SOFTWARE TOOL ARTICLE

CytoBrowser: a browser-based collaborative annotation platform for whole slide images [version 1; peer review: 1 approved with reservations, 1 not approved]

Christopher Rydell , Joakim Lindblad

MIDA Group, Department of Information Technology, Uppsala University, Uppsala, 751 05, Sweden

V1 First published: 22 Mar 2021, 10:226
<https://doi.org/10.12688/f1000research.51916.1>
 Latest published: 22 Mar 2021, 10:226
<https://doi.org/10.12688/f1000research.51916.1>

Abstract

We present CytoBrowser, an open-source (GPLv3) JavaScript and Node.js driven environment for fast and accessible collaborative online visualization, assessment, and annotation of very large microscopy images, including, but not limited to, z-stacks (focus stacks) of cytology or histology whole slide images. CytoBrowser provides a web-based viewer for high-resolution zoomable images and facilitates easy remote collaboration, with options for joint-view visualization and simultaneous collaborative annotation of very large datasets. It delivers a unique combination of functionalities not found in other software solutions, making it a preferred tool for large scale annotation of whole slide image data. The web browser interface is directly accessible on any modern computer or even on a mobile phone, without need for additional software. By sharing a "session", several remote users can interactively explore and jointly annotate whole slide image data, thereby reaching improved data understanding and annotation quality, effortless project scaling and distribution of resources to/from remote locations, efficient creation of "ground truth" annotations for methods' evaluation and training of machine learning-based approaches, a user-friendly learning environment for medical students, to just name a few. Rectangle and polygon region annotations complement point-based annotations, each with a selectable annotation-class as well as free-form text fields. The default setting of CytoBrowser presents an interface for the Bethesda cancer grading system, while other annotation schemes can easily be incorporated. Automatic server side storage of annotations is complemented by JSON-based import/export options facilitating easy interoperability with other tools. CytoBrowser is available here: <https://mida-group.github.io/CytoBrowser/>.

Keywords

virtual microscopy, digital cytology, whole slide image, annotation, assessment, visualization, bioimage informatics, JavaScript

Open Peer Review

Approval Status

	1	2
version 1		
22 Mar 2021	view	view

1. **Simon Nørrelykke** , ETH Zürich, Zürich, Switzerland
2. **Peter Bankhead** , University of Edinburgh, Edinburgh, UK

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

This article is included in the **Bioinformatics** gateway.

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

Corresponding author: Joakim Lindblad (joakim@cb.uu.se)

Author roles: **Rydell C:** Conceptualization, Software, Validation, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing; **Lindblad J:** Conceptualization, Data Curation, Funding Acquisition, Project Administration, Resources, Software, Supervision, Validation, Visualization, Writing – Original Draft Preparation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

Grant information: The project is financially supported by Sweden's Innovation Agency, VINNOVA, through grants 2017-02447 and 2020-03611.

The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Copyright: © 2021 Rydell C and Lindblad J. This is an open access article distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: Rydell C and Lindblad J. **CytoBrowser: a browser-based collaborative annotation platform for whole slide images [version 1; peer review: 1 approved with reservations, 1 not approved]** F1000Research 2021, **10**:226 <https://doi.org/10.12688/f1000research.51916.1>

First published: 22 Mar 2021, **10**:226 <https://doi.org/10.12688/f1000research.51916.1>

Introduction

The acquisition and use of whole-slide images (WSIs) is rapidly growing in research and medical practice^{1,2}. The use of WSIs is often referred to as *virtual microscopy*, due to the appealing ability to digitally recreate the microscopy experience. WSIs are produced by high-resolution scanning of entire microscopy glass slides. The resulting images are often larger than 100,000 × 100,000 pixels and many gigabytes in size; one single image of a cytological or histological specimen may contain more than 100,000 cells. The ability to inspect and annotate WSIs and therefrom derive results has become a crucial component of many bioimage analysis workflows and is also making headway into modern clinical practice. The sheer size of WSI data, where a single image may be larger than the available RAM memory of a standard workstation, is problematic for many standard visualization tools, imposing a need for specialized software solutions.

There is lately a strong desire within the scientific community towards increased use of open and collaborative solutions and the development of open-source software (OSS)³. A strong argument for the open philosophy is that it efficiently reduces the risk for vendor lock-in situations, where a customer using a proprietary product or service cannot easily transition to another technology due to incompatibilities, inefficient processes, or contract constraints.

A variety of OSS project that support WSIs exist, examples include caMicroscope⁴, Digital Slide Archive⁵, Sedeen viewer⁶, TissUMaps⁷, Cytomine⁸, OMERO⁹, and QuPath¹⁰. The vast majority of existing WSI tools are developed with histology in mind, whereas cytology is only recently gaining increased attention.

One specificity of cytopathological analysis, as compared to tissue based histopathological analysis, is the need for several focus levels. Where a tissue sample provides a reasonably flat surface, cells of a cytological sample are distributed in a layer of a certain thickness, from a few micrometers for a liquid-based sample, up to several tens of micrometers for a smear. The depth of field of a high resolution light microscope is often below 1 µm, and microscopic analysis of cytological specimen typically involves adjusting focus individually for each cell. A skilled cytopathologist utilizes the limited depth of focus as a way to separate overlapping cells and also to get a 3D feeling of the sample. To accommodate this property in virtual microscopy, it is common to acquire several focal layers of a specimen and create a so called z-stack. Several studies have examined the importance of z-stacking for whole slide imaging; Hanna *et al.*¹¹ conclude that z-stacks offer a superior mechanism for overcoming focusing problems encountered with digital cytology slides, while Lu *et al.*¹² demonstrated that even such a small deviation from the in-focus plane as 0.8 µm may have a clear negative impact on deep learning-based classification performance. Visual assessment of scanned samples requires easy interactive inspection at multiple resolutions and focus levels, but only very few of the above mentioned OSS solutions offer smooth and fast handling

of z-stacks, where changing focus is as easy as in a light microscope.

A great benefit of virtual microscopy over optical microscopy is the possibilities for sharing and collaboration. Digitized WSIs can be made accessible via the Internet, thereby enabling efficient usage of distributed resources and reaching remote parts of the world. Tools for interactive visualization and collaborative assessment of WSIs facilitate improved understanding as well as increased quality of analysis, where several experts may share experiences. Collaborative tools for WSI annotations are also gaining popularity in medical education; in a mixed methods trial¹³ it was found that collaborative WSI annotation provided superior understanding of key microscopic features by senior medical students as compared to individual WSI annotation.

Despite the growing number of tools for WSI visualization and annotation, we find that there is a lack of light-weight, easy to use OSS tools which are well suited for collaborative annotation of cytological data. The majority of existing tools are directed towards histopathology, with different needs and requirements. Aiming to gather the experience and knowledge of experts from all over the globe, an easy to use, browser-based solution, which may be directly utilized on any computer or tablet without need for software installation, is highly desired. The ability for several users to simultaneously view and comment the same sample improves understanding and analysis quality. The CytoBrowser software, presented here, aims to meet all of these needs and desires. It delivers easy and flexible access to WSIs through a browser-based interface, see [Figure 1](#) for an example view. CytoBrowser aims to well replicate the optical microscope feeling, with good support for z-stacks. A growing set of annotation tools are well suited for cytological data. Efficient collaboration is achieved through shared views, with option to follow the view of another user. Annotations may be added and commented on concurrently by several users. The software, completely in JavaScript and Node.js, is light-weight and very easy to install and run. CytoBrowser is typically hosted as a web service, but may also be used locally. We share the source code of CytoBrowser openly and freely under the GNU General Public License v3.0, with the hope that a growing community of users will jointly contribute to its further improvement.

Methods

CytoBrowser is a fork and extension of TissUMaps⁷ (GPLv3 license), which in turn builds on the OpenSeadragon¹⁴ (BSD-3-Clause license) web-based viewer for high-resolution zoomable images. Although TissUMaps is a very capable tool for visualization of cytology WSIs, it is developed with tissue analysis in primary focus and the user interface and tools are not always ideal for cytological data. CytoBrowser instead puts cytological analysis in the main focus. An important feature for cytological data is efficient handling of z-stacks (currently not supported in TissUMaps); CytoBrowser delivers fast browsing through focus levels by use of transparent layers. CytoBrowser also allows z-level differentiated annotations, useful, e.g., for annotating overlapping cells.

Figure 1. Example of typical usage – annotation of a brightfield WSI in a standard web browser. The user has just placed a marker on a cell nucleus, indicating the Bethesda grade HSIL (High-grade squamous intraepithelial lesion). An overview of the WSI is presented in the lower left part of the screen, with the current zoomed-in viewport indicated as a small red rectangle.

The development of CytoBrowser is aimed towards ease of use by cytotechnologists and cytopathologists on a global scale. Implemented tools for shared multi-user sessions facilitate efficient remote collaboration, including options for joint-view visualization and collaborative annotation and assessment of very large datasets. CytoBrowser delivers this through a web-browser interface which is functional on any modern computer or tablet, facilitating efficient utilization of distributed resources. A demonstration setup of CytoBrowser running on several devices (Microsoft Edge browser under Windows, Firefox browser under Linux, Chrome browser on Android tablet and phone) is shown in Figure 2. The multi-user design of CytoBrowser offers concurrent access to samples, where different experts may assess, comment on, or edit each others annotations, or where an educator may, e.g., show students different medically relevant examples.

Implementation

The frontend of CytoBrowser is implemented in JavaScript. The implementation makes use of the Revealing Module¹⁵ design pattern to clearly separate different parts of functionality and to make it both easily maintainable and expandable. OpenSeadragon¹⁴ is used to display a viewport in the user interface and to dynamically load different parts of the image from the server as needed. The annotations that the user can place on

an image belong to one of several annotation classes. By default these classes follow the Bethesda grading system¹⁶. Classes are defined in a separate file and can easily be modified to suit the specific task.

CytoBrowser utilizes the Deep Zoom Image (DZI) format for fast rendering of very large WSIs. Currently, each z-layer is provided as a separate DZI file, where a filename of the form `imagename_zNNN.dzi` indicates an image with name 'imagename' at z-level 'NNN', and where 'NNN' is any number, typically an offset in nanometer.

The backend of CytoBrowser is run in Node.js¹⁷, with a web server implemented with the Express¹⁸ module. The server provides HTTP endpoints for getting information about the available images as well as for writing and reading annotation data in server-side storage. While these endpoints are mainly intended to be used by the CytoBrowser client, they could also be used to easily access this information from other applications. The server also provides a WebSocket endpoint for full-duplex communication between the client and server to facilitate the collaboration functionality. This communication is done using JSON messages. The only mandatory field in a such a message is the *type* field, which informs the recipient how to process any other fields that may be present.

Figure 2. Demonstration of several users collaboratively annotating a cytology WSI using different devices. The tablet and the laptop in the foreground follow the view displayed on the larger monitor in the background, while the phone is showing a somewhat larger view. The user of the laptop in the foreground is adding a text comment to the upper left marker, while the (followed) user has just placed a HSIL marker on the nucleus in the lower part of the screen. CytoBrowser is compatible with all modern browsers.

When a user opens an image in CytoBrowser, they either select an existing set of annotations to work on or start a new set. Different sets of annotations for the same image may, e.g., originate from different experts or be imported from a machine learning-based classification. The annotation set also defines a collaboration session, consisting of all users currently working on the same annotation set, much like users collaborate on the same document in web-based applications such as Google Docs¹⁹. By encoding the current state of the interface, including the active image and annotation set, but also viewport location and zoom level, in the dynamically updated URL, it becomes very easy to invite colleagues to discuss a sample by simply sharing the active URL. In the same manner, it is easy to return to a previous image location through a stored browser bookmark.

The protocol implemented for collaborations has been created with robustness in mind. The backend stores a canonical copy of the shared data, including information about the participating users, the active image and the placed annotations, that the clients request when entering a collaboration. The backend also verifies that clients are displaying the correct

image. When data is altered by a connected client, only the alteration is broadcast to other collaborators.

For additional details regarding the implementation, we refer the reader to the [implementation.md](#) file of the software.

Operation

Opening images stored on the server is easy; available images are presented to the user as a gallery of thumbnails, see [Figure 3](#) for an example. By default these thumbnails display a lower-resolution version of the full image, but by hovering over a specific entry the user is presented a zoomed-in view from the middle part of the image. After selecting an image, the user is prompted to either start a new annotation session or to continue on a previously created.

CytoBrowser enables placing several different kinds of annotations on an image. Markers can be placed by clicking a single point, rectangular regions can be placed by clicking two opposite corners, and polygonal regions can be placed by clicking each vertex, see [Figure 4](#) for an example of the latter. While these three annotation tools are currently present in

Figure 3. Example of a user selecting which WSI to open, the active image (3rd row, 2nd column) is shown zoomed-in.

Figure 4. Example of a user in the process of outlining a polygonal region in a fluorescence microscopy z-stack.

CytoBrowser, the system is designed to allow easy addition of new tools.

Once annotations are placed, there are different ways of interacting with them. They can be moved or deleted using the mouse. By right-clicking an annotation, the user gets access to more actions in a context menu. Here they can change the class assigned to the annotation or leave text comments on it to provide information that is not captured by the class label alone. In [Figure 2](#), the user of the laptop in the foreground is adding such a comment.

Three different options for saving and loading annotation data are provided. The easiest option is for the user to rely on the autosave functionality of CytoBrowser, where the current annotation set is continuously stored on the hosting server. The second option is to manually store the data on the server. For this option, we also provide a simple versioning system. If a file is saved to a directory that contains a file with the same name, the user gets the option to either rewrite the latest version of the file or to create a new version of it. The user is then able to choose which version of a file to load when browsing the server files. The third option is to store the annotation set on the local machine of the user, using the “import” and “export” buttons in the interface. For all three options, the annotations are stored as JSON files, making the data easily transferable between different tools. This enables the annotations to be used for data selection or training of machine learning methods. Inversely, CytoBrowser may be used to visualize the classification or segmentation results generated by other tools.

We believe that easy collaboration between experts from different fields is important for an efficient workflow. For CytoBrowser, we have implemented functionality that enables such collaboration to be performed remotely, where two or more users can browse and annotate the same image together in a shared session. Any changes made by one user will be seen by the other users in real-time. Users can see each other’s mouse cursors in the viewport, and they have the option to lock their view to follow that of another user. This may be useful, e.g., in cases where an expert from one domain needs to visually explain something about the data to an expert in another domain. This capability may thus enhance the transfer of knowledge between different fields and simplify multidisciplinary research.

Use cases

To download and install the software, follow the instructions at <https://mida-group.github.io/CytoBrowser/>. It is very easy to set up a local CytoBrowser server to be accessed from the same machine (localhost), or, e.g., through an SSH tunnel. For larger and more persistent use cases, we recommend to rely on one of popular open-source web servers, such as Apache and Nginx, as a secure loadbalancing frontend for the server.

Example Bash scripts, demonstrating how to download and convert publicly available images in Zeiss (.zvi) and Hamamatsu (.ndpi) format into the Deep Zoom Image (.dzi) format, suitable for CytoBrowser, are provided together with the software in the [examples](#) directory. Software packages which may be useful for converting WSIs include [libvips](#), [Bio-Formats](#), and [NDPITools](#).

Conclusions

We present CytoBrowser, an OSS developed in response to the request for light-weight, open, flexible, and easy to use tools for collaborative visualization, annotation and assessment of whole slide image data. CytoBrowser is developed particularly with cytological analysis on mind, where efficient handling of z-stacks is a central requirement. The combination of a JavaScript based browser component and a Node.js driven server-side component enables fast access to large image data together with easy and flexible storage and sharing of annotations. Being a tool particularly designed for collaborative interactive multi-user access, it facilitates knowledge exchange and alleviates interdisciplinary research.

In addition to being a much desired resource for the creation of large annotated datasets, crucial for training and evaluation of machine and deep learning-based approaches to bioimage analysis, as well as the visualization and qualitative evaluation of the results of such methods, we also envision that CytoBrowser may be a useful tool for distributed education, quality improvement, competency assessment, proficiency testing, and performance evaluation²⁰.

By delivering a powerful virtual microscope environment which is easily accessed through a web-browser interface and functional on any modern computer or tablet, we hope that CytoBrowser, facilitating efficient utilization of distributed resources and expert knowledge, may contribute towards fair and universally accessible health-care.

Software availability

- Software available from: <https://mida-group.github.io/CytoBrowser/>
- Source code available from: <https://github.com/MIDA-group/CytoBrowser>
- Archived source code at time of publication: <https://doi.org/10.5281/zenodo.4582760>²¹
- License: GNU General Public License v3.0

Acknowledgements

We thank Leslie Solorzano and Prof. Carolina Wählby for their support in forking TissUMaps.

This publication was supported by COST Action NEUBIAS (CA15124), funded by COST (European Cooperation in Science and Technology).

References

- Aeffner F, Zarella MD, Buchbinder N, *et al.*: **Introduction to digital image analysis in whole-slide imaging: a white paper from the digital pathology association.** *J Pathol Inform.* 2019; **10**: 9.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Kumar N, Gupta R, Gupta S: **Whole slide imaging (WSI) in pathology: current perspectives and future directions.** *J Digit Imaging.* 2020; **33**(4): 1034–1040.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Marée R: **Open practices and resources for collaborative digital pathology.** *Front Med (Lausanne).* 2019; **6**: 255.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Saltz J, Sharma A, Iyer G, *et al.*: **A containerized software system for generation, management, and exploration of features from whole slide tissue images.** *Cancer Res.* 2017; **77**(21): e79–e82.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Gutman DA, Khalilia M, Lee S, *et al.*: **The digital slide archive: A software platform for management, integration, and analysis of histology for cancer research.** *Cancer Res.* 2017; **77**(21): e75–e78.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Martel AL, Hosseinzadeh D, Senaras C, *et al.*: **An image analysis resource for cancer research: Piip—pathology image informatics platform for visualization, analysis, and management.** *Cancer Res.* 2017; **77**(21): e83–e86.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Solorzano L, Partel G, Wählby C: **TissUUmaps: Interactive visualization of large-scale spatial gene expression and tissue morphology data.** *Bioinformatics.* 2020; **36**(15): 4363–4365.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Marée R, Rollus L, Stévens B, *et al.*: **Collaborative analysis of multi-gigapixel imaging data using cytomine.** *Bioinformatics.* 2016; **32**(9): 1395–1401.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Allan C, Burel JM, Moore J, *et al.*: **OMERO: flexible, model-driven data management for experimental biology.** *Nat Methods.* 2012; **9**(3): 245–253.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Bankhead P, Loughrey MB, Fernández JA, *et al.*: **QuPath: open source software for digital pathology image analysis.** *Sci Rep.* 2017; **7**(1): 16878.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Hanna MG, Monaco SE, Cuda J, *et al.*: **Comparison of glass slides and various digital-slide modalities for cytopathology screening and interpretation.** *Cancer Cytopathol.* 2017; **125**(9): 701–709.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Lu J, Sladoje N, Stark CR, *et al.*: **A deep learning based pipeline for efficient oral cancer screening on whole slide images.** In *International Conference on Image Analysis and Recognition.* 2020; 249–261. Springer.
[Publisher Full Text](#)
- Sahota M, Leung B, Dowdell S, *et al.*: **Learning pathology using collaborative vs. individual annotation of whole slide images: a mixed methods trial.** *BMC Med Educ.* 2016; **16**(1): 311.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- OpenSeadragon: **An open-source, web-based viewer for high-resolution zoomable images, implemented in pure JavaScript, for desktop and mobile.** Accessed on: Feb. 2021.
[Reference Source](#)
- The revealing module pattern in javascript.** Accessed on: Feb. 2021.
[Reference Source](#)
- Nayar R, Wilbur DC: **The Bethesda system for reporting cervical cytology: definitions, criteria, and explanatory notes.** Springer, 2015.
[Reference Source](#)
- Node.js: **A javascript runtime built on Chrome's V8 JavaScript engine.** Accessed on: Feb. 2021.
[Reference Source](#)
- Express: **Fast, unopinionated, minimalist web framework for Node.js.** Accessed on: Feb. 2021.
[Reference Source](#)
- Announcing Google Workspace, everything you need to get it done, in one location.** Google Cloud Blog. Accessed on: Feb. 2021.
[Reference Source](#)
- Quinn AM, Minhajuddin AT, Hynan LS, *et al.*: **Agreement between cytotechnologists and cytopathologists as a new measure of cytopathologist performance in gynecologic cytology.** *Cancer Cytopathol.* 2017; **125**(7): 576–580.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Rydell C, Lindblad J: **MIDA-group/CytoBrowser: Bugfix release v1.0.1, URL location encoding (Version v1.0.1).** Zenodo.
<http://www.doi.org/10.5281/zenodo.4582760>

Open Peer Review

Current Peer Review Status:

Version 1

Reviewer Report 30 April 2021

<https://doi.org/10.5256/f1000research.55127.r81855>

© 2021 Bankhead P. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Peter Bankhead

Edinburgh Pathology, University of Edinburgh, Edinburgh, UK

This paper introduces CytoBrowser, an open-source solution for viewing and annotating whole slide images and other microscopy data. CytoBrowser is a fork of TissUMaps, adding new functionality to support z-stacks and collaborative annotation.

CytoBrowser looks like it could be very useful for a limited number of specific applications already, and potentially many more with additional development. I believe the paper would be strengthened considerably by describing in more detail the applications where CytoBrowser has been used.

Further documentation and information about future development plans would also be essential if the software is to be widely adopted. To make use of CytoBrowser's collaborative features (which seem a major strength), it would be necessary to set up a server - requiring some level of computing knowledge, effort and resources. For this reason, I think it is important to make the benefits of using the software as clear as possible, and the process for using it as straightforward as possible; without this, I am not sure why a potential user would make the effort. A demo server, video demos, more illustrated documentation, and more detail within this article would all help with this.

Specific points and questions

'A variety of OSS project that support WSIs exist, examples include caMicroscope4, Digital Slide Archive5, Sedeen viewer6, TissUMaps7, Cytomine8, OMERO9, and QuPath10. The vast majority of existing WSI tools are developed with histology in mind, whereas cytology is only recently gaining increased attention.'

- I think SlideRunner (<https://github.com/DeepPathology/SlideRunner>) should be included. Since there is no real direct comparison, it is not clear what the benefits of CytoBrowser are over the others for any applications. A table comparing relevant features could address this.

'The majority of existing tools are directed towards histopathology, with different needs and

requirements.'

- Other than z-stack support (which at least some of the listed software provides), I'm not sure what these different needs and requirements are - at least in terms of the annotation tools (points, rectangles, polygons) provided by CytoBrowser.

'We present CytoBrowser, an OSS developed in response to the request for light-weight, open, flexible, and easy to use tools for collaborative visualization, annotation and assessment of whole slide image data.'

- I suggest swapping 'request' for 'need', or else elaborating upon who requested it and for what purpose.

'CytoBrowser provides a web-based viewer for high-resolution zoomable images and facilitates easy remote collaboration, with options for joint-view visualization [...].'

- It's not obvious to me what 'joint-view visualization' means (I assume multiple users seeing the same region?).

'The multi-user design of CytoBrowser offers concurrent access to samples, where different experts may assess, comment on, or edit each others annotations, or where an educator may, e.g., show students different medically relevant examples.'

- Has CytoBrowser been used in education, or would it require further development? Can access and annotation be controlled (i.e. to restrict changes students can make)? In a sense, any platform that allows image viewing might be used in education. If there are specific strengths in using CytoBrowser for this, these should be specified.

'The protocol implemented for collaborations has been created with robustness in mind. The backend stores a canonical copy of the shared data, including information about the participating users, the active image and the placed annotations, that the clients request when entering a collaboration. The backend also verifies that clients are displaying the correct image. When data is altered by a connected client, only the alteration is broadcast to other collaborators.'

- I believe the robustness claim needs supporting evidence, particularly given the focus on being lightweight and using json files for storage. Has CytoBrowser been stress-tested, with a large number of concurrent users? Is there an expected limit to the number of concurrent users?

'Inversely, CytoBrowser may be used to visualize the classification or segmentation results generated by other tools.'

- How? Are any open standards supported for importing annotations, or must they be converted to a CytoBrowser-specific format? How scalable is this, e.g. does it support thousands or even millions of 'objects'/regions? Are rasterized segmentation results supported (i.e. overlapping another image, rather than annotations)?

'CytoBrowser delivers fast browsing through focus levels by use of transparent layers.'

- What are 'transparent layers' here?

'While these three annotation tools are currently present in CytoBrowser, the system is designed to allow easy addition of new tools.'

- How? Is this documented?

'CytoBrowser is a fork and extension of TissUUmaps7 (GPLv3 license)'

- Is there a justification for forking TissUMaps, rather than contributing to the original project? How will both be maintained in the future? Is there any intention to incorporate future improvements to one software within the other?

'We share the source code of CytoBrowser openly and freely under the GNU General Public License v3.0, with the hope that a growing community of users will jointly contribute to its further improvement.'

- Given that fulfilling the potential of CytoBrowser (as outlined in the paper) would seem to require further development, I would like to understand if the authors intend to further develop the software themselves. Community development would require central maintenance; without clear evidence of active ongoing support it is difficult to see how this hope can be fulfilled.

Use of CytoBrowser

I was able to successfully run CytoBrowser locally on a Mac. There were a two minor complications, and one possibly bigger limitation:

- To run the example scripts, I needed to install wget and libvips (via homebrew), as well as finding, downloading & unzipping bftools. Further documentation, with links, would help others do this.
- No images are available when using Safari locally, but without explanation - it simply appears to be broken. With Chrome and Firefox it works, but some form of notification when Safari fails would be helpful. This is not entirely compatible with the statement *'CytoBrowser is compatible with all modern browsers.'*
- The proposed ZVI example file is extremely dark. Since .dzi involves conversion to JPEG, colours and contrast seem to be 'fixed' at the conversion step - without any option to adjust these through the UI. This seems like a significant limitation for non-RGB images. Is there a solution?
- The need to generate .dzi datasets is a barrier to testing the software (which I imagine is harder on Windows). Providing a relevant dataset (ideally for cytology) without a need for conversion would help overcome this.

Some further comments on using the software:

- I don't recognize the format of the JSON export; is it custom for CytoBrowser? For interoperability, I think a standard such as GeoJSON would be strongly preferable.
- The 'Usage instructions' within the UI are very helpful, although in some cases it would be helpful to have instructions closer to the relevant command, e.g. via tooltips.
- Shortcuts are unexpected (e.g. c, v, b to change annotation tool). Shift + mouse wheel is nice to rotate, but I expected the rotation direction to change with the scroll direction.
- I find the Session and Collaboration interfaces confusing. I found myself inadvertently creating many new sessions when exploring whether I can make changes to the current one (e.g. to set its name). I think 'Session name' should be above 'Username' under the 'Collaboration' window, but it's not clear to me what is the impact of changing either, nor whether 'Change session' means 'change the properties of the session' or 'switch to a different session'.

- The use of separate .dzi datasets for z-slices seems a bit awkward; it would help if these could at least be arranged within the same subdirectory (rather than all inside one 'data' directory) for better data organization.
- Clicking on 'Session' at the top of the screen allows me to change the user, but clicking on 'User' does not.
- The collaborative features of CytoBrowser sound like a particular strength of the platform, however I was unable to test these.

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

Partly

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?

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?

No

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: image analysis, digital pathology, open-source software

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.

Reviewer Report 08 April 2021

<https://doi.org/10.5256/f1000research.55127.r81858>

© 2021 Nørrelykke S. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Simon Nørrelykke

Image and Data Analysis group, ScopeM, ETH Zürich, Zürich, Switzerland

The authors introduce CytoBrowser, an open-source, server-client, browser-interface solution for collaboratively viewing and annotation of large microscopy image data.

CytoBrowser is a fork of TissUUmaps, adding handling of z-stacks incl. z-level dependent annotations.

The intended audience are cytotechnologist and cytopathologists, as opposed to histopathologists.

CytoBrowser also aims to be used in a (remote) teaching setting, through the use of synchronisation between the teacher's/expert's browser-view and the students.

Overall the paper is well written with few if any typographical or grammatical errors and lays out its goal and ambitions in a rather clear fashion.

However, the overall feeling is that CytoBrowser is not ready to be shared with the world as it is missing proper installation instructions, a user-manual, and a demo-site. It feels like a local lab-tool that the authors are taking steps to make more broadly accessible, which is commendable, but there is still some way to go.

OBSERVATIONS ON WRITING STYLE:

The Introduction is comparatively long, reads somewhat like a sales pamphlet, and includes most of the information repeated in the very brief Methods.

There is a tendency of mixing up turns of phrase, e.g. "making headway into", "There is lately", "in the main focus", and "on mind"; which are probably better put as "making its way into" or "making headway in" etc.

Variations of "easy" (easy, easily, easiest) appear 24 times - in most instances the word could be omitted for a more sober delivery, losing no information.

RECOMMENDATIONS:

Make a web-server demo-version available so the potential user can try it out without having to install the full solution.

Compare to commercial solutions - open source is not the only player out there.
How does it stack up against e.g. Arivis (<https://imaging.arivis.com/>)?

Improve the installation instructions by having someone try to follow them: I gave it 30 minutes and failed to get it to run (details below).

Make it run on MacOS as well: Many, if not most, biologists use Macs, not Windows, and definitely not Linux.

Be explicit about which other softwares CytoBrowser can directly import/export annotations

from/to. Does it play well with e.g. ilastik or Imagej (see here for some ref: <https://f1000research.com/articles/9-1248> incl. Jan Eglinger's referee report)

Be explicit about what image formats are natively supported - it seems only .dzi and the user have to make the conversion using some of your provided scripts or find another way. This is fine, but be very upfront about this

Write a proper user-manual either in GitHub or in <https://readthedocs.org/> - this is the accepted standard for useful open-source software looking for users. It is work and it is boring, but it is necessary if you want to have users outside your own lab.

LITTLE THINGS:

The figures online and in PowerPoint are not vector graphics and do not allow zooming to a useful level. PDF file is fine.

INSTALLATION STEPS TAKEN:

Installation on Ubuntu 20.04.2 LTS running on a Thinkpad.

Notice that the Linux commands are different from and additional to the ones you give in your instructions, in order to get just some of the way.

I am not saying it cannot be done, I am saying that it would likely take more time than is usual for shared open-source code and definitely more than a test-user can be expected to invest.

```
#+begin_src
git clone https://github.com/MIDA-group/CytoBrowser.git
cd CytoBrowser/

sudo apt install npm

sudo apt install curl
sudo apt-get install -y nodejs
curl -fsSL https://deb.nodesource.com/setup_14.x | sudo -E bash -

npm install express --save
npm install

mkdir data
mkdir storage
mkdir collap_storage

#+end_src
```

Downloaded the Zeiss example images and moved them to ./data

Running (failing to run):

```
#+begin_src
simon@simon-ThinkPad-P1-Gen-2:~/Desktop/CB/CytoBrowser$ node cytobrowser.js localhost
```

8080

CytoBrowser server listening at http://127.0.0.1:8080

/home/simon/Desktop/CB/CytoBrowser/server/availableImages.js:142

```
    let names = dir.map(name => name.match(nameEx)).flat();
```

^

TypeError: dir.map(...).flat is not a function

at fs.readdir (/home/simon/Desktop/CB/CytoBrowser/server/availableImages.js:142:57)

at FSReqWrap.args [as oncomplete] (fs.js:140:20)

#+end_src

CONCLUSION:

Currently I cannot approve for indexing as I could not verify that the software actually works or does any of the wonderful things described in the manuscript: The test of reproducibility failed! If the relevant recommendations re. documentation, installation instructions, and a test-server are followed, I would be happy to recommend indexing (after verifying that the software works) - any stylistic and linguistic idiosyncrasies are entirely the choice of the authors and will not influence my decision.

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

Yes

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?

No

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

Partly

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. Biophysics.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to state that I do not consider it to be of an acceptable scientific standard, for reasons outlined above.

Author Response 11 Apr 2021

Joakim Lindblad**Response to Reviewer Report #1 (S. Nørrelykke)**

Reviewer: *"Overall the paper is well written with few if any typographical or grammatical errors and lays out its goal and ambitions in a rather clear fashion."*

Response: We are glad to read that the reviewer appreciates our aim towards clarity of presentation.

Reviewer: *"However, the overall feeling is that CytoBrowser is not ready to be shared with the world as it is missing proper installation instructions, a user-manual, and a demo-site. It feels like a local lab-tool that the authors are taking steps to make more broadly accessible, which is commendable, but there is still some way to go."*

Response: We agree with the reviewer that a user-manual and a demo-site are certainly desirable additions that will further improve accessibility to the tool. However, we strongly disagree with the reviewer that an absence of such renders it inappropriate to be shared. Yes, it is a "local lab-tool", which we have developed based on our own needs, and yes, we feel that it may be useful for a wider community and therefore we wish to inform a broader audience about its existence and invite them to try, use, and possibly contribute to its further development in the spirit of FOSS.

The alternative, to not publish the tool before we have found resources to host a demo-site and to write a separate user-manual (to complement the existing help pages of the software), seems as a rather unneeded delay. Further, we consider the installation instructions, which assume an ability to install required auxiliary software, to be sufficiently detailed for the envisioned users of the server side of the tool (see also later part of this response). We emphasize that once set up, users of the browser based interface (including cytotechnologist and cytopathologists) need not perform any installation at all.

Regarding **Observations on writing style:** We appreciate the reviewers opinions and will take them into consideration when updating the manuscript; we will await other reviewer's responses to spare them from reviewing a moving target.

Regarding **Recommendations:** We hope to find resources to host a demo web-server. We do not consider that to be a requirement for publication though. On a *comparison with commercial solutions*, we do not consider a closed proprietary solution a reasonable alternative, as clearly explained in the paper. *"Make it run on MacOS"* – All parts of the software runs well on MacOS. (We wonder how the reviewer reached the (incorrect) conclusion that it does not.) *"Many, if not most, biologists use Macs"* – Additionally, the end user may run the interface on any browser, in any OS (we believe this is made clear in the manuscript). *"Be explicit about what image formats are natively supported"* – This is clearly stated at several places in the text, including very explicitly in the "Use cases". *"The figures online and in PowerPoint are not vector graphics"* – This is due to F1000Research, we will ask them to improve this; we recommend the PDF version where figures are of higher resolution.

INSTALLATION STEPS TAKEN:

Prompted by the reviewer's attempt, we performed an installation on a clean Ubuntu

20.04.2 LTS. This confirmed that the steps listed in the installation instructions are accurate and work as intended. The instructions explicitly point to external sources on how to install an up-to-date Node.js (version 12 or later required).

For the reviewer's specific case of Ubuntu (which ships with v10 of Node.js, despite v12 being out since April 2019) we recommend to verify that the Node.js update actually succeeded, by typing 'node -v' in a terminal and confirming the correct version. The error performed by the reviewer is that the following two operations were executed in the wrong order (see also the output of the 'setup_14.x'-step):

1. curl -fsSL https://deb.nodesource.com/setup_14.x | sudo -E bash -
2. sudo apt-get install -y nodejs

The below short list of steps perform the complete installation on a clean Ubuntu 20.04.2 LTS (these steps also match the installation instructions):

```
#+begin_src
```

```
#curl needed for the Node.js update, libvips needed for conversion of data to .dzi
sudo apt -y update && sudo apt -y install git npm curl libvips-tools
```

```
#install a newer node.js for Ubuntu (v12 works as well)
curl -fsSL https://deb.nodesource.com/setup_14.x | sudo -E bash -
sudo apt-get install -y nodejs
```

```
#the CytoBrowser installation starts here
git clone https://github.com/MIDA-group/CytoBrowser.git
cd CytoBrowser
npm install
```

```
#start the CytoBrowser server
node cytobrowser.js localhost 8080
```

```
#In a separate terminal we download and convert some data using one of the provided
example scripts:
```

```
cd CytoBrowser
./examples/Hamamatsu-OS-2.sh #downloads, converts, and opens a browser window
```

```
#+end_src
```

If there are any problems, we recommend the Reviewer to post an 'Issue' on GitHub, which is the standard and recommended way to handle such events.

Competing Interests: No competing interests were disclosed.

Author Response 13 Apr 2021

Joakim Lindblad

LITTLE THINGS:

Reviewer: "The figures online and in PowerPoint are not vector graphics and do not allow zooming to a useful level."

Thanks for pointing this out. We have contacted F1000Research about the issue and they have now (this morning) replaced the web images (and pptx) with higher resolution files.

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