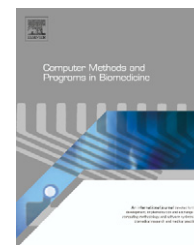




ELSEVIER

journal homepage: www.intl.elsevierhealth.com/journals/cmpb

BlobFinder, a tool for fluorescence microscopy image cytometry

A. Allalou*, C. Wählby

Uppsala University, Centre for Image Analysis, Box 337, SE-751 05 Uppsala, Sweden

ARTICLE INFO

Article history:

Received 23 November 2007

Received in revised form

13 August 2008

Accepted 20 August 2008

Keywords:

Image cytometry

Single cell analysis

FISH

Software

ABSTRACT

Images can be acquired at high rates with modern fluorescence microscopy hardware, giving rise to a demand for high-speed analysis of image data. Digital image cytometry, i.e., automated measurements and extraction of quantitative data from images of cells, provides valuable information for many types of biomedical analysis. There exists a number of different image analysis software packages that can be programmed to perform a wide array of useful measurements. However, the multi-application capability often compromises the simplicity of the tool. Also, the gain in speed of analysis is often compromised by time spent learning complicated software. We provide a free software called BlobFinder that is intended for a limited type of application, making it easy to use, easy to learn and optimized for its particular task. BlobFinder can perform batch processing of image data and quantify as well as localize cells and point like source signals in fluorescence microscopy images, e.g., from FISH, *in situ* PLA and padlock probing, in a fast and easy way.

© 2008 Elsevier Ireland Ltd. All rights reserved.

1. Introduction

In recent years, great progress has been made in the field of fluorescent staining and image acquisition of cells. This has made it possible for scientists to generate vast amounts of images in a short period of time which in turn has led to an increased demand for fast analysis of the image data. Many image analysis software packages can be used for a wide array of applications which often compromises the simplicity and time required for an analysis. In this paper we describe a free, fast and easy-to-use image analysis software for detection and quantification of signals in fluorescence microscopy images of cells.

In biomedical research fluorescent markers are often used to identify sub-cellular structures such as protein complexes, chromosomes, genes and mutations in genes. There are many different methods for *in situ* detection of sub-cellular structures, e.g., immunofluorescence staining, *in situ* Proxim-

ity Ligation Assay (*in situ* PLA) [1], FISH (fluorescent *in situ* hybridization) and padlock-probing [2]. Although the procedures for these methods differ, the images produced share many similarities. The task for analyzing these images typically consists of identifying and counting the signals and the cells. Manual counting of cells and signals is possible when there are few cells, signals and images, but when these parameters increase, the task becomes extremely time consuming and in some cases even impossible. In contrast to manual measurements, image analysis can provide fast and quantitative measurements of these types of image data. On the other hand, automated analysis will naturally introduce some form of bias depending on the limitation of the algorithms. However, these biases are consistent for all analysis in contrast to variations in the human subjectiveness that appears in manual analysis.

A cell culture often contains cells that show different characteristics, e.g., due to different stages of the cell cycle [3]. Cells

* Corresponding author. Tel.: +46 18 471 3469; fax: +46 18 553447.

E-mail addresses: amin@cb.uu.se (A. Allalou), carolina@cb.uu.se (C. Wählby).
0169-2607/\$ – see front matter © 2008 Elsevier Ireland Ltd. All rights reserved.
doi:10.1016/j.cmpb.2008.08.006

in tissue show even greater heterogeneity due to variations in tissue functionality. Important information may be lost if these cell to cell differences are not taken into consideration. Instead of averaging over the image, every cell has to be analyzed by its own, resulting in a more difficult task than only counting signals and cells in an image. In this type of analysis, i.e., single cell analysis, each fluorescent signal has to be assigned to a particular cell; for this to be possible, each cell has to be delineated. For easier detection of the area covered by each cell a cytoplasmic stain can be used as a guide when delineating the cytoplasmic borders [4,5]. However, a cytoplasmic stain may not always provide sufficient information on the position of the border of the cytoplasm. A different approach is to use a membrane stain [6]. Automated cell delineation by use of signals from a membrane stain may work well for cultured cells with non-touching cells. Cells in tissue will however be very difficult to delineate as cell membranes have a complex 3D structure that is not easily handled in thin sections and 2D analysis. In many occasions the red and green color channel is used for molecular detection and the blue channel is used for nuclear staining. This, together with fluorescence spectral overlap, limits the possibility of using a unique color for a cytoplasmic or cell membrane stain. In addition, a cytoplasmic stain may interfere with the molecular stain producing yet another obstacle in the use of a stain to facilitate in the delineation of cytoplasm. Another way to create an approximate outline of each cytoplasm is to assume that the cytoplasm is located within a fixed radius from the nucleus for each cell [7]. This approach uses no cytoplasmic stain and has been proven to give satisfying results when applied to cultured cells [8]. Moreover, this method requires that the image has no or few signals in the background and large numbers of clustered cells can decrease the accuracy of the method.

