Cytometry
PART A
Journal of the International Society for Advancement of Cytometry

NeuronCyto II: An Automatic and Quantitative Solution for Crossover Neural Cells in High Throughput Screening

Kok Haur Ong,[1] Jaydeep De,[2] Li Cheng,[2]* Sohail Ahmed,[3]* Weimiao Yu[1]*

[1]Central Imaging Facility, Institute of Molecule and Cell Biology (IMCB), a*STAR, Singapore

[2]Imaging Informatics Division, Bioinformatics Institute (BII), a*STAR, Singapore

[3]Neural Stem Cell Lab, Institute of Medical Biology (IMB), a*STAR, Singapore

Received 30 November 2015; Revised 4 April 2016; Accepted 21 April 2016

Grant sponsor: A*STAR JCO, Grant number: 1231BFG040

Additional Supporting Information may be found in the online version of this article.

*Correspondence to: Li Cheng; 61 Biopolis Drive, Proteos, Singapore 138673. E-mail: chengli@bii.a-star.edu.sg OR Dr. Sohail Ahmed; 61 Biopolis Drive, Proteos, Singapore 138673. E-mail: sohail.ahmed@imb.a-star.edu.sg OR Dr. Weimiao YU; 61 Biopolis Drive, Proteos, Singapore 138673. E-mail: wmyu@imcb.a-star.edu.sg
Kok Haur Ong and Jaydeep De have contributed equally to this work.

Kok Haur Ong and Weimiao Yu are software developers for *NeuronCyto II* software package.

Published online 27 May 2016 in Wiley Online Library (wileyonlinelibrary.com)

DOI: 10.1002/cyto.a.22872

© 2016 The Authors.

Abstract

Microscopy is a fundamental technology driving new biological discoveries. Today microscopy allows a large number of images to be acquired using, for example, High Throughput Screening (HTS) and 4D imaging. It is essential to be able to interrogate these images and extract quantitative information in an automated fashion. In the context of neurobiology, it is important to automatically quantify the morphology of neurons in terms of neurite number, length, branching and complexity, etc. One major issue in quantification of neuronal morphology is the "*crossover*" problem where neurites cross and it is difficult to assign which neurite belongs to which cell body. In the present study, we provide a solution to the "crossover" problem, the software package *NeuronCyto II*. *NeuronCyto II* is an interactive and user-friendly software package for automatic neurite quantification. It has a well-designed graphical user interface (GUI) with only a few free parameters allowing users to optimize the software by themselves and extract relevant quantitative information routinely. Users are able to interact with the images and the numerical features through the *Result Inspector*. The processing of neurites without crossover was presented in our previous work. Our solution for the "crossover" problem is developed based on our recently published work with directed graph theory. Both methods are implemented in *NeuronCyto II*. The results show that our solution is able to significantly improve the reliability and accuracy of the neurons displaying "crossover." *NeuronCyto II* is freely available at the website: https://sites.google.com/site/neuroncyto/, which includes user support and where software upgrades will also be placed in the future. © 2016 The Authors. Cytometry Part A Published by Wiley Periodicals, Inc. on behalf of ISAC. This is an Open Access article under the terms of the Creative Commons Attribution-Noncommercial License which permits use, distribution and reproduction in any medium, provided that the Contribution is properly cited and is not used for commercial purposes.

Key terms

neurite tracing; quantitative information; neurite outgrowth; software development; high throughput screening

Neurons consist of an axon and dendrites, or neurites. The nervous system processes and transmits biochemical and electrical information via neurites and their connections (synapses). Thus, investigating the morphology of individual neurons and their connections is fundamental to understanding the nervous system. Nowadays, the advancement of microscopy has enabled us to acquire large amount of high quality neural images using image-based High-Throughput Screening (HTS) experiments. The morphology of neurons is very unique compared with other cell types. Important parameters to understand neuronal morphology include neurite length and number, branching and complexity, *among others*. Quantification of such information will be useful in drug discovery programs aiming at therapies for neuronal regeneration and for brain disorder such as Alzheimer's, Huntington's, and Parkinson's disease.

