Peer Review

Review of: "CSMA: An ImageJ Plugin for the Analysis of Wound Healing Assays"

Laurent S.V. Thomas[1]

1. MODIS, European Molecular Biology Laboratory, Heidelberg, Germany

This article describes a new ImageJ plugin for the quantification of the regeneration rate for the common “cell scratch assay”. The authors compared this plugin with 2 previously available software tools and reported a better accuracy when quantifying the wound area, by considering individual cells that might have migrated ahead of the main regenerating edge.

The manuscript reads well overall, although some more precision and/or reformulation would be needed (detailed below).

The proposed approach using a specific edge filter (canny) is original, besides the authors have put some effort in describing the implementation in details and also providing documentation in the code repository of the project. This is highly valuable both for end-users to handle the tool and for developers of image-analysis workflows, who could find here some inspiration for similar use cases.

Software architecture

I agree with the first reviewer that combining ImageJ for its UI capabilities, with python for the processing is a bit unusual and makes the installation process a bit cumbersome, although a “turn key” installer is provided for windows. In my opinion, this could prevent larger adoption of the tool.

I could not actually install the plugin on my pc following the current installation instructions of the GitHub repository, I did not use the installer, I managed to create the anaconda environment, but could not find the “target” directory (step 4 of the installation instructions) where the jar file for the plugin is supposed to be. Is it supposed to be provided by the installer too? It’s a windows-only installer though, which would exclude users of other OS.

I guess the authors were thinking to put the jar file in the GitHub repository together with the source code. I would rather recommend archiving the compiled jar as a “release” in the repository (Releases ·

AminaSagymbayeva/CSMA_WoundHealing). When drafting the release, it is possible to attach the .jar file as a binary file so it gets archived with the corresponding source code (the latter being automatically archived by github when making a release). This would also provide a solution to releasing new versions of the plugin.

While it would be possible to rewrite the python code relying on OpenCV to a java equivalent, to make it an integral part of the ImageJ plugin (see IJ-OpenCV for how to use OpenCV functions in Fiji), it would be a major effort.

However, given that the core functionality is written in python, it would be valuable to demonstrate how python programmers could take advantage of the functions available in this core unit, for instance in an example jupyter notebook hosted in the repository. Especially since python is increasingly used for image-analysis, and that software like napari also provide flexible UI capacities for such applications.

Benchmarking of the tool

The authors extensively benchmarked their plugin against existing tools, with both previously available datasets and with own data for different controlled conditions. They also compared the performance of their plugin with default settings and user optimized ones. While the authors report better accuracy of their tool compared to existing ones, they also propose hypotheses for why former tools underperform in some cases and discuss potential limitations of the proposed CSMA plugin. The image-data is also well completed by quantification of the closure rate, demonstrated by fitting model equations. Overall the scientific approach is thus appropriate and convincing.

I also agree with the previous reviewer, that in figures comparing different software, the representation of the wound in images should ideally be the same to ensure the most objective comparison. Example in figure 1, the wound is highlighted with edge for HTM while it’s a “filled” ROI in CSMA.

The low contrast and size of the microscopy image in the pdf is such that it’s difficult to distinguish whether an area is part of the gap or covered with cells. It could be a good idea to thus share some data together with this paper, for instance in a public repository like Zenodo.

In figure 5 panel A, the meaning of the coloured arrows is described in the main text but not in the figure legend. Please add it there too.

Closure rate and data-fitting

Proposing an exponential decay for the gap size over time is interesting. I have a few suggestions here though.

In *Dynamics of wound closure* P.11, the motivation for the exponential decay is clear, yet it would be good to provide a "biological" interpretation for the parameter lambda. I would actually favour an equivalent form of the exponential decay equation, that uses the t1/2 "half-life" instead of lambda (the 2 being related, see https://en.wikipedia.org/wiki/Exponential_decay#Half-life).

This way the reported t1/2 is the time for the wound to regenerate half of the original wound size. The shorter the t1/2 the faster the regeneration. I think it would be a more suitable parameter for end-users to make sense of it.

When comparing the R^2 for the linear and exponential decay (Figure S4), I think it could be good to show the individual datapoints, as what's interesting here is to compare for a given data series, the R^2 for linear vs exponential. I would thus suggest a scatter plot, with the R^2 value on the y-axis and the different tools on the x-axis. Then for each tool showing the R^2 for the 3 replicates, using different colour for linear and exponential. If possible, using one type of marker per replicate so one can identify pairs of R^2 values to compare (linear vs exponential).

