Journal of Biotechnology 261 (2017) 70–75

Contents lists available at ScienceDirect

Journal of Biotechnology

journal homepage: www.elsevier.com/locate/jbiotec

Workflows for microscopy image analysis and cellular phenotyping

Thomas Wollmann[a,*], Holger Erfle[b], Roland Eils[a], Karl Rohr[a], Manuel Gunkel[b,*]

[a] *Dept. Bioinformatics and Functional Genomics, Biomedical Computer Vision Group, University of Heidelberg, BioQuant, IPMB, and DKFZ Heidelberg, Im Neuenheimer Feld 267, 69120 Heidelberg, Germany*

[b] *High-Content Analysis of the Cell (HiCell) and ViroQuant-CellNetworks RNAi Screening Facility, BioQuant, University of Heidelberg, Im Neuenheimer Feld 267, 69120 Heidelberg, Germany*

ARTICLE INFO

Keywords:
Workflow
Pipeline
Image analysis
Microscopy
KNIME
Galaxy
de.NBI

ABSTRACT

In large scale biological experiments, like high-throughput or high-content cellular screening, the amount and the complexity of images to be analyzed are steadily increasing. To handle and process these images, well defined image processing and analysis steps need to be performed by applying dedicated workflows. Multiple software tools have emerged with the aim to facilitate creation of such workflows by integrating existing methods, tools, and routines, and by adapting them to different applications and questions, as well as making them reusable and interchangeable. In this review, we describe workflow systems for the integration of microscopy image analysis techniques with focus on KNIME and Galaxy.

1. Introduction

Since several years, high-throughput (HT) and high-content (HC) microscopy and screening enable many advances in biology. This fast growing technology is transforming cell biology into a big data driven science (Usaj et al., 2016). The generated, huge amount of image data pose many new challenges for data storing, processing, analysis, and interpretation. A main bottleneck is the automated analysis of microscopy images (Schindelin et al., 2012; Swedlow and Eliceiri, 2009; Peng, 2008) and, unfortunately, scaling up workflows is not straight forward. Infrastructure for sharing resources needs to be established, like data storage, computer cluster execution or collaboration functionality with co-workers, which requires large additional efforts. There exists a vast variety of microscopy image analysis software with multiple versions, which often impedes reproducing analysis results (Goodman et al., 2016), or sharing workflows. Hence, there are several reoccurring challenges in large scale microscopy image analysis.

Automated microscopy image analysis aims to retrieve qualitative and quantitative information from the observed objects – usually single cells – and assign specific well defined phenotypes to them (Neumann et al., 2010). For this, dedicated image analysis workflows with corresponding tool pipelines are required (Dinov et al., 2011). There exist several scientific workflow management systems (SWMS) (Achilleos et al., 2012) (e.g., KNIME, Galaxy, Taverna) to ease the creation, provision, and maintenance of data analysis workflows. SWMS allow scaling up the image analysis, enable reproducible science, and collaborative data analysis, and be easily usable for the target user group. A SWMS should be able to effectively use shared computation resources like a high-performance computing (HPC) environment or a cloud. To benefit the scientific community, sharing of software, data, and results should be as easy as possible (Ison et al., 2016). Therefore, central working units as well as tool-, data-, and workflow-sharing platforms have to be developed and respectively incorporated.

In this review, we describe two complementary workflow management systems for large scale microscopy image analysis based on KNIME and Galaxy. Both systems are implemented within the 'German Network for Bioinformatics Infrastructure' (de.NBI). de.NBI is a national infrastructure supported by the Federal Ministry of Education and Research (BMBF) providing bioinformatics services and tools to users in life sciences research and biomedicine. For both workflow management systems, versatile services and tools are offered via de.NBI. We present our experience with the KNIME platform (Berthold et al., 2009) and our recent efforts using the Galaxy platform (Afgan et al., 2016) for microscopy image analysis. Considering not only one platform within de.NBI was motivated by the fact that although most SWMS can generally perform similar tasks, they have a different focus. We will illustrate similarities, differences, and problems of SWMS in the context of large scale microscopy image analysis, based on our two exemplary systems. For example, the focus of the Galaxy platform is a web-based client running in a high-performance computing environment. On the other hand, the focus of the KNIME platform is a desktop client using local resources. However, both systems provide web-based and local clients, and high-performance computing support. Galaxy and KNIME have different mechanisms for workflow sharing, and a direct

[*] Corresponding authors.
E-mail address: thomas.s.wollmann@gmail.com (T. Wollmann).

http://dx.doi.org/10.1016/j.jbiotec.2017.07.019
Received 24 February 2017; Received in revised form 18 July 2017; Accepted 21 July 2017
Available online 27 July 2017

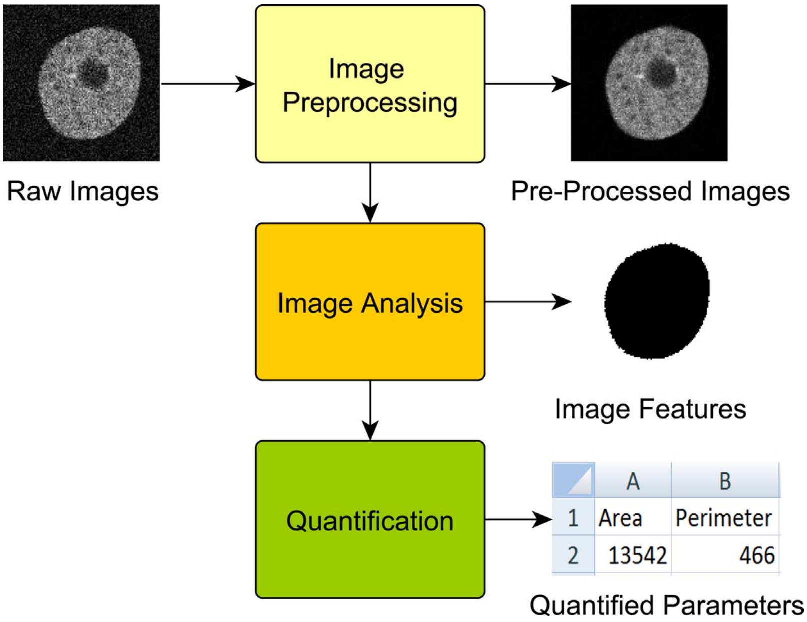

Fig. 1. Schematic workflow for microscopy image analysis.

conversion of workflows between the two systems is currently not possible. However, exchange of Galaxy and KNIME workflows is supported by the project MyExperiment (Goble et al., 2010).

2. Microscopy image analysis workflows

Using a workflow for microscopy image analysis, the acquired images are processed to retrieve quantitative information about a biological experiment. Quantitative information can be determined from the images as a whole or from individual objects within those images (e.g., cells or subcellular structures). The complexity spans multiple levels of image resolution and multiple dimensions depending on the biological question or imaging technique. A wide variety of different microscopy imaging modalities exists. Thus, a large range of readouts can be obtained (e.g., global intensity level, cell count, cell shape, cellular and subcellular constellations, colocalization information).

A schematic workflow for microscopy image analysis is presented in Fig. 1. It consists of a preprocessing step to improve the image quality and enhance meaningful content, an image analysis step including, for example, segmentation, feature extraction, and classification, and a quantification step. The quantified parameters are combined with additional metadata (e.g., position within a plate or well, color channel, time point, or layout information), to generate the readout for a biological experiment.

The following basic steps are usually performed for analyzing individual cells and represent a typical workflow for microscopy image analysis to determine cellular phenotypes.

Preprocessing. Basic image preprocessing steps are performed to improve the image quality, for example, improve the contrast between foreground objects (cells) and background. To remove an image gradient in the background, methods like rolling ball background subtraction can be applied. Noise can be reduced by filtering steps (e.g., using Gaussian and median filters), but at the cost of high frequencies and thus small object details. Images can be normalized and converted to data types suitable for further processing.

Cell segmentation. Image foreground and background are separated based on the intensity values (gray scale distribution) and applying a threshold. This can be performed globally or locally. The images are converted to binary images, where 0 represents the background and 1 the foreground of an image. By determining connected components in the binary images obtained from thresholding, individual cells can be identified and unique IDs are associated with them, in most cases consecutive numbers starting from 1 for each image. Usually, the type of connection of the connected components can be specified, in 2D images typically eight or four adjacent pixels (top, bottom, left, and right) are used. Further, undersegmented cells can be separated using a watershed transformation, convexity measures, or clustering.

Feature extraction. Features of each identified cell are extracted, for example, geometric features (e.g., size, perimeter, circularity) and intensity features (e.g., minimum, maximum, and mean intensity). Based on the extracted features, undesired segmented cells with certain properties out of range (e.g., too dim, too small, too big) can be excluded from further analysis (quality control).

Classification. The resulting identified cells are then classified based on the features, normalization to control experiments should be performed, and specific phenotypes are assigned.

3. Toolboxes for microscopy image analysis

SWMS should integrate established image analysis toolboxes so that they can be used in a standardized way. In most popular programming environments there exist at least one microscopy image analysis toolbox (Eliceiri et al., 2012). Within the Matlab framework, the Matlab Image Processing Toolbox and DIPimage (Hendriks et al., 1999) are widely used. In Python, scikit-image (Van der Walt et al., 2014) and Mahotas (Coelho, 2012) are popular. Most of the Java community on image analysis is using ImageJ (Schneider et al., 2012). Projects like Fiji (Schindelin et al., 2012), Icy (de Chaumont et al., 2013), or CellProfiler (Carpenter et al., 2006) use image analysis components of ImageJ. The programming language C+ + offers OpenCV (Bradski and et, 2000), VTK (Schroeder and Martin, 1996), ITK (Johnson et al., 2015), MITK (Wolf et al., 2004), Ilastik (Sommer et al., 2011), and many more toolboxes. Most of the C+ + toolboxes also offer wrapper for Python, Java, and Matlab. All these image analysis toolboxes share several basic image analysis features, but also have their own unique advantages. In the ideal case, a SWMS would integrate all features of the toolboxes in a meaningful way and therefore achieve interoperability.

Typical components of microscopy image analysis toolboxes are input/output (IO), image processing, image analysis, and rendering. The toolboxes usually embed IO libraries to read and write image files. Every image analysis toolbox focuses on a specific application area and therefore cannot read all common image file types. In addition, they provide custom data structures to optimize image processing and access