Manual assessments of neuronal morphology using visual observation (1) and visual scoring (2,3) approaches were first available to and applied by biologists to pro-

ISAC
International Society for Advancement of Cytometry

Table 1. Summary of existing software packages for neurite outgrowth quantification

SOFTWARE NAME	KEY MEASUREMENTS	PLATFORM	APPROACH	EXPORTED MEDIA
Neurite-J (7)	Neurite intersection and neurite occupied area	ImageJ	Semi-automated	Annotated image and text
NeuronMetric (5)	Primary neurite count, neurite length, branch number, quantity of soma, and soma's size.	ImageJ	Semi-automated	Annotated image and text
NeuriteTracer (10)	Neurite length, quantity of soma.	ImageJ	Semi-automated	Annotated image and text
NeuriteQuant (8)	Neurite length and quantity of soma and soma's size.	ImageJ	Automated	Web-based HTML format
NeuriteIQ (9)	Neurite length, quantity of soma and soma's size.	Matlab	Automated	Excel
HCA-Vision (11)	Primary neurite count, neurite length, branch number, quantity of soma, and soma's size.	C, C++, C#	Automated	Annotated image, built-in database and CSV
NeurphologyJ (12)	Neurite length and quantity of soma and soma's size.	ImageJ	Automated	Annotated image and text
NeuroCyto (13)	Neurite length, neurite complexity, branch level, quantity of soma, and soma's size	Matlab	Automated	Annotated image and text

cess a small number of images. However, in the HTS experiments, manual assessments of the cellular images acquired under different experimental conditions are labor intensive, time-consuming and tedious, even if feasible. Such analysis is also subjective, highly dependent on the experience, skill and knowledge of the end users and prone to bias, inconsistency and inaccuracy. Thus, the generation of an image-processing tool that can automatically and quantitatively analyze neuronal morphology becomes critical. In recent decades, much effort has been made to develop computer-assisted approaches and/or fully automatic methods for the assessment of neuronal morphology. There are several semi-automated and computer-assisted quantitative solutions proposed, such as Refs. 4–7. In general, these approaches require intensive user interaction, for example, the allocation of "seeds" (annotation) inside or outside the cell in Neurite-J[1] and NeuronMetric. More recently, a few automated approaches have been reported (8–12). Obviously, they are becoming a more preferable solution than the semi-auto methods because automated solutions can efficiently extract relevant, reliable and quantitative information from HTS experiments with limited user interference. In addition, the intra/inter-observer variation and measurement errors can be minimized. Several software packages based on the automated neuronal morphology and neurite outgrowth analyses have been published, such as NeuriteQuant (8), Neurite Image Quantitator (NeuriteIQ) (9), NeuriteTracer (10), HCA-Vision (11), NeurphologyJ (12), and NeuronCyto I (13,14). The popular semi-auto or fully automatic solutions are summarized in Table 1.

Under certain experimental conditions, the neurites may grow very long, that is, from short neurites without *crossovers* to long neurites with substantial amount of *crossovers*, as shown in the acquired image of Figure 2C. Such *crossovers* between neurites bring additional challenges to automatic analysis. Most of the automatic solutions for quantitative neurite analysis currently work well with the neurites that do not display crossovers. We have developed a two-step directed graph theoretical approach to address the "crossover" problem (15). Our solution is to reformulate the neurite tracing problem as a label propagation of a directed graph model such that the crossed neurites can be separated with each other. Therefore, we are able to extend our analysis capability from short non-crossover neurites to long crossover neurites. Besides the capability of solving the crossover, another major improvement compared with our previous work (13) is that we build a user-friendly graphical user interface (GUI). Our package, named *NeuronCyto II*, allows users with limited computer programming skills and image processing knowledge to extract the useful quantitative information independently. Our software package and other related materials, such as a user manual, demo movies and testing images, are freely accessible at https://sites.google.com/site/neuroncyto/.

Implementation And Methods

The implementation of *NeuronCyto II* with GUI, as shown in Figure 1A, aims to minimize human intervention and to be user friendly, facilitating better performance in terms of accuracy, efficiency, and reliability. In the GUI design, we optimized the number of parameters and focused on several key parameters, which can be modified by end users in a simple manner. The limited number of parameters reduces the complexity of parameter tuning process. Furthermore, it minimizes human intervention and reduces the potential human errors. We implemented *Batch Processing* function, highlighted by the red arrow in Figure 1A, to automatically quantify all acquired images under the similar conditions. The