In paragraph *Performance comparison of CSMA, MRI, and HTM algorithms* / figure 5, it would be good to provide besides the R^2 the actual values for the parameter lambda to see if there is any major difference for the evaluated closure rate between the tools. It could be a small table as an additional panel D.

In figure 4.B, the linear fit is made on 2 separate sections of the curve ("fast" and "slower" phases). It could be interesting to show the variability of the parameter lambda estimated by the exponential decay using only the first part of the curve (until about 28 hours) compared to the value with the full curve. This would quantify the robustness of the exponential fit, especially to know if it's important to have datapoints covering the later slower phase/ "plateau" region to have a reliable fit with the exponential model. If so, then studies covering only shorter durations should rather stick to the linear fit. This could be a good addition to discussion in my opinion.

Is the slow down of the closure rate really a biological characteristic of tissue regeneration, or could it be due to depletion of nutrients from the media over time for instance ? Do the authors also observed almost full regeneration, with a percentage of original gap close to 0% within the 48 hours or for longer incubations. I don’t think more data is needed here, but this could be mentioned in discussion too.

Finally, when reporting mean values with +/- uncertainty, please indicate the type of the uncertainty range (STD I supposed). This can be in bracket (mean +/- STD).

Implementation

For Equation 1, the formula of the normalised gap ratio, shouldn’t the denominator rather be A_0*(pixel)* instead of A'_t*(pixel)* since it is the initial gap area in pixels at timepoint 0.

The notation A_t*(pixel)* and A_t*(%)* actually suggests that *pixel* and % are variables of the function A, while the variable here is rather the time. I would thus suggest the following notation

$$A\%(t) = \frac{Apixels(t)}{Apixels(0)}$$

Maybe for discussion, I feel like the normalisation of the gap area might not be completely necessary to compare the regeneration between different conditions, since one could still compare growth rates in pixels/time which is basically the value for the slope, while disregarding the original wound area (the intercept), similar reasoning would apply for the exponential decay. The normalisation helps with the visualization though, but could hide disparities in the original wound area betweeen samples, so both original and normalised data are somehow informative.

While the area of the wound is the main metric described in the paper. The tool can also quantifiy the average gap width, from one edge of the wound to the other ("width-based approach"). One suggestion here would be to provide, besides the average distance from one side to the other, either the standard deviation or an histogram of the computed distances. I think this could be useful to quickly identify when the wound is not a regular gap all along the image height, which could happen when the creation of the wound was not ideal.

Image processing

The image-processing workflow is extensively described but I found it still hard to grasp, especially in figure 2 since the different detection pipelines are interconnected. I think a reworking/simplification of this figure is really needed.

Both the "first mask creation" and "cell edge detection" pipeline have for output an "Image with detected wound edges", but it's not really clear what's the difference.

Is it the same input image for the "First mask creation" and the "Wound edge detection" ? the first step "Contrast enhancement and Gaussian blur" is identical with same parameter values according to the text but the images shown after this step look very different in both pipelines.

In discussion page 21, it is stated that "*the CSMA plugin applies the mask from the previous image to the current one*", suggesting some kind of recursion, which is not conveyed in figure 2. Is the "first mask creation" pipeline applied to the first image of the timelapse only then? Maybe reformulate as "Mask for image at t0".

Page 9 what is meant by "overlaid" in the sentence *"the black-and-white binary mask created earlier is overlaid on top of the newly obtained image to ensure that no residual holes remain in the cell monolayer in the output image".* Is it a binary AND operation?

What is meant by "combining pipelines" in the sentence "*Finally, the wound edge and cell edge detection pipelines are combined to produce an image with refined wound edges and cells within the wound.*".

In figure 2, both terms "wound edges" and "wound contour" are used and illustrated with similar thumbnails. Is it the same operation? If so please use one of the 2 terms only to avoid confusion, otherwise clarify the difference.

Typos

I spotted in Material and Methods, a recurrent typo for the symbol of the temperature unit (a z instead of °C).

Legend of figure S1, "wound" is mistyped as "would", also in that sentence missing an "s" to

"the two yellow arrow<u>s</u>".

Conclusion

While the proposed plugin might not be the simplest to install and use, the content of the article is still very informative, and opens interesting perspectives for discussion or similar developments. Besides the code is available and documented. If the authors can address most of the comments above, I am convinced this would be a valuable scientific publication.

Declarations

Potential competing interests: No potential competing interests to declare.