EUSIPCO 2013 1569744383

ADVANCED IMAGE PROCESSING FOR BIOLOGY, AND THE OPEN BIO IMAGE ALLIANCE (OBIA)

Michael Unser, Daniel Sage and Ricard Delgado-Gonzalo

EPFL, CH-1015, Lausanne Switzerland

ABSTRACT

The field of biological imaging has evolved considerably during the past decade as a result of recent (r)evolutions in fluorescence labeling and optical microscopy. Bioimage informatics has been identified as a top priority to cope with the ever-increasing amount of microscopy data.

The challenges and opportunities for researchers in image and signal processing are manyfold. They span the areas of mathematical imaging, with problems such as denoising, 3-D deconvolution and super-resolution localization, as well as image analysis for the segmentation, detection and recognition of biological structures in 3-D. The dynamic aspect of the data requires the development of novel algorithms for tracking fluorescent particles and analyzing high-throughput microscopy data (labeling of cells, phenotyping, extraction of gene expression profiles).

A crucial aspect of bioimage informatics is making image analysis tools available to biologists so that they can be applied to real data and used on a routine basis. Developers may benefit from open-source frameworks and international initiative such as OBIA for easying-up this process and creating collaboration networks with biologists.

Index Terms— Bioimage analysis, open-source software, mathematical imaging, fluorescence microscopy

1. INTRODUCTION

Imaging in biology has evolved significantly during the past two decades due to major improvements in fluorescence labeling and the development of new high-resolution microscopes (e.g., confocal, two-photon, STED, PALM/STORM). Fluorescence microscopy is presently having a profound impact on the way research is being conducted in molecular biology. Biomedical scientists can visualize sub-cellular components and processes, both structurally and functionally, in two or three dimensions, at different wavelengths (spectroscopy), and can perform time-lapse imaging to investigate cellular dynamics [1]. Bioimaging devices generate a huge amount of high-dimensional data in high-resolution format. The sheer amount of data is such that it generally becomes infeasible to visually inspect them all; moreover, it is highly desirable to automatize the extraction of objective quantitative features [2, 3].

The data analysis and processing techniques that are currently used in the field, however, are still relatively crude if one compares them with the state-of-the art in medical imaging. Yet, there is a growing consensus that bioimage analysis software is of paramount importance for the future of biological research. To quote Gene Myers (Why bioimage informatics matters, Nature Methods, July 2012, pp. 659-660), "*bioimage informatics increasingly matters because of the increasing scale of the production of imagery and because of the increasing number of systems genetics explorations aimed at understanding the crucial physical and spatial nature of proteomics signals and machinery.*" The field is emerging as a key priority and is rapidly gaining in importance for all areas of bioimaging [4]. The community is largely relying on open-source software [5, 6]. Almost any biologist who is acquiring microscopic images will either be a user of ImageJ/Fiji [7, 8] or at least be aware of the existence of this tool and its alternatives such as CellProfiler [9] and Icy [10], among others. These bioimage analysis tools are extremely useful to biologists and microscopists who are typically not computer specialists, nor Matlab users. The present impact of such open source software is already quite sizable (and measurable in terms of citations). It can be expected to increase much further in the future as the tools become user-friendlier, the ultimate goal being to make bioimaging a more quantitative science.

The purpose of this paper is to make signal-processing researchers aware of the main trends and challenges in the rapidly-developing field of bioimage informatics. The first part presents to a brief review of the specificities of biological imaging, the basic workflow(s) in bioimage analysis, and the primary image-processing tasks to which researchers in signal processing can contribute by designing better algorithms. The second part is about software enabling technologies and good practices to produce image-analysis tools that are directly usable by microscopists and biologists. While producing user-friendly software is time-consuming and does typically not constitute the first priority of someone involved in signal-processing research, it is an aspect that cannot be ne-

This work was funded in part by the Euro-BioImaging research infrastructure project and by the Swiss SystemsX.ch initiative under Grant 2008/005.

glected in the interdisciplinary context of bioimaging. The good news is that there are powerful development frameworks (typically in JAVA) to ease the process and that the payoff in terms of scientific impact can be substantial. Indeed, there are thousands of users in the biological sciences in need of better image-processing tools and who are eager to apply them right away.

2. CHALLENGES AND OPPORTUNITIES FOR SIGNAL PROCESSING

Imaging in biology has evolved dramatically during the past decade due to major improvements in fluorescence labeling, optics and imaging sensors. The aspects that are specific to modern optical microscopy and contribute to making signal processing research in that area particularly challenging are (see [1]):

- the sophistication and variety of imaging techniques; the development during the past years have been truly phenomenal with new modalities such as STED, STORM and PALM overcoming Abbe's physical limit on resolution (by a factor between 2-10) [11].
- the increasing need for quantitative image analysis;
- images that are often very noisy, and at the limit of resolution;
- multiplicity of dimensions: 2-D or 3-D, time (dynamic imaging), multi-spectral.

We have organized our discussion of needs for advanced image processing around three primary topics.

2.1. Computational imaging

The quality of the micrographs, both in terms of resolution and signal-to-noise ratio, can be improved significantly by applying advanced signal-processing techniques.

Image denoising: There is a strong incentive to rely on denoising algorithms in order to gather images faster with less photons [12]. The typical source of noise is counting statistics (Poisson distribution). Note that noise reduction yields more impressive results when it is performed jointly on a high number of dimensions (3-D, 2-D+t, or even 3-D+t).

Deconvolution of fluorescence micrographs: This is one of the few areas of imaging where deconvolution can really make a difference, especially in the case of 3-D fluorescence microscopy [13]. The primary difficulty there is the huge size of the data. While total variation regularization has been applied to the problem, it tends to create unnatural staircase artifacts. This calls for higher-order methods. Another challenge is to be able to handle spatially-varying point spread functions which result from a non-constant refractive index within the specimen.

Quantitative phase imaging: A possible improvement over classical phase contrast microscopy is to apply an inverse-problem formulation to jointly recover the phase and amplitude of the optical wave. The challenge is to be able to do so under incoherent light illumination.

Super-resolution localization: Novel microscopy modalities such as PALM and STORM rely on the localization of individual point sources (single molecules) with an accuracy far beyond the traditional diffraction limit. The price to pay is a much longer acquisition time. It is of interest to develop more sophisticated estimation and/or deconvolution algorithms in order to be able to handle denser source distributions.

2.2. Shape and morphology

Biologists are in crucial need of quantitative methods for characterizing the shape of biological structures. The first step of such an analysis is to segment the image, or to detect objects of interest based on their morphology.

2-D and 3-D image segmentation: While segmentation is one of the oldest tasks in image processing, there is no generic algorithm that provides a universal solution. Consequently, there are many opportunities in the field for designing methods for specific classes of biological images. For instance, phase-contrast (or DIC) images are notoriously difficult to segment because there are primarily edge-based (difference of refractive index). The segmentation of fluorescence micrographs is largely dependent upon the type labeling used, the latter being under the control of the biologist. Typical sources of disturbance are the density of organelles, scattering, photobleaching, and the unavoidable presence of autofluorescence. Here, it helps to use prior shape information. Rather than aiming at a fully automated solution which is often illusive, it can make sense to rely on user-input to guide/correct the detection process. Active contours are especially helpful in that respect [14]. It is also desirable to provide some measure of reliability of the output so that the user can quickly focus on the errors (hopefully few) and correct them manually. The field is still crucially in need of good (and preferably, semi-interactive) segmentation tools for 3-D.

Detection of specific structures: Beside the nucleus of a cell which is typically blob-shaped, there are many characteristic 3-D structures in biology such as spots, vesicles, filaments, dendrites, membranes, etc. that call the design of specialized detectors. In 3-D, one also has to distinguish between different types of geometric varieties (e.g., lines vs. surfaces). One possibility is to design detectors based on steerable wavelets. The field of bioimage informatics is still missing the equivalent of the SIFT detectors which are so widely used in computer vision. The fundamental difference in context is that biological data is intrinsically 3-D and that invariance to projective geometry is irrelevant. On the other hand, it is highly desirable to enforce translation, scale, and rotation invariance.

2.3. Temporal analysis

The use of endogenous fluorescent markers such as GFP (Green fluorescence protein) allows *in vivo imaging* which enables the observation of dynamic biological processes, both at the cellular (5-10μm) and molecular ($< 1\mu$m) levels. Microscopists face the problem of analyzing and quantitating huge amounts of sensitive time-lapse image data.

Tracking cells and building lineage trees: The problem of tracking cells is central to high-throughput microscopy [3]. It is essential for extracting dynamic gene expression profiles, characterizing temporal relationships and establishing cell lineage [15]. One of the main difficulty is to be able to accommodate coarse temporal sampling to minimize the exposition of cells (photobleaching). The current conceptual challenge is to integrate the information from as many timeframes as possible in order to improve the robustness of the procedure. Besides the pairing of cells from one image to the next, one also needs to properly handle the problem of cell division. Conceptually, it would be preferable to address the segmentation and tracking problems jointly, which is typically not the way it is being done right now.

Tracking particles: As one moves to finer scales, one can start visualizing molecular processes that are highly dynamic. The computational task is then to detect and track individual fluorescent particles that can be very mobile and also densely packed [16]. They are often at the limit of resolution in a very noisy background. While fast imaging is quite feasible in 2-D, the difficulty is that the physical movement is intrinsically 3-D, meaning that the particles can easily move out of focus. Retrieving the information in the third dimension calls for innovative schemes, possibly in the spirit of "compressed sensing".

3. IMAGE ANALYSIS SOFTWARE

As already stated, the process of converting algorithms into robust, user-friendly bioimage analysis software is of paramount importance. In this section, we present a list of good practices for software development to ensure a successful conversion and maximize usability. We also briefly review the history and current state of the most popular open image-analysis platforms.

3.1. Software Design

The primary users of bioimage-analysis software are biologists with little or no programming training who are operating their microscope and analyzing their own data. They require user-friendly, well-supported, and flexible software to easily fulfill their particular needs [5]. The following list of good practices is aimed at facilitating the creation software that is usable and helpful to a broad segment of the bioimaging community [6]:

Fig. 1. Samsung Slate PC Series 7 running the open image analysis software Icy [10] and one of the plug-ins implementing the method of [14]. This is the result of the efforts of the open-source community of developers to produce an user-friendly image analysis software.

1. *User-friendliness*: The software should be intuitive, easy-to-use and accessible (one-click installation). Moreover, it should be accompanied with clear user manuals and offer feedback mechanisms (*e.g.*, forums, mailing lists, bug report systems) [17]. We show in Figure 1 an intuitive interface of an image analysis software running on a tablet computer.

2. *Developer-friendliness*: A good documentation of the structure of the code and of its modules is crucial since it allows developers to understand what a program does and how it works. Open-source software is a good example of developer-friendly software.

3. *Interoperability*: It is important to make software that communicates using the available open standards. In this way, different software can easily interact without having to define complementary components to translate the data. A successful example is the Bio-Formats project, a Java library for reading and writing life sciences image file formats [18].

4. *Modularity*: The implicit modularity of object-oriented design is key when maintaining a large piece of software. The use of modular structures with common interfaces allows developers to update their software with minimum effort.

5. *Validation and quality control*: The software should be tested in ways that are relevant to the user. Moreover, for the benefit of making research reproducible, it must be possible to replicate the exact same computations and quantitative results that the developers advertise. A recent trend is to define computational challenges for some well-defined bioimage analysis tasks such as deconvolution or particle tracking. This allows for objec-

tive performance assessment and comparison of algorithms and software solutions.

3.2. Open Image Analysis Platforms

In order to properly analyze an experiment and draw conclusions from the data provided by an image-analysis software, the biologist must be aware of what the algorithm really does. Open-source software provides the necessary transparency, giving scientists the opportunity to fully understand the computational methods behind their tools.

Among all open-source bioimage analysis tools, the one that has had the most impact so far is ImageJ [7]. It was initiated by Wayne Rasband at the National Institutes of Health (NIH) under the name of NIH Image. The idea was to develop a low-cost image-processing platform for the Apple Macintosh II. This piece of software was coded in Pascal, and had add-on capabilities in the form of expansion slots in order to enable other developers to easily extend the software for their own applications.

In the mid-nineties, the programming language Java was created by Sun Microsystems. Java applications are typically compiled to bytecode that can run on any machine regardless of the architecture. This allowed developers to write their software independently of the platform. Rasband ported NIH Image to Java in the late-nineties under the name of ImageJ. As a result, the base of NIH Image users and developers was extended to PC and Unix.

ImageJ upgraded the expansion slots of NIH Image into the more modular concept of plug-ins. Since its creation, ImageJ has enjoyed a great popularity, and resulted in the development of a wide variety of plug-ins for very diverse applications.

Besides the core application, another popular distribution is Fiji. It is a user-friendlier distribution of ImageJ together with Java, Java 3-D and the most prominent plug-ins as well as transparent installation and updates [19].

The largest upgrade of ImageJ since NIH Image is being prepared involving several research laboratories under the name of ImageJ2. It involves a full rewrite of the source code using new architectures in order to overcome the limitations of ImageJ.

Recently, other open-source related platforms are emerging. Among them, we can find: μManager, a software package for the control of automated microscopes [20]; CellProfiler, a software specialized in measuring phenotypes automatically within images [9]; and Icy, a full integrated easy-to-use platform extensible with plug-ins [10]. We summarize all these open-source projects in Table 1 [8].

3.3. Open Bio Image Alliance

The Open Bio Image Alliance[1] (OBIA) was constituted in 2012 with the aim of federating the development of the aforementioned image-processing packages and improving their interoperability. It is an international consortium that brings together the major developers of bioimage analysis software ranging from biologists, microscopists, computer scientists, to researchers involved in biomedical image processing. Given the mission of OBIA stated below, we recommend its web site as the primary entry point for gathering information about the open-source resources for bioimage informatics, both at the level of the users and developers.

The primary mission of OBIA is to

- *provide biologists and researchers in the life sciences with the highest quality public-domain software resources and a corresponding knowledge base to analyze and quantitate their image data in a sound and reproducible fashion,*

- *to strengthen the interaction between biologists, imaging scientists and developers of bio-image analysis software and algorithms.*

OBIA capitalizes on the existence of highly successful software packages such as ImageJ. [...] OBIA promotes long-term availability and backward compatibility, federates the harmonious community-based development of interoperable software, and promotes good software development practices. OBIA will meet these challenges by implementing mechanisms and initiating actions in order to:

- *facilitate the diffusion of bioimaging software and guide the choice of image analysis tools with special attention to quality (curation), long-term availability and (backward) compatibility;*

- *federate the harmonious community-based development of interoperable software and promote good practices, including the careful validation of tools;*

- *reinforce interactions between imaging scientists/developers and create a sense of community;*

- *be a catalyst for new software development projects, advanced image-analysis initiatives, and interdisciplinary collaborations in the computational and biological sciences.*

We can only encourage our colleagues to take part in this alliance, or, at least, to closely follow what is going on and available in terms of development tools and software/algorithm deployment channels and repositories.

[1] http://www.openbioimage.org/

	Initiated	Status	Language	License
NIH Image	1987	Discontinued	Pascal	Public domain
ImageJ	1997	Active	Java	Public domain
μManager	2005	Active	C++/Java	BSD, Lesser GPL
CellProfiler	2006	Active	Python	GNU
Fiji	2007	Active	Java	GNU
ImageJ2	2009 beta version	Under development	Java	Simplified BSD
Icy	2011	Active	Java	GPL

Table 1. Summary of open-source image-processing platforms.

4. CONCLUSION

We hope to have convinced people involved in signal processing of the strategic importance of bioimage informatics. So far, the field has been mostly defined by biologists who have become software developers by necessity. The topics are plentiful and challenging intellectually with a pressing need for better image processing and analysis tools.

Our advice to designers of new algorithms is to think about user interactions issues from the very start, to take advantage of existing software frameworks such as ImageJ and Icy, and to work in close interactions with biologists. This is the best way to maximize the impact of one's research output.

5. REFERENCES

[1] C. Vonesch, F. Aguet, J.-L. Vonesch, and M. Unser, "The colored revolution of bioimaging," *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 20–31, May 2006.

[2] Editorial, "The quest for quantitative microscopy," *Nature Methods. Special Issue Focus on Bioimage Informatics*, vol. 9, pp. 627, 2012, 2012.

[3] R. Pepperkok and J. Ellenberg, "High-throughput fluorescence microscopy for systems biology," *Nature Reviews Molecular Cell Biology*, vol. 7, no. 9, pp. 690–696, 2006.

[4] H. Peng, "Bioimage informatics: a new area of engineering biology," *Bioinformatics*, vol. 24, no. 17, pp. 1827–1836, 2008.

[5] A. Cardona and P. Tomancak, "Current challenges in open-source bioimage informatics," *Nature Methods*, vol. 9, no. 7, pp. 661–665, July 2012.

[6] A.E. Carpenter, L. Kamentsky, and K.W. Eliceiri, "A call for bioimaging software usability," *Nature Methods*, vol. 6, no. 7, pp. 666–670, July 2012.

[7] T.J. Collins, "ImageJ for microscopy," *BioTechniques*, vol. 43, no. 1, pp. 25–30, July 2007.

[8] C.A. Schneider, W.S. Rasband, and K.W. Eliceiri, "NIH Image to ImageJ: 25 years of image analysis," *Nature Methods*, vol. 9, no. 7, pp. 671–675, July 2012.

[9] M.R. Lamprecht, D.M. Sabatini, and A.E. Carpenter, "Cellprofiler™: Free, versatile software for automated biological image analysis," *BioTechniques*, vol. 42, no. 1, pp. 71–75, January 2007.

[10] F. De Chaumont, S. Dallongeville, N. Chenouard, N. Hervé, S. Pop, T. Provoost, V. Meas-Yedid, P. Pankajakshan, T. Lecomte, Y. Le Montagner, T. Lagache, A. Dufour, and J.-C. Olivo-Marin, "Icy: An open bioimage informatics platform for extended reproducible reresearch," *Nature Methods*, vol. 9, no. 7, pp. 690–696, July 2012.

[11] L. Schermelleh, R. Heintzmann, and H. Leonhardt, "A guide to super-resolution fluorescence microscopy," *The Journal of Cell Biology*, vol. 190, no. 2, pp. 165–175, 2010.

[12] F. Luisier, T. Blu, and M. Unser, "Image denoising in mixed Poisson-Gaussian noise," *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 696–708, March 2011.

[13] A. Griffa, N. Garin, and D. Sage, "Comparison of deconvolution software in 3D microscopy: A user point of view—Part 1," *G.I.T. Imaging & Microscopy*, vol. 12, no. 1, pp. 43–45, March 2010.

[14] R. Delgado-Gonzalo, P. Thévenaz, C.S. Seelamantula, and M. Unser, "Snakes with an ellipse-reproducing property," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 1258–1271, March 2012.

[15] D. Sage, M. Unser, P. Salmon, and C. Dibner, "A software solution for recording circadian oscillator features in time-lapse live cell microscopy," *Cell Division*, vol. 5, no. 17, July 2010.

[16] E. Meijering, I. Smal, and G. Danuser, "Tracking in molecular bioimaging," *Signal Processing Magazine, IEEE*, vol. 23, no. 3, pp. 46–53, 2006.

[17] D. Bolchini, A. Finkelstein, V. Perrone, and S. Nagl, "Better bioinformatics through usability analysis," *Bioinformatics*, vol. 25, no. 3, pp. 406–412, February 2009.

[18] M. Linkert, C.T. Rueden, C. Allan, J.-M. Burel, W. Moore, A. Patterson, B. Loranger, J. Moore, C. Neves, D. MacDonald, A. Tarkowska, C. Sticco, E. Hill, M. Rossner, K.W. Eliceiri, and J.R. Swedlow, "Metadata matters: Access to image data in the real world," *Journal of Cell Biology*, vol. 189, no. 5, pp. 777–782, May 2010.

[19] J. Schindelin, I. Arganda-Carreras, E. Frise, V. Kaynig, M. Longair, T. Pietzsch, S. Preibisch, C. Rueden, S. Saalfeld, B. Schmid, J.-Y. Tinevez, D.J. White, V. Hartenstein, K. Eliceiri, P. Tomancak, and A. Cardona, "Fiji: An open-source platform for biological-image analysis," *Nature Methods*, vol. 9, no. 7, pp. 676–682, July 2012.

[20] A. Edelstein, N. Amodaj, K. Hoover, R. Vale, and N. Stuurman, "Computer control of microscopes using μmanager," in *Current Protocols in Molecular Biology*. John Wiley & Sons, Inc., 2010.