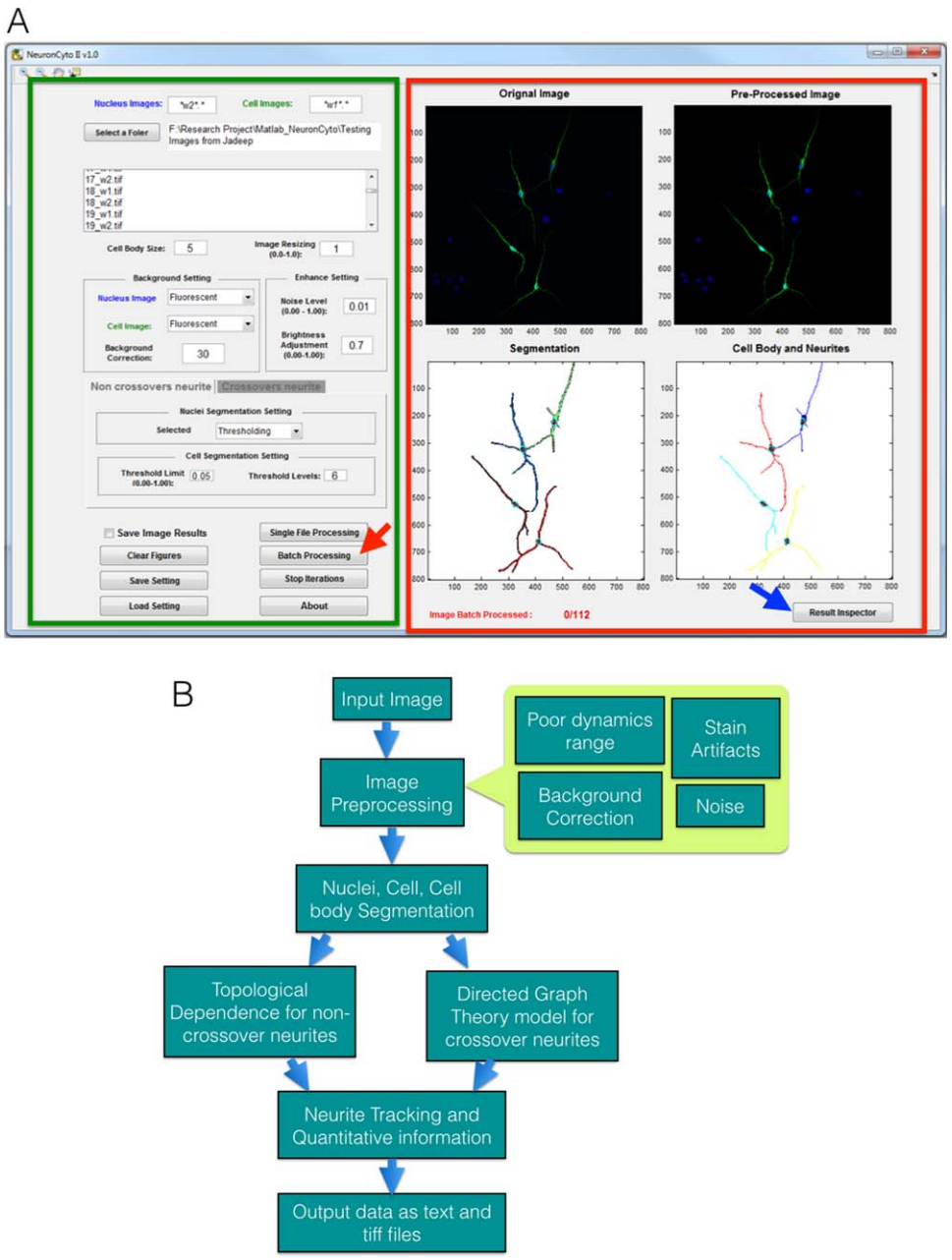

Figure 1. Introduction of *NeuronCyto II* software package. (Panel **A**). The interface of *NeuronCyto II*. Left side, indicated by green frame, is the control panel for the image loading and parameter selection; right side, indicated by the red frame, is the visualization of the images, intermediate results, segmentation and the final tracing results. *Red arrows* highlighted the *Batch Processing* function; *blue arrow* indicates the results visualization and interacting tool, that is, the *Results Inspector*. (Panel **B**). Processing pipeline of *NeuronCyto II*. It includes a few steps, such as image loading, image pre-processing, segmentation and separation of neurons, neurite tracing, quantitative information extraction and data output.

automatic neurite outgrowth quantification pipeline of *NeuronCyto II* is illustrated in Figure 1B, including image input, image preprocessing, segmentation, neurite tracing, quantification, and results output.

Image Loading and Preprocessing

The main interface of *NeuronCyto II* is shown in Figure 1A. The left panel of the interface, indicated by the green frame, is the control panels, including both parameter input and functional buttons. Two channel images are required as input for *NeuronCyto II* as shown in Figure 1A. They are cell image and soma/nucleus image. In the acquired raw images, we often face some issues of image qualities, including indirect immunofluorescence staining, noise and non-uniform background. Since the image quality issues significantly affect the accuracy of segmentation, filtering non-uniform background,