There is a variety of image analysis software packages that possess the ability to perform single cell analysis using the fixed radius method. ImageJ/NIH Image [9] is an open source image analysis package that can perform a vast variety of biological image analysis. ImageJ is mainly concentrated on analyzing individual images; while macros can be written for ImageJ that can perform batch analysis this would require the user to have knowledge in programming. CellProfiler is a free, open-source system designed for flexible, high-throughput cell image analysis [7]. It has a user-friendly interface and can perform a wide range of cell analysis. Both these software packages are designed for a wide array of different applications and as a result, a lot of functions and adjustable input parameters are available. This can be more confusing than helpful if the user only performs one specific type of analysis. In addition, these software packages are made so that the user must control, and optimize all the image processing steps in the analysis and this can be a disadvantage when the user has little knowledge in image processing or performs the same analysis every many times. The user does not need to know and have control of all the details in the analysis, but should still have an understanding of the basic ideas behind the techniques used. This basic knowledge should help the user to understand why some images do not produce satisfying results and which parameters that can be changed in order to get the most accurate results from the image data.

BlobFinder provides automatic detection and counting of fluorescent signals in microscopy images of cells and can perform two types of analysis, average and single cell analysis. The average analysis counts all signals and nuclei in an image while the single cell analysis counts signals for each individual cell. If z-stack data is available, each z-stack is pre-processed individually, and signals are counted in a projection of the 3D data. Also, batch processing of large data sets is made possible after initial tuning of parameters on a sample image. Results, as well as input parameters, are saved in a text file that can easily be imported to any statistical software. The software comes with a help file, explaining the input parameters and a general description of the methods used in the analysis, a default configuration file and a test image data set aiding the user to get started with the software.

BlobFinder has been developed and optimized for the analysis of images generated by the *in situ* PLA developed by Landegren and coworkers, and commercialized by Olink Biosciences (Uppsala Science Park, Sweden) [10], but can be used for any type of fluorescence microscopy images with point-source signals, such as data from for example FISH. In the process commercializing the *in situ* PLA, and related technologies through Olink Biosciences, a strong need for an analysis tool appeared. The BlobFinder was initially developed in cooperation with Olink Biosciences, and is now their recommended tool for analysis, downloaded approximately 60 times by external users between the launch in June 2007 and May 2008. The software is implemented in MATLAB with a user-friendly GUI (graphical user interface) making it easy to use and get started with [11]. The Matlab code is compiled into a standalone application that can be run without MATLAB installed on the user's computer.

This paper is structured as follows. Section 2 describes the methods and theory of the algorithms used in the analysis. Section 3 introduces the GUI and the parameters that need to be set and tuned. A few sample applications together with results from BlobFinder are provided in Section 4. Section 5 consists of a short conclusion. The Appendix brings up hardware/software requirements and availability of BlobFinder.

2. Computational methods and theory

The main objective of BlobFinder is to detect and count all cells and signals in an image and export a data file containing these quantified results. There are two types of analysis that can be performed, an average and a single cell analysis. The average analysis counts the number of nuclei and signals in an image, resulting in an average signal per cell count. Alternatively, the single cell analysis will assign each signal to the closest cell and get a signal count for each cell in the image.

The first step in the analysis is to segment all cell nuclei in the image. To separate cell nuclei from image background Otsu's method of thresholding, which minimizes the variance of the foreground and background, is used [12]. The nuclei are often clustered and have to be separated in order to identify individual cells. A distance transform is applied to the binary image obtained from the previous step [13]. This will create a landscape like image where the intensity represents the height in the landscape. The watershed transform together

with a h-extended maxima algorithm makes use of the landscape like image to separate clustered nuclei [14,15].

Once the nuclei are separated an area defining the cytoplasm belonging to each cell has to be delineated. Since it often is the case that no cytoplasmic stain is present, cytoplasm delineation is purely based on the distance from the nuclei. A distance transform is performed on the background of the image of the nuclei. A user-defined threshold, representing the maximum distance from cell nucleus to cell border, is thereafter applied, defining the outer border of each cytoplasm. A watershed algorithm is then used to label and separate touching cytoplasm. Nuclei touching the border will be flagged in the results, this way the user can choose to omit these cells in the final results.

When each cell nucleus and cytoplasm is labeled correspondingly the next step is to detect and assign each signal to a particular cell. If there are several z-stacks available these will be pre-processed and thereafter projected into a 2D image by a maximum intensity projection. The pre-processing step consists of a filtration with a variable sized kernel, defined by the user, enhancing local maxima in the image; this is also done in the case when no z-stacks are present. Subsequently, all local maxima in the image are identified and a user defined threshold decides which of the local maxima should be regarded as detected signals. This threshold can be optimized by the user in real time when setting the configuration parameters.

In the last step of the analysis, the detected signals inside each cytoplasm and nucleus are counted. There are two measurements of signals made; a signal count and an intensity measure. The signal count measures all detected signals as

1 while the intensity measure looks at a 5×5 neighborhood around the center of a detected signal and measures the total intensity in that neighborhood. This information is collected and exported to a text file. The content of the output file differs depending on the choice of average or single cell analysis. The main difference is that the single cell analysis output file will show information for each cell (cytoplasm and nucleus) while the average analysis will only show total cell and signal counts. The text file can easily be imported to any type of statistical software for further analysis. The method is summarized in the flowchart in Fig. 1.

3. Program description: GUI and input parameters

BlobFinder is developed for one type of application so that an analysis can be performed using less parameters and less user input. It has been developed in collaboration with the final users of the product in order to make the software package more user friendly and user oriented. Before the development started, the users defined the type of analysis they needed and user-feedback was continuously received during the development. Furthermore, to minimize the number of input parameters in the software, all the parameters were discussed together with the end-users before the start of the project. This way we could optimize the minimum number of input parameters, making the software easier and faster to learn and run, with maximum robustness of the software.

The BlobFinder GUI (Fig. 2) consists mainly of three parts; the image import, the configuration and the analysis. The aim of the configuration part is for the user to easily tune the parameters on one or several test images before running an analysis.

In the configuration setup there are a few parameters that need to be set. First, the user has to make a choice of single cell or average analysis, depending on the type of information that is desired. Second, the user has to set a minimum area in order to remove false nuclei. Third, if the single cell analysis is chosen then a maximum size for the cytoplasm has to be set. Fourth, an approximate blob-size has to be specified in order to use the most appropriate filter for enhancing the signal. If the signal detection is not satisfying the user can change the blob-threshold with a slider bar in real time. After these parameters are set a test run should be performed on one or several images before processing a complete batch.

The output from BlobFinder is a text file that can easily be imported to, for example, Excel (Microsoft Office) for further statistical analysis. If the average analysis is chosen the output will contain information regarding name of the image, number of blobs, intensity of blobs, number of nuclei in each image and area of nuclei. If the single cell analysis is used more information are available in the output file; name of the image, ID for each cell, blobs in cytoplasm, blobs in nucleus, blobs in background, cytoplasm area, nuclei area, intensity measure for signals and a flag for nuclei touching the border.

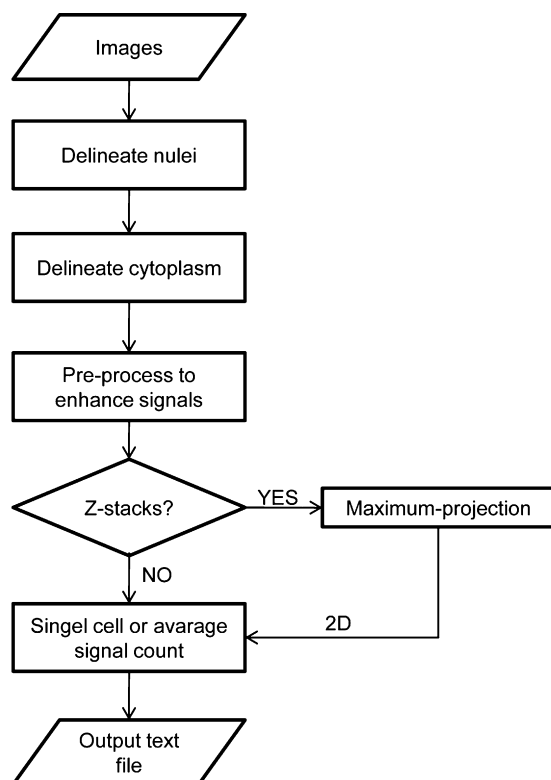


Fig. 1 – Flowchart of analysis method.

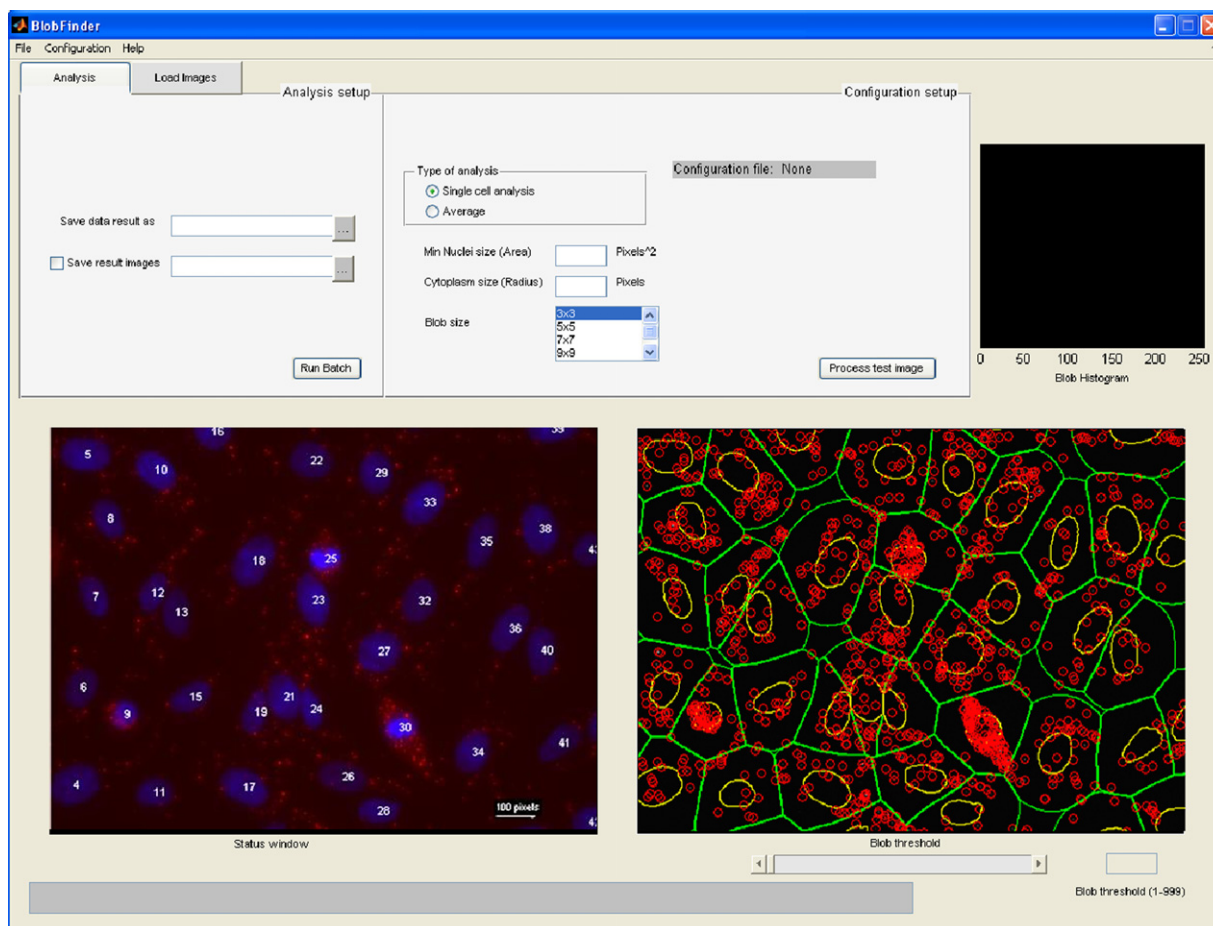


Fig. 2 – Analysis tab of GUI window in BlobFinder. Analysis results are from images provided by Olink Biosciences (Uppsala Science Park, Sweden).

4. Sample applications

In this section, three different analysis are performed. First, an average analysis is described to illustrate the simplest type of analysis. Second, a single cell analysis is described to illustrate how to quantify differences among cells within the same image. The last analysis is a comparison between using the

count and intensity measurement of signals in a simulated image.

4.1. Cell average analysis

The average analysis gives the least information, but often this can be enough, e.g., for images containing cells with approx-

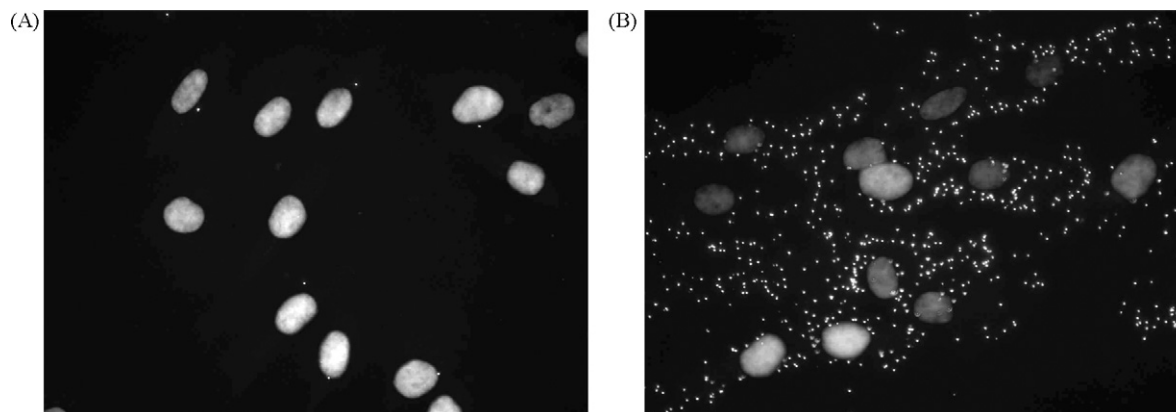


Fig. 3 – A: Cell culture exposed to 0.1 ng/ml of PDGF-BB. B: Cell culture exposed to 100 ng of PDGF-BB.

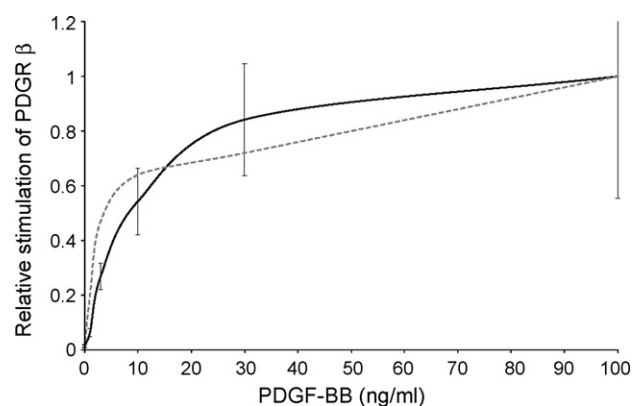


Fig. 4 – Signal per nucleus vs. PDGF-BB concentration. Solid line: Normalized results from BlobFinder. Error-bars represent the standard deviation between images in the analysis. Dashed line: Normalized results obtained with immunoblotting.

imately equal distribution of signals. BlobFinder was used to batch process 60 images of cultured cells exposed to different concentrations of platelet-derived growth factor BB (PDGF-BB). PDGF-BB stimulates the phosphorylation of platelet-derived growth factor receptor β (PDGFR β). PDGFR β phosphorylation was detected by *in situ* PLA, resulting in bright point like signals [1]. Ten images were acquired for each concentration 0, 1, 3, 10, 30 and 100 ng/ml of PDGF-BB (see Fig. 3).

Data from BlobFinder illustrate a significant increase in signals per nucleus as the PDGF-BB concentration increases, which correlates well with results obtained by immunoblotting (Fig. 4).

4.2. Single cell analysis

For the next image set a single cell analysis was required as the signal distribution differed between cells in the

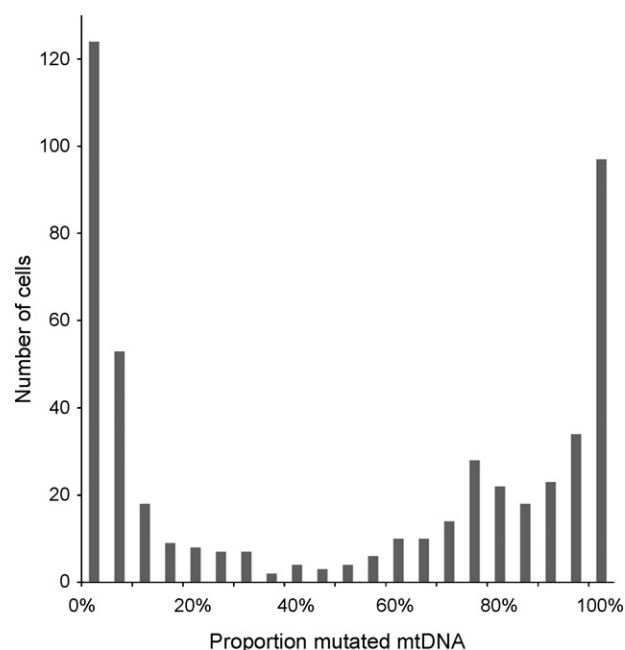


Fig. 6 – Number of cells vs. proportion of mtDNA mutation per cell.

same image. Two different signals; green and red, representing non-mutated and mutated mitochondrial DNA (mtDNA), detected with padlock probes were counted [16]. The objective was to determine the proportion of non-mutated to mutated mtDNA in each cell, quantifying the amount of mutation that has occurred. Since there were two color channels containing signals the analysis had to be run twice; once for each type of signal. Fifty-nine images with a total of 522 cells from a co-culture containing cells with approximately 0% or 100% mtDNA mutation (Fig. 5) were analyzed in batch using BlobFinder.

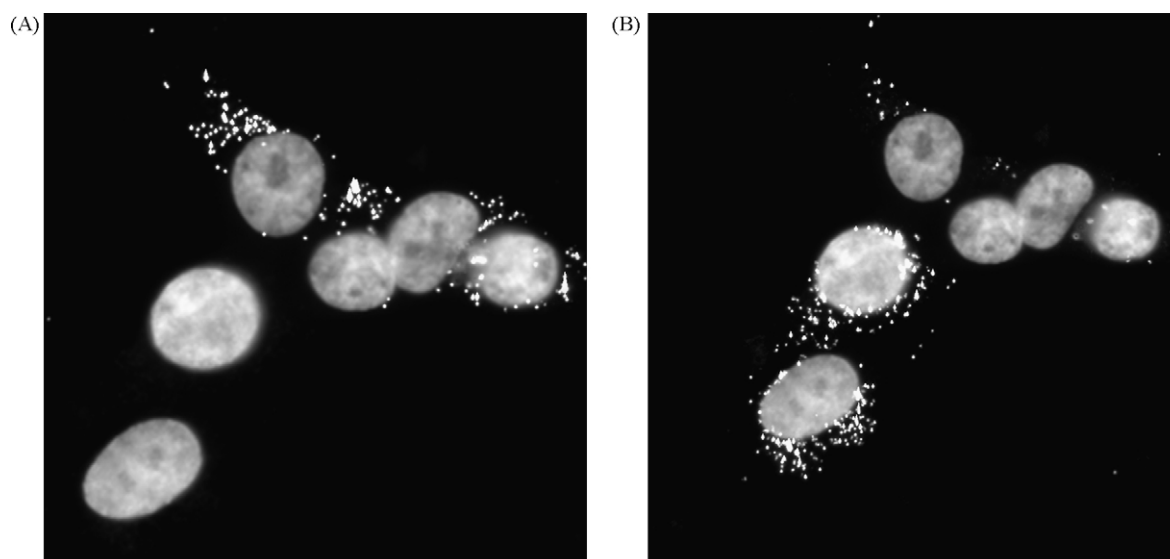


Fig. 5 – Data set containing images of a co-culture; cells should be either 100% mutant or 100% non-mutant (wild type). A: Image channels R and B; red signals show mutated mtDNA. B: Image channels G and B; green signals show non-mutated mtDNA.

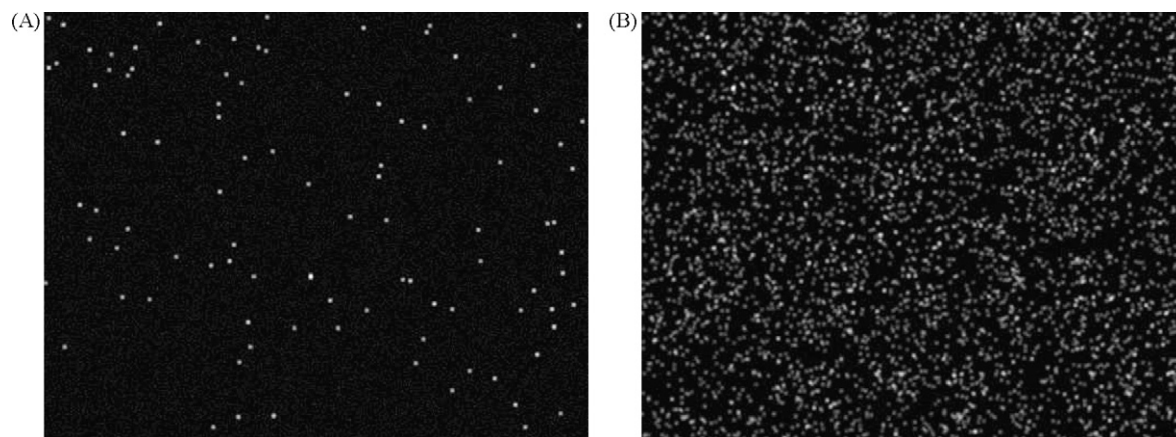


Fig. 7 – A: Artificial image containing 100 signals per image (500 pixels \times 400 pixels). B: Artificial image containing 4000 signals per image (500 pixels \times 400 pixels).

The graph in Fig. 6 shows the number of cells that have 0% through 100% mutated mtDNA. The data from the co-culture shows, as expected, distinct distributions of cells at the extremes, i.e., cells with 100% and 0% mutated mtDNA. The BlobFinder analysis confirms that the image data consist of a co-culture as there are only a few cells with intermediate levels of mutated mtDNA.

4.3. Signal count vs. total signal intensity

BlobFinder provides two ways of quantifying the amount of signals; signal count and signal intensity measure. Artificial data was produced in order to compare the two different methods (Fig. 7). The artificial data were made by random distribution of point source signals in an image. These signals were then filtered with a Gaussian filter to make the point source signals look like real signals blurred by a point spread

function. Also, Gaussian noise was added to make the artificial image more realistic. To get the effect of two signals on top of each other and also produce images with varying amounts of signals, layers with random signals were produced and then summed into images with varying signal density counts. The artificial images were 500 pixels \times 400 pixels in size and contained a range of 100 to 4000 signals per image. Ten images for each concentration of signals were created, resulting in a total of 400 images. All signals were counted, i.e., the image was treated as one cell, in order to compare with the true value from the simulation.

The graphs in Fig. 8 show how the two methods of signal quantification compare with the true number of signals. Up to approximately 1000 signals per image both methods give similar and reasonably accurate results. In contrast, when reaching a high signal concentration the intensity measure gives better results. While both methods miss signals at extremely high

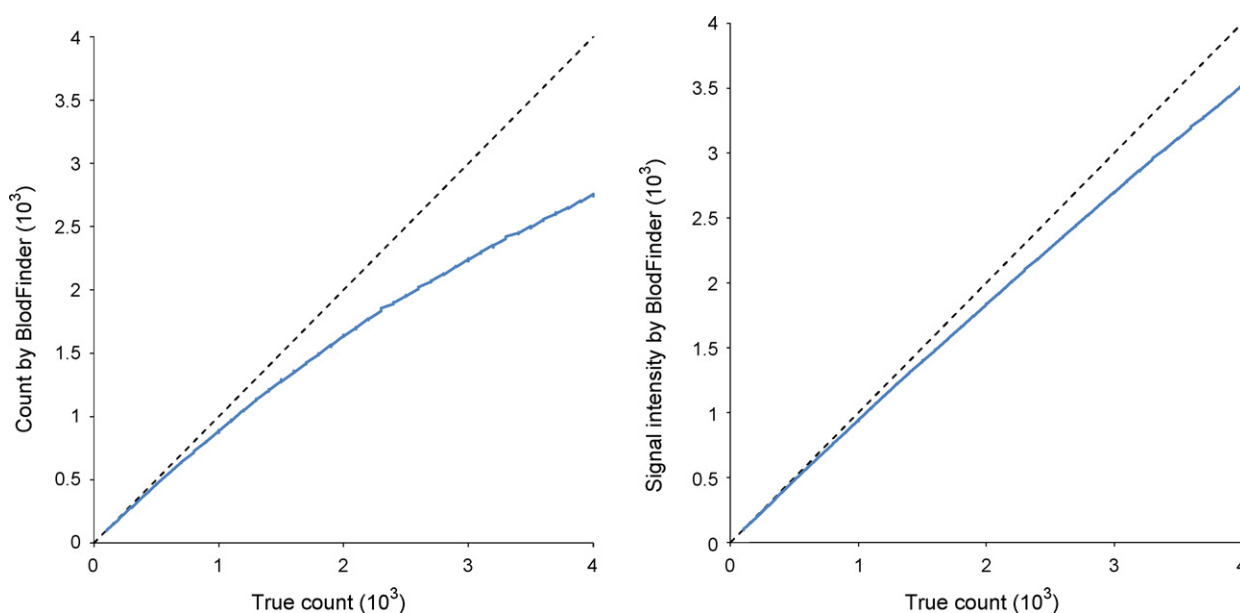


Fig. 8 – Left. Solid line: signal count by BlobFinder. Dashed line: ideal signal detection ($y = x$). Right. Solid line: signal intensity measure by BlobFinder. Dashed line: ideal signal detection ($y = x$).

concentrations the intensity measure produce results with a smaller error. One can conclude that when working with images with a high concentration of signals it is better to use the intensity measure. On the other hand, when working with low concentrations of signals, it is more suitable to use the signal count data, as it is easier to interpret data that gives information on number of signals present in the cell. A general rule when deciding method to be used for signal detection is; if there are many clustered signals use the intensity measure and if the majority of the signals are well separated use the signal count.

5. Conclusion

Fluorescence microscopy images of cells with a point-source signals are a commonly used when studying at genes, gene-expression and proteins. BlobFinder, described in this paper, is a free image analysis software package intended for this type of applications. The focus has been on making a tool that produces good results with minimal input on a limited type of images, in this case fluorescent images of cells with pointsource signals, keeping the software simple and fast to use and learn. It should be enough for the user to have a basic understanding for the techniques used in the software to get started with the analysis. BlobFinder comes with a help file that gives the user a general introduction to the methods and algorithms used in the analysis. With this basic knowledge of the methods in BlobFinder, the user should be able to understand the effect of each of the parameters and also have the ability to identify why and where in the analysis there is a problem when some image data are not giving satisfying results.

Depending on the information desired from the image data, BlobFinder performs two types of analysis; cell-specific analysis that gives information on each single cell, providing an opportunity to analyze differences between cells in the same image, and non-cell-specific analysis that gives a total number signals and cells in the image. In addition, there are two different types of signal measurements providing the user with more accurate and easier interpretable results depending on the density of signals in the image. In the user friendly GUI there are only a few parameters that need to be set before running the analysis on a whole image set.

BlobFinder is a first version and improvements are planned for future versions, e.g., better separation of big cell clusters. In addition, the signal detection works well but more powerful algorithms are considered, especially in cases when there is a high signal concentration.

Acknowledgements

This project was funded by the EU-Strep project ENLIGHT (enhanced ligase-based histochemical techniques) and Olink Bioscience, Uppsala, Sweden. The authors would also like to thank Irene Weibrecht, Malin Jarvius and Ola Söderberg at the Department of Genetics and Pathology, Uppsala University, Sweden; Frans M. van de Rijke at the Department of Molecular Cell Biology, Leiden University Medical Center, the Nether-

lands; and Olink Biosciences, Uppsala Science Park, Sweden, for providing image data.

Appendix A

A.1. Hardware and software specifications

BlobFinder was developed with Matlab for Windows XP operating systems. The Matlab-code is compiled into an executable standalone application that can be run on a computer with or without Matlab installed. If the user does not have Matlab ver. R2006b installed on the computer an installation of the MCR (Matlab component runtime) libraries, provided by The MathWorks, Inc., is needed. BlobFinder as well as the MCR installer can be downloaded from www.cb.uu.se/~amin/BlobFinder. The hardware requirements for BlobFinder are minimal as it can be run on most modern PCs. The example analysis was performed on an Intel Xeon 3.00 GHz with 2.00 GB of RAM.

A.2. Availability of the software

A version of BlobFinder and MCR installer can be downloaded from our website www.cb.uu.se/~amin/BlobFinder.

REFERENCES

- [1] M. Jarvius, J. Paulsson, I. Weibrecht, K. Leuchowius, A. Andersson, C. Wählby, M. Gullberg, J. Botling, T. Sjöblom, B. Markova, A. Ostman, U. Landegren, O. Söderberg, In situ detection of phosphorylated platelet-derived growth factor receptor beta using a generalized proximity ligation method, *Mol. Cell Proteomics* 6 (9) (2007) 1500–1509.
- [2] C. Larsson, J. Koch, A. Nygren, G. Janssen, A. Raap, U. Landegren, M. Nilsson, In situ genotyping individual DNA molecules by target-primed rolling-circle amplification of padlock probes, *Nat. Methods* 1 (2004) 227–232.
- [3] J. Levsky, R. Singer, Gene expression and the myth of the average cell, *Trends Cell Biol.* 13 (2003) 4–6.
- [4] J. Lindblad, C. Wählby, E. Bengtsson, A. Zaltsman, Image analysis for automatic segmentation of cytoplasm and classification of Rac1 activation, *Cytometry* 57 (1) (2004) 22–33.
- [5] C. Wählby, J. Lindblad, M. Vondrus, E. Bengtsson, L. Björkstén, Algorithms for cytoplasm segmentation of fluorescence labelled cells, *Anal. Cell. Pathol.* 24 (2002) 101–111.
- [6] C. Ortiz de Solorzano, R. Malladi, S. Lelievre, S. Lockett, Segmentation of nuclei and cells using membrane related protein markers, *J. Microsc.* 201 (3) (2001) 404–415.
- [7] A. Carpenter, T. Jones, M. Lamprecht, C. Clarke, I. Kang, O. Friman, D. Guertin, J. Chang, R. Lindquist, J. Moffat, P. Golland, D. Sabatini, