytracing that recapitulate this behavior and illustrate that BigStitcher correct for it using affine, split-affine or non-rigid -- as long as the lightsheets stay within the depth-of-field of the common detection objective (Supp. Fig. 1). We now make it very clear that if that is not the case, the data cannot be reconstructed using BigStitcher: “… however, once the lightsheet is out-of-focus, blurred image data is acquired that cannot be reconstructed using BigStitcher. Such first-order defocusing can, however, be minimized by employing autofocusing during the acquisition process[ref]”

*If one light-sheet provides minimal information while the other complete diverges deep within a sample, there will be a high mean intensity signal from the diverged light sheet. The illumination detection schemes implemented in BigStitcher would incorrectly assign this high mean intensity signal as the 'correct' channel.*

*This false-positive identification is well known in autofocusing algorithms. An excellent discussion of the various options for autofocusing in light sheet fluorescence microscopy is found in Royer et al. Nature Biotechnology 2016. An application of a Fourier-based algorithm for autofocusing in cleared tissue imaging is found in Ryan et al. Nature Communications 2017.*

*While these manuscripts are mainly concerned with correcting first order defocus aberration during imaging, the computational principles of identifying the highest signal-to-noise carry over to selecting the best illumination direction.*

*Have the authors considered frequency-domain based algorithms to determine the best illumination pathway? How do these compare to the summary statistic based algorithms currently implemented in BigStitcher?*

Thanks a lot for this detailed input. We added additional methods (mean gradient & relative Fourier Ring Correlation (FRC)) to determine the correct illumination direction and quantified their performance (Suppl. Fig. 2). Plain FRC did not perform well on lightsheet data due to sCMOS camera patterns, which we overcame with a new implementation of relative FRC. Indeed, both perform better than simple intensity measurements. We added some further description of the methods to the user guide: <https://imagej.net/BigStitcher_Select_illumination>

In general, however, our experience is that all three methods arrive at similar illumination selection results, with gradient magnitude and FRC providing more differentiation power in close cases (at the cost of longer compute times).

Motivated by another one of your comments, we additionally extended our relative FRC to enable image quality estimation throughout the sample, which turns out to be an extremely valuable tool for quantification of image quality throughout large datasets (Supp. Fig 7,8 & Supp. Video 8,9).

*2. First order defocusin the imaging pathway is another common aberration in light sheet imaging. This aberration will lead to defocused images with minimal signal-to-noise. While the outcome is similar to the aberration discussed in Point #1, the cause of the aberration is different due to the orthogonal pathways in LSFM.*

*Because of the loss of signal-to-noise, the algorithms contained within BigStitcher cannot compensate for this aberration. The author's do not present a signal-to-noise plot as a function of depth from the surface of the cleared sample in this study. There is minimal information on experimental parameters (step size, confocal parameter, etc.) for the Zeiss Z.1 experiments as compared to the IsoView experiments.*

We agree as outlined above and made that clear in the text. We added a plot that highlights the signal-to-noise in typical cleared datasets as a function of depth and show that the relative FRC accurately measures this quality (Supp. Fig 1). We also added supplementary table 1 with detailed parameters of all datasets used in this publication.

*They also do not make clear if the Zeiss Z.1 or IsoView attempts to correct for first order defocus due to the refractive index mismatch between the imaging objective, refractive index matching media, and optically cleared or expanded tissue. This is particularly relevant for the larger sample imaged in the Zeiss Z.1.*

As already mentioned above, we now make clear that BigStitcher does not attempt to correct for first order defocus, but is limited to geometric transformations and intensity adjustment.

*It is not clear from the Figure 1, Figure 2, supplemental movies (Video 1 and Video 6) what the signal-to-noise deep within the cleared tissue sample is. Figure 2e is low-resolution imaging and box (iii) only shows the multi-view fusion of the data.*

We added that plot (Supp. Fig. 1) together with quality estimation based on relative FRC that illustrate the image quality in a single stack (Supp. Fig. 7 & Supp. Video 8) and an entire mouse brain (Supp. Fig. 8 & Supp. Video 9).

*I believe it is not feasible for the author to provide a link to the raw light sheet data because of the large size. Instead of this, more detailed discussion of the experimental parameters as well as improved supplemental figures and movies would demonstrate that BigStitcher is properly accounting for optical aberrations deep within the sample.*

We added supplementary table 1 that shows more detailed acquisition parameters for all experiments, added several detailed supplementary figures (1,2,3,4,7,8,17) that illustrate and quantify image reconstruction and perform automated image quality measurements, and in the new Fig. 2 we also show a practical example deep inside a mouse brain and how we reconstruct it. At the end, BigStitcher will not correct for any defocusing, but the rFRC plots will alert the user if image significantly quality drops in some parts of the sample.

*Have the authors performed a signal-to-noise analysis as a function of depth? For example, how does the Fourier Ring Correlation change as a function of depth? Can the authors please expand on the experimental details for the Zeiss Z.1 experiments?*

We made a detailed figure showing image quality as measured by (relative) Fourier Ring Correlation as a function of depth, which shows that it is an appropriate way of visualizing image quality within BigStitcher (Suppl. Fig. 1). It is now available as an option in BigStitcher and can on-demand produce plots as shown in (Supp. Fig. 7,8 & Supp. Video 8,9) for any dataset. The experimental details are now listed in Suppl. Table 1.

*3. Higher order aberrations can be dealt with using deconvolution. Because of the depth-dependence of optical aberrations in cleared tissue, this can be a complicated and difficult task. The authors attempt to solve this problem by imaging diffraction limited fluorescent beads embedded within a hydrogel matrix. They do not present quantitative evidence that this approach is a feasible method for deconvolution. The obtained PSF are not provided, deconvolution results from image planes at the surface and at depth for each detection pathway are not provided, and deconvolution results from image planes at the surface and at depth for multiview deconvolution are not provided.*

*If the author's wish to claim that BigStitcher accounts for all major optical effects, theymust*

*present a quantitative evaluation of their deconvolution scheme for thick cleared tissue.*

As discussed already, we do not wish to claim that. What we would like to claim is that we compensate for all major geometric transformations introduced by optical effects and also overall intensity adjustments by linearly (add and mul) modifying values. Regarding deconvolution, we provide our implementation of Lucy-Richardson deconvolution that is able to run on parts of these gigantic datasets.

*4. Matsuda et al. Scientific Reports 2018 demonstrated fiducial free chromatic aberration correction. A key point of their work is that the same features must be imaged in all detection channels using one excitation color for accurate correction if using interest points. On Line 108 of this manuscript the author's state that their approach works if 'autofluorescence levels are sufficiently high.' This seems to be a critical point and should be reinforced to users that interest point extraction followed by ICP cannot correct for all chromatic aberration unless the same structure is present through all channels when excited with the lowest wavelength excitation (e.g., DAPI across all images without other excitations).*

In order to make that clearer we added a special figure (Suppl. Fig. 3) that shows what kind of chromatic aberrations we can correct in BigStitcher. We also make clear that using affine models is an approximation and that more specialized chromatic aberration software can be used before import to BigStitcher. If there is demand for a specific method we would in long run happily consider to integrate it into BigStitcher in order to enable virtual chromatic aberration correction.

*Have the authors quantified how this single-excitation/multiple-detection chromatic correction scheme compares with their current interestpoint across all channels method?*

To elaborate a bit more, we do not intend to provide a fully-fledged chromatic aberration correction with BigStitcher, just a quick solution that works in the cases where aberrations are small and there is a lot of autofluorescent signal visible across channels as shown in Suppl.   
Fig. 3. We are aware that there are better solutions, but thanks a lot for pointing them out to us. We actually cite them now in the Limitations section and consider to incorporate them into BigStitcher if there is demand and if it is possible. Otherwise, they can always be run prior to import into BigStitcher.

*Minor comments:*

*1. We attempted to evaluate BigStitcher on tiled 3D stacks with 10 x 10 XY tiles with 20% overlap, 51 z positions, 4 colors, 2 spectrally separated detection sCMOS cameras (OrcaFlash4.0 v2) each with 2048x2048 pixels. These stacks were acquired using the Multi-Dimensional Acquisition plugin in MicroManager 2.0 gamma build 2018.04.01. BigStitcher was unable to parse the metadata correctly for tile positions outputted from MM 2.0. We had to arrange the tiles using the built-in functions manually.*

*It would be helpful to have a 'how-to' guide on parsing the metadata from the MDA plugin into a BigStitcher compatible format so that the stage positions could be utilized.*

Thanks a lot for this precise feedback. This actually constitutes a perfect GitHub issue (<https://github.com/PreibischLab/BigStitcher/issues>), which is the preferred way to handle such problems as well as feature requests (we added that to the Limitations section). It is a bit hard to address without any example data, but most likely it is a problem in Bioformats rather than BigStitcher, but hard to say. Still, great to see that you found a way around that bug.

We tried to reproduce the error using simulated hardware (DemoStage, DemoXYStage, 2 DemoCameras) in micro manager (same build, macOSX) as follows:

\* We set up an MDA with multiple, hand-picked, XY positions, relative z stacks and 2 channels, each using a different DemoCamera

\* We chose to save the Images as Image stack files: MMStack\_Pos\*.ome.tif

We then imported the data into BigStitcher as follows:

\* Define a new dataset using the Automatic Loader (Bioformats based)

\* As the file path, we used /path/to/images/MMStack\_Pos\*.ome.tif

\* We mapped BioFormats Series to Tiles (BioFormats Series are? -> Tiles)

\* For the numerical pattern detected in the files (MMStack\_Pos{0}.ome.tif) we chose to --ignore this pattern-- (in the newest version, this pattern is ignored automatically)

\* Under Move to Grid, we chose “Do not move Tiles to Grid (use Metadata if available)”

By following this procedure, BigStitcher seems to be able to load xy positions from metadata correctly, at least for the DemoXYStage.

If you tried the same steps on your setup and it did not work, we would of course be happy to look at your data and figure out the problem, if you could provide a small example dataset anonymously somehow or please file an issue as mentioned above.

Other users have reported problems loading MM data as well (<https://github.com/PreibischLab/BigStitcher/issues/37>), so we fixed the problem of having to select the counterintuitive “--ignore this pattern--” and we provide an easy way of flipping images if your stage coordinates do not comply to our x=left, y=down scheme. Hopefully, this solved the problem.

Importantly, we’ve also recorded a screencast showing how to import TIFF datasets (including MM) to make it easier for people if the loading of the original image data fails: <https://youtu.be/aUofNP6V0lg> We added this to the supplement.

*2. On this same dataset (and four replications of this dataset from different preparations), we attempted to use the interest point extraction followed by ICP. Each channel is 2048x2048 pixels with 51 z positions separated by 250 nm. An Olympus 40x NA 1.3 oil immersion objective was utilized for excitation and detection for an effective pixel size of ~162 microns. We intentionally introduced a slight mismatch in XY position between the two spectrally separated detection cameras to test the chromatic aberration algorithm. An imaging flat beamsplitter from Semrock directed emitted photons due to 405/488 excitation to camera #1 and photons due to 555/640 excitation to camera #2. There is some spectral bleed-through of photons due to 488 excitation to camera #2.*

*The Register with ICP options 'Simple (tile registration)' or 'Simple (all together)' options causes the axial extent of the stitched image to expand by almost ten-fold with huge distortions in the channel associated with 405 nm excitation (DAPI nuclear counterstain in this case). However, only performing 'Simple (chromatic aberration)' correctly aligned the offset in the cameras with no axial distortion.*

*This behavior seems abnormal. Perhaps we are utilizing the software incorrectly?*

The ICP algorithm might produce these erroneous results if no reliable point matches can be determined. Point matches are iteratively determined by matching interest points in one image to their nearest neighbor in the other, calculating and applying a least-squares optimal transformation to align the point sets and repeating the process until it converges. In the presence of too few actually correct interest-points present in both images (e.g. due to too many detections in the background) or a too large neighbor search radius, the procedure might converge to a matching of actually unrelated points, and thus a bad registration.

One requirement of ICP is that the images are already roughly aligned. For tile registration, it should therefore be used as a refinement step after an initial alignment with phase correlation. Did you by any chance perform the ICP before stitching via phase correlation (guess because your question about that comes after this)? In this case, the observed behavior would make sense, as the channels, with only a slight shift are sufficiently pre-aligned for ICP to work, while the tiles are not pre-aligned enough and erroneous registrations are generated from the unreliable point matches.

Otherwise, playing with the detection parameters (downsampling and threshold) to detect more sensible interest points (they can be visualized in Multiview mode) or adjusting the neighbor search radius (Fine/Normal/Gross alignment) might help. Furthermore, by going to the Expert... Option for the ICP refinement, one could restrict the transformation model to rigid (no scaling/shearing), apply stronger regularization to the model or use manually detected InterestPoints (detected in Multiview mode) for the refinement. In general it is very helpful to actually look at the detected interest points as well as the correspondences that were found. You can do this in multiview mode, active Interest Point Explorer and click on the number of the respective set of interest points. This gives you exactly the same image as shown in Supplementary Fig. 17,20. This will help you to understand what actually went wrong.

We have in-depth documentation for this step of the processing pipeline on the ImageJ wiki:<https://imagej.net/BigStitcher_ICP_refinement>

It would also be helpful to have the example data to really advise what is actually going wrong in this case. Feel free to open an issue on github or give us access to the data anonymously (<https://github.com/PreibischLab/BigStitcher/issues>). All-in-all this sounds like this is a problem that is fixable by choosing the right set of parameters, which is sometimes a question of experience. We actually plan on doing a second course on BigStitcher use in the future and will make the course available online where we will discuss exactly such issues and how to figure it out best.

*3. It is not clear how the 'best' n points are selected during the Phase Correlation step. Can the authors please clarify the process of the algorithm?*

Ideally, the phase correlation algorithm produces a phase correlation matrix (PCM) with a single intensity peak, whose location corresponds to the shift vector between the two input images. However, in real images with noise and repetitive structures, the PCM might contain multiple peaks. Furthermore, due to the circular nature of the Fourier shift theorem, each peak might correspond to 2^(number of dimensions) actual shifts. In our implementation, we therefore check the n peaks (subpixel-localized local maxima) of highest intensity and validate each possible shift by calculating the cross correlation of the shifted images. The final result is the shift yielding the highest cross correlation.

The procedure is explained under Pairwise Stitching using Fourier-based Phase Correlation in the online methods as well as in the BigStitcher documentation on the ImageJ wiki: https://imagej.net/BigStitcher\_Advanced\_stitching#Advanced\_phase\_correlation

*4. How does the BigStitcher multiview fusion compare to the CUDA-based IsoView multiview fusion presented in Chhetri et al. Nature Methods 2015? A quantitative comparison with the instrument's original reconstruction algorithm would strengthen the author's argument that BigStitcher incorporates multiple views in a streamlined pipeline that corrects for major optical effects.*

Both are implementations of the Lucy-Richardson deconvolution algorithm: “we wrote our own implementation of the Lucy-Richardson three-dimensional multiview deconvolution algorithm4,17,24” (citing the Chhetri paper, who is co-author of this paper as well). Reference 17 describes an older version of the multiview-deconvolution used in this paper. This means the underlying algorithm is identical, as long as no speed optimization introduced in [17] is used. The contribution this paper makes is to extend our own implementation so that it can directly run on parts of a multi-terabyte volume, which the CUDA software cannot.

*5. We find the memory estimates that BigStitcher provides for image fusion are consistently incorrect. Using a workstation with 128 GB of RAM, we regularly run out of memory for stitching processing that the BigStitcher plugin estimates will only require ~90 GB. Providing accurate estimates to users will streamline their use of the plugin and aid adoption. Particularly because the plugin offers an alternate fusion strategy that does not require as much RAM.*

Two things come to mind at first assuming that you refer to the fusion process. Did you resave your dataset to HDF5 in the beginning? Do you assign enough RAM to the Fiji Java instance, i.e. does Java use the full 128GB of the system? Both could maybe explain this behavior that we could not recapitulate on our side. It would be very useful to have the data and your set of commands that recapitulates this behavior. This is also a perfect github issue where we, I am convinced, would relatively quickly be able figure out together how to solve this problem and maybe fix a bug or two in the process. I am sorry that we cannot fix this issue without more information on the problem.

For example, other users contacted us (by email) about a similar OutOfMemoryError during the stitching (phase correlation) step. We soon figured out that it was caused by them using a machine with a very large number of CPU cores, which led to an unreasonably large number of pairwise shifts being calculated in parallel (exhausting available RAM). Inspired by this feedback, we now support manual setting of the number of parallel tasks in the expert parameters of the stitching step in the most recent release of BigStitcher.

*Reviewer #3:*

*Remarks to the Author:*

*This paper describes the BigStitcher software for reconstruction of large (TBs sized) multi-tile, multi-view light microscopy acquisitions of whole-tissue and whole-organism.*

*I enjoyed reading this paper. The paper is well written and the chain of thoughts are easy to follow. The results look very promising and useful to the community.*

First of all, thanks a lot for this positive remark on our work.

*Can the authors please comment to why their tool is significantly slower in execution time than most of the other tools for the Fusion step (as shown in table 1)?*

The difference is that BigStitcher fuses using affine models, not translation models, which is a more complex and compute-intense problem, which also implies that data cannot be loaded just plane-by-plane as it is possible for all other tools. As pointed out already in comments above, we added light simulations (Fig. 2), illustrations (Fig. 2, Suppl. Fig. 3,4,9), and quantifications (Fig. 2m, Suppl. Fig. 17), that show that translation is not sufficient for these kinds of datasets. We therefore made it clearer in the table (now in supplement due to size restrictions) that the fusion times are not really comparable. At the same time, BigStitcher offers the virtual fusion as well as interactive display of the data – often there is simply no need to fuse the data. Future developments directly work on the multiresolution data as displayed in BigDataViewer as fusions of these gigantic datasets cannot be handled in a reasonable way. What might remain very important are 1) downsampled fusions (e.g. for visualization or 3d rendering), where BigStitcher, despite using affine models, outperforms most “classic” stitching tools and 2) fusions of specific areas at full resolution (e.g. where machine learning pointed out a potentially interesting structure), which is very intuitively possible with BigStitcher. We pointed all this out in the Online Methods/Limitations. However, we will continue to work on the fusion to make it more efficient wherever possible, and Supplementary Fig. 17 shows that fusion times can be significantly decreased using SSDs.

*I am surprised that the authors did not address the accuracy of their stitching method. Stitching is the first computational step when conducting a large scale biological experiment and some conclusions depend on the quality of the stitched image.*

*Do the authors intend to quantify the accuracy of their stitching tool in the future? Visual inspection is not enough especially for such large specimen.I would like to point the authors to a recently developed tool:* [*https://www.nature.com/articles/s41598-017-04567-y*](https://www.nature.com/articles/s41598-017-04567-y) *that does microscopy stitching in 2D where a reference dataset was created to quantify the stitching accuracy of the developed tool.*

Thanks a lot for the input and the hint, and we use a somewhat similar approach here now. In addition to the quantification of the pairwise stitching (Supp. Fig. 13-15) we now added an error quantification of different transformation models (translation, affine, split-affine, non-rigid) for an entire dataset that we specifically acquired for this purpose. Although rather small (166GB), it contains multiple tiles (2x3) and is imaged using dual-illumination and with two views (multi-view). To precisely quantify errors, we manually identified points in between all overlapping images and measured their distance relative to each other after alignment (Supp. Fig. 17). We then compared theoretically possible best alignment based on the corresponding points to the actually achieved registration error (Fig. 2m). We thereby show that translation alone is not sufficient for alignment of these kinds of datasets, which is now also pointed out more clearly in Supp. Table 2. Just to mention it, for two reasons we did not include comparisons to the other software solutions. Most importantly, this precise alignment quantification can only be extracted before fusion of the data, which is not possible for most of the solutions. This is because after fusion (especially with blending or similar) it is practically impossible to faithfully re-detect the same points with subpixel accuracy as the original input images -- see for example our Supp. Fig. 4, inset (iii), where parts of some views (green) become practically invisible due to the fusion mode. Therefore, faithful error quantification is simply not possible on fused datasets. Additionally, there is a recent editorial on benchmarking that rightfully points out that benchmarks performed by the authors of a new paper will always have a bias towards their own solution (<https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1006494>). And this is entirely true, specifically for image registration. Even when assuming best of intentions (that we all have of course), every author will know how to tune parameters of their own software better than any other. Therefore, we, as a community, should have open, blind benchmarks where every team submits their software that is then evaluated in an unbiased way. That, however, still needs to be developed to be really fair. Nevertheless, we hope that with our quantification, illustrations (Supp. Fig. 3,4,9), and data simulation (Fig. 2) we can convincingly show that (1) translation is not sufficient, (2) the choice of the required transformation model depends on the amount of aberrations present in the dataset, and (3) that BigStitcher is able to align such complex, large datasets robustly, quickly, and interactively.