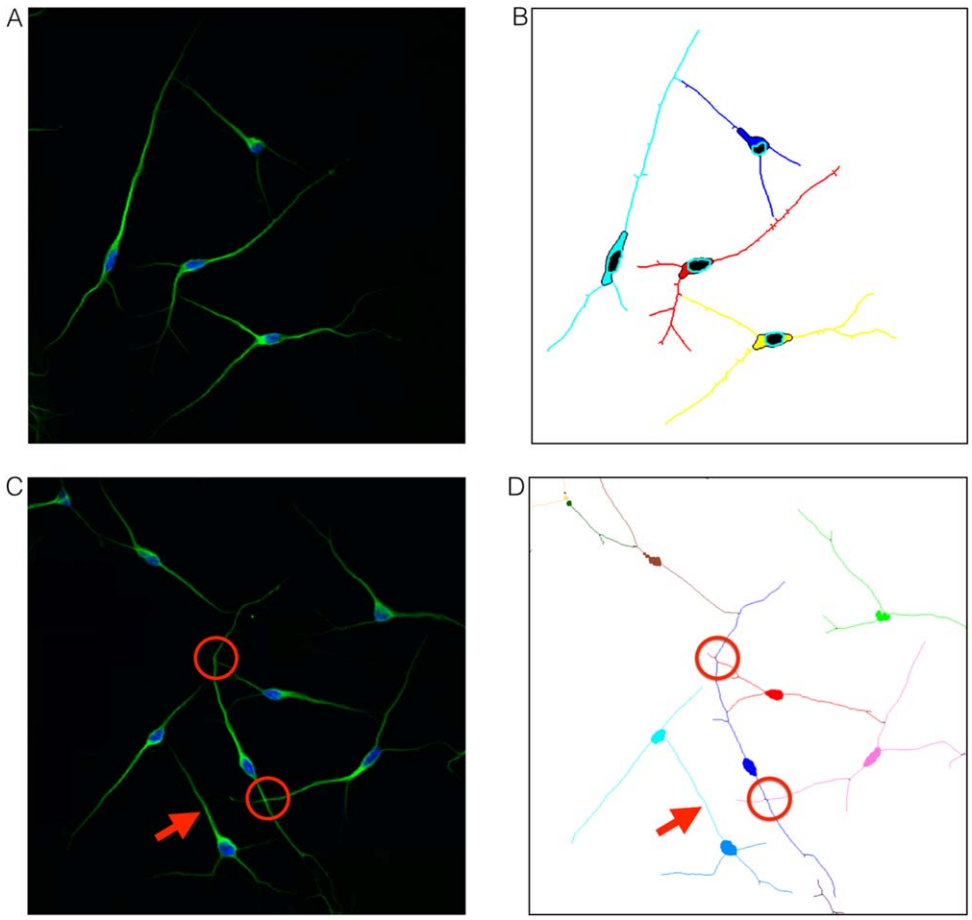

Figure 2. The neurite tracing results generated by *NeuronCyto II* for the images with and without crossover problems. The original images without crossover and with crossover are shown in Panel **A** and Panel **C**, respectively. The tracing results are shown in Panel **B** and Panel **D**, respectively. The *red circles* in Panel **C** and **D** illustrate the events of crossovers in the given image.

and removing noise are critical steps. Image preprocessing algorithms, such as intensity normalization, top-hat filtering and intensity contrast adjustment, are implemented to address the image quality issues and further improve the quality of image. (A separate technical document of the image preprocessing is provided as Supporting Information in this work.) The right panel of the interface, indicated by the red frame in Figure 1A, is the image viewers. In our GUI design, both cell and soma channels are merged together into green and blue channel respectively, as shown in Figure 1A. Merged image is visualized in the image viewers as original image and pre-processed image. Output of preprocessed image will be updated in real-time on image viewers when users modified any parameters on the left side. Hence, the impacts of any parameter adjustments are visualized in real-time and this facilitates the users to efficiently select proper parameters at each step.

Nucleus/Cell Segmentation

Cell segmentation for neuronal morphology assessment is not a trivial issue and much work has been done to address this problem. In *NeuronCyto II*, two different pipelines are designed for cell segmentation depending on whether or not the cells display "crossover." Our tracing algorithms are adapted and improved based on Refs. 13,14, and 16.

Neurites Without Crossover

Typical neurites without crossovers are shown in Figure 2A. In our previous work, we have shown the topological dependence algorithm (13,14) is fast, reliable, and accurate. However, our previous work did not provided a user interface for the end users. It is rather difficult for the users with limited programming and image processing skills to apply our scripts/code for their studies. *NeuronCyto II* solves this problem. In the GUI implementation, the soma/nucleus channel will be used to detect the identity each cell. Then cells will be detected and segmented based on topological dependence as described in Ref. 13. Furthermore, we simplified complex parameter setting in the segmentation process with only two important parameters. They are namely *threshold limit* and *threshold level*. Institutively, the *threshold limit* is to identify the foreground (lower value indicates that weaker signal will be considered as part of some cells) and *threshold level* controls the quality of the cell segmentation (greater threshold level will in general provide more accurate segmentation although more time-consuming.)

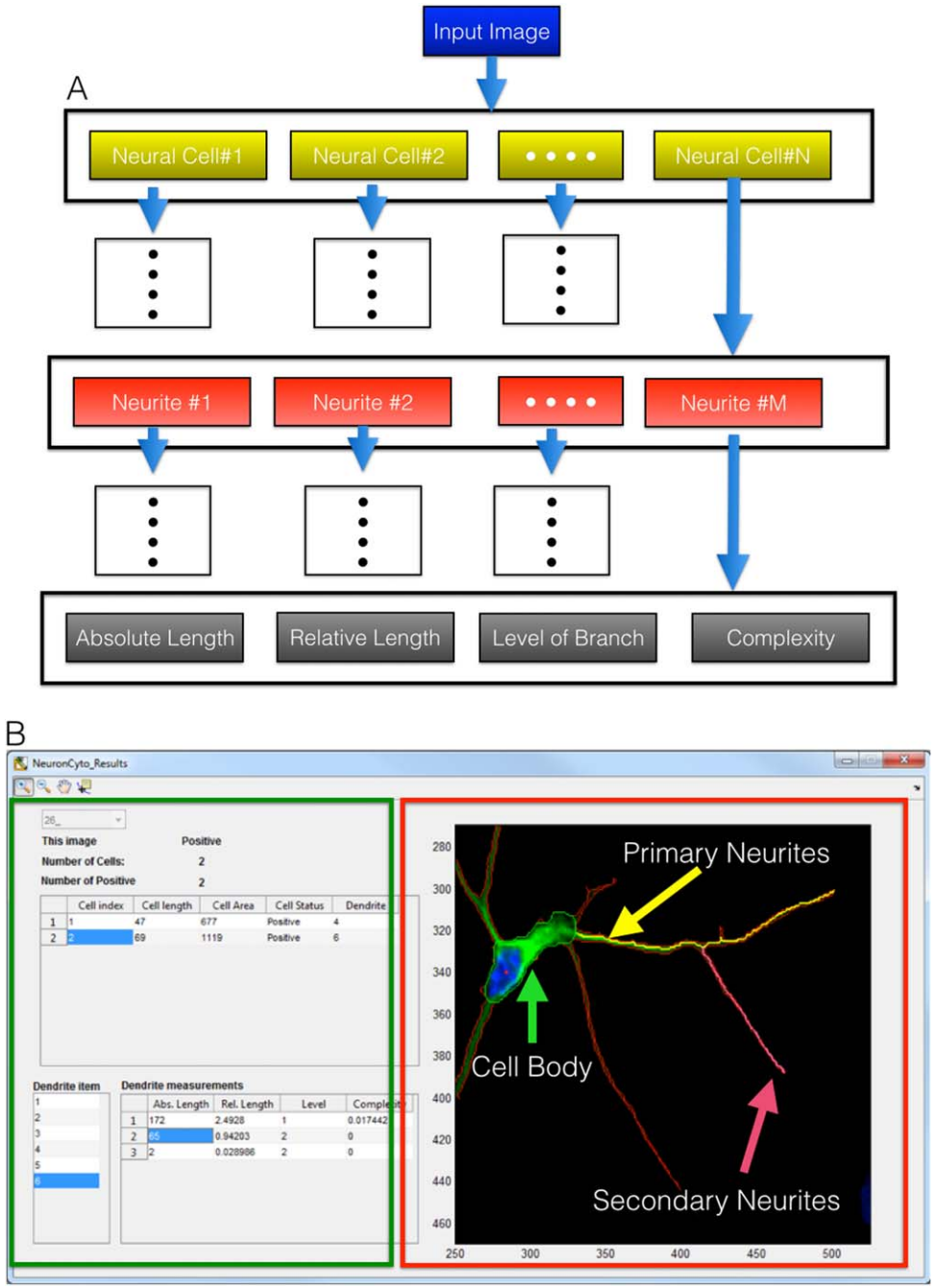

Figure 3. The data structure of *NeuronCyto II* and the interactive Results Inspector. Panel **A** shows the three levels of quantitative information, that is, image, cell, and neurite levels. The quantitative information is outputted into formatted text file for further analysis. Panel **B** illustrates the interactive *Results Inspector*. The left side, indicated by the green frame, is the numerical features of three different levels; the right side, indicate by the red frame, is the visualization of selected features. The cell body, primary neuritis, and the secondary neurites are highlighted by the *green*, *yellow*, and *pink arrows*, respectively. This tool allows the user to interact with acquired images and the numerical features. It is also a platform to validate the quantitative results and eliminate any potential errors.

Finally, neurites of each cell are segmented and visualized with random color for each cell in the image viewer as shown in Figure 2B.

Neurites With Crossover

In certain experimental conditions, for example, some compounds are able to significantly induce the neurite outgrowth, the neurons will produce very long neurites and easily cause the "crossovers," as shown by the red circles in Figures 2C and 2D. Identifying and tracing neurite from original cells is critical for quantitative neurite outgrowth analysis. Exiting application such as NeuriteTracer (10), HCA-vision (11), Vaa3D neuron module (17), Simple Neurite tracer (18) and

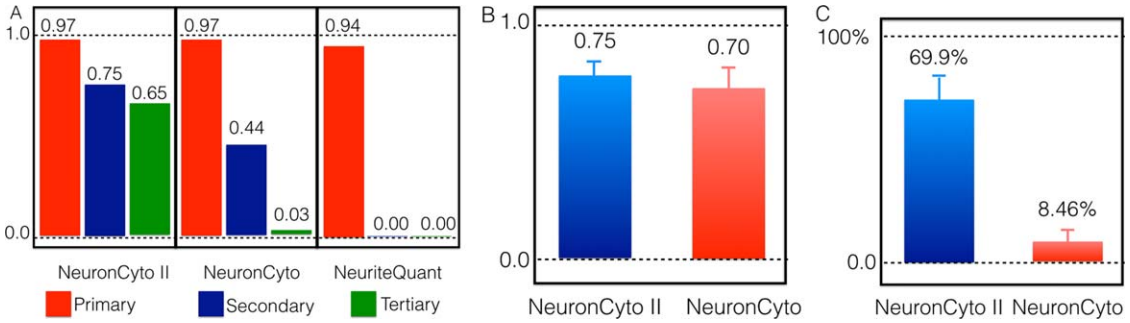

Figure 4. The quantitative comparison of *NeuronCyto II* with existing solutions on the images with crossover problems. Panel **A** shows the correlation coefficient of *NeuronCyto II*, NeuronCyto, and NeuriteQuant to the ground truth for the selected images with crossover problems. The three software packages achieved similar results at primary neurites, while *NeuronCyto II* significantly improved the accuracy of secondary and tertiary neurites. Panel **B** illustrates the accuracy of segments detection between *NeuronCyto II* and NeuronCyto. We have about 5% improvements. Panel **C** shows the accurate rate of separation of crossover neurites. We can see that *NeuronCyto II* has much better performance to separate the crossed neurites.

tracing module of Metamorph NX[1] are designed for neurite tracing tasks, while the presence of such neurite crossovers is still a challenging task (15). Recently, we presented a new neurite tracing theory based on directed graph model (15). Our method is tested and shows promising results for neurite with crossovers (15). In *NeuronCyto II*, we implemented our graph-theoretical approach to isolate the long neurite structure with crossovers. The approach consists of two steps. First, we segment the cells and their neurites from the background using an approach that penalizes the likelihood estimated with graph Laplacian (15). Second, the neurite will be separated based on digraph-based label propagation using Matrix-forest theorem. In *NeuronCyto II*, images are binarized using threshold method. The binary image can be visualized in real-time when the end users determine the optimized threshold parameter. In the final step, neurites of different cells are separated and differentiated from each other, as shown in Figure 2D.

Neurite Tracing and Quantitative Parameters

Once the neural cells are separated, the final segmentation will be provided to neurite tracing module to quantify the relevant features of each neuron. In our previous work (13), we presented our neurite tracing algorithms. Our tracing algorithm is based on the skeletons of the binary images. We defined pixels on skeletons into the five different categories: Root Point, Body Point, Node Point, Branch Point, and Leaf Point. Our tracing algorithm then automatically searches around the boundary of each cell body. When it encounters a neurite root point, it will follow the neurite and quantitatively measure the lengths of its branches. Every traced point will contain the information of its distance from the root point and the point it is being traced from. Due to the limitation of space, we refer readers to our previous work for more details on the segmentation and tracing algorithms (13,14).

In *NeuronCyto II*, the cells are visualized and evaluated by end user to understand the impact of different conditions, such as size, shape, appearance, and complexity of neurite. These features are presented at three levels: individual image, individual cell, and individual neurite, as shown in Figure 3A. In order to further enhance the interaction between numerical parameters and images, we developed an interactive *Results Inspector*, as shown by the blue arrow in Figure 1A. The interface of *Results Inspector* is illustrated in Figure 3B. The left panel, indicated by the green frame, is the quantitative features extracted from the given image. The right panel is the visualization of quantified structures. The primary and secondary neurites are highlighted by yellow and pink lines respectively. This interactive tool bridges images and the phenotypic results. This tool allows the users to validate and filter the obtained results to further improve the reliability of their data. For more details, please refer to our online demo and our user manual for *Result Inspector*.

Results

Accurate quantification of neurites with crossover relies on the precise separation of neurons from each other. Our proposed approach (15) is based on Matrix Forest Theorem on Directed Graph to minimize the given cost function such that we can achieve the correct separation of neurons. In this section, we select the images with crossover neurons and compare the performance of *NeuronCyto II* with our previous work and other existing solutions. For the performance of the neuron without crossover problem, we refer readers to our previous work (13,14).

All images in this section contain the crossover issues. The branching level analysis is a key perspective of neurite outgrowth. We use NeuronJ (19) (ImageJ plug-in) to create the ground truth of about 10 images with approximately 100 neurons. The neurites of each cell are manually annotated and the quantitative measurements are extracted. First of all, we compared the accuracy of branching level between *NeuronCyto II*, NeuronCyto (13) and NeurtieQuant (8). Figure 4A shows the correlation coefficients of these three packages for primary, secondary, and tertiary neurite outgrowth of each image. For the primary branch, the three solutions achieve

[1]http://www.moleculardevices.com/systems/metamorph-research-imaging/metamorph-nx-microscopy-automation-image-analysis-software

similar accuracy, while it is not supervising that *NeuronCyto II* has better accuracy for secondary and tertiary branch measurement compared with the other two methods because it can better solve the crossover problems in the given images. NeuriteQuant is an automatic solution to analyze the neural cell; however, it can only extract the information for primary neurite and is not able to solve the crossover problem.

On a cell-by-cell basis, we measured the correct segments for *NeuronCyto II* and NeuronCyto as shown in Figure 4B. We can see that the accuracy of neurite segments are improved about 5% by *NeuronCyto II*. This is because not all the neurons will display the "crossover." If we only consider the neurons with "crossover" problem, the improvement will be more significant.

Another important parameter is the accurate ratio of *NeuronCyto II* for a given number of crossover problems presented in those images. Based on our ground truth, we calculate this ratio as N/M, where N is the number of correct separation of crossovers by *NeuronCyto II* and M is the total number of crossovers in an image. The accurate ratio is improved from less than 10% to about 70% as shown in Figure 4C. As we can see, the reliability of the neuron with crossover problem is significantly improved and we are able to describe the neuron with crossovers in a more accurate and reliable manner. The original testing images and raw measurements for comparisons in the section are provided as Online Supporting Information.

Discussion

It is not trivial to process large amount of images acquired by HST experiments. It is probably more challenging to extract reliable and quantitative information from neural cells automatically. In the past ten years, much work has been done by both biologists and computer scientists to address this problem. The field is growing rapidly. However, the "crossover" problem is still to be resolved. Furthermore, applying those algorithms to different dataset is not straightforward and this is especially true for the biologists, as it generally requires coding and image processing skills. There is a big gap from good algorithms to a user-friendly and usable software package, which can be routinely used by the biologist to process their acquired images. The development of a software package for a specific scientific problem requires systematic work. Such software packages need to be carefully designed and engineered.

In the HST experiments, the neurites tend to grow very long under certain experimental conditions and they will inevitably crossover. We have proposed a directed graph approach to address this "crossover" problem. In this work, we present our software package, *NeuronCyto II*, with a user-friendly interface of limited parameters, which includes the implementation of our directed graph approach. Our software package provides a comprehensive solution to neuronal morphology analyses and also addresses the "crossover" problem in general. Nevertheless, there is still room to further improve the performance and accuracy of our software. For example, there is a mistake of crossed neuron separation as highlighted by the red arrow in Figures 2C and 2D. The reason is that although our solution is able to separate most of the crossovers, we have not yet to integrate the "thickness" and "brightness" as a clue into our graph model. If such constraints are incorporated in our model, we expect that the overall accuracy of separation will be further improved.

Bioimage analysis software packages generally require a few free parameters to be determined by the end users. Once a software package is designed, the accuracy and performance will rely on those "user-defined" parameters. However we found parameter tuning is not a trivial task for the users who lack image analysis experience. This is a common issue for many software packages. It is critical to minimize the number of free parameters. Let us consider the performance of a software package as a function of those given free parameters, where the users have to find the optimized performance by tuning some image processing parameters even if they do not really understand their meaning. This blind parameter tuning processing is tedious and painful for the end users. Users with more knowledge and skills will uncover the "good" parameters faster while it might be slower and more difficult for the other users with less experience. To improve this, we strive to lessen the parameter tuning burden in a more natural, friendly and interactive way. As shown in the online demo video, the user could immediately visualize the results once they adjust a parameter. Such visual feedback loop is critical for the user to quickly find the optimized parameters and achieve optimal results.

The interactions between the end users, the images and the quantitative parameters are important. To enhance such interaction, we build a dedicated Result Inspector as shown in Figure 3B. It allows the users to directly interact with numerical parameters, the detected structures and acquired images. This tool also facilitates the users to validate the results, filter away unreliable readings and further improve the accuracy of their measurements. Eventually, the level of users experience and skill is important for the practical adoption of a software package. We have prepared detailed user manual and demo videos for our end users. Those online materials will help the user to get familiar with our package quickly and apply it to their daily research activities. We established a website at https://sites.google.com/site/neuroncyto/. It is a platform to provide technical assistance for users. The end users will also be able to feedback the software developers with potential bugs. We will also release our further improvements and software upgrade on this website.

Literature Cited

1. Lemmon V, Burden SM, Payne HR, Elmslie GJ, Hlavin M. Neurite growth on different substrates: Permissive versus instructive influences and the role of adhesive strength. J Neurosci 1992;12:818–826.
2. Tsui CC, Copeland NG, Gilbert DJ, Jenkins NA, Barnes C, Worley PF. Narp, a novel member of the pentraxin family, promotes neurite outgrowth and is dynamically regulated by neuronal activity. J Neurosci 1996;16:2463–2478.
3. Rutishauser U, Gall WE, Edelman GM. Adhesion among neural cells of the chick embryo. IV. Role of the cell surface molecule CAM in the formation of neurite bundles in cultures of spinal ganglia. J Cell Biol 1978;79:382–393.
4. Orike N, Thrasivoulou C, Cowen T. Serum-free culture of dissociated, purified adult and aged sympathetic neurons and quantitative assays of growth and survival. J Neurosci Methods 2001;106:153–160.
5. Narro ML, Yang F, Kraft R, Wenk C, Efrat A, Restifo LL. NeuronMetrics: Software for semi-automated processing of cultured neuron images. Brain Res 2007;1138:57–75.

6. Fanti Z, De-Miguel FF, Martinez-Perez ME. A method for semiautomatic tracing and morphological measuring of neurite outgrowth from DIC sequences. 2008 EMBS 2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 2008. pp 1196–9.
7. Torres-Espín A, Santos D, González-Pérez F, Del Valle J, Navarro X. Neurite-J: An Image-J plug-in for axonal growth analysis in organotypic cultures. J Neurosci Methods 2014;236:26–39.
8. Dehmelt L, Poplawski G, Hwang E, Halpain S. NeuriteQuant: An open source toolkit for high content screens of neuronal morphogenesis. BMC Neurosci 2011;12:1–14.
9. Zhang Y, Zhou X, Degterev A, Lipinski M, Adjeroh D, Yuan J, Wong STC. Automated neurite extraction using dynamic programming for high-throughput screening of neuron-based assays. Neuroimage 2007;35:1502–1515.
10. Pool M, Thiemann J, Bar-Or A, Fournier AE. NeuriteTracer: A novel ImageJ plugin for automated quantification of neurite outgrowth. J Neurosci Methods 2008;168: 134–139.
11. Wang D, Lagerstrom R, Sun C, Bishof L, Valotton P, Götte M. HCA-vision automated neurite outgrowth analysis. J Biomol Screen 2010;15:1165–1170.
12. Ho S-Y, Chao C-Y, Huang H-L, Chiu T-W, Charoenkwan P, Hwang E. NeurphologyJ: An automatic neuronal morphology quantification method and its application in pharmacological discovery. BMC Bioinform 2011;12:230.
13. Yu W, Lee HK, Hariharan S, Bu W, Ahmed S. Quantitative neurite outgrowth measurement based on image segmentation with topological dependence. Cytometry Part A 2009;75:289–297.
14. Yu W, Lee HK, Hariharan S, Bu W, Ahmed S. Evolving generalized Voronoi diagrams for accurate cellular image segmentation. Cytometry Part A 2010;77:379–386.
15. De J, Cheng L, Zhang X, Lin F, Li H, Ong KH, et al. A graph-theoretical approach for tracing filamentary structures in neuronal and retinal images. IEEE Trans Med Imag (TMI) 2015;35:257–272.
16. Yu W, Lee HK, Hariharan S, Bu W, Ahmed S. Detection and quantitative measurement of neuronal outgrowth in fluorescence microscopy images. Proceedings of Medical Image Understanding and Analysis (MIUA); 2009.
17. Peng H, Ruan Z, Long F, Simpson JH, Myers EW. V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. Nat Biotech 2010;28:348–353.
18. Longair MH, Baker DA, Armstrong JD. Simple Neurite Tracer: Open source software for reconstruction, visualization and analysis of neuronal processes. Bioinformatics 2011;27:2453–2454.
19. Meijering E, Jacob M, Sarria JCF, Steiner P, Hirling H, Unser M. Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images. Cytometry Part A 2004;58A:167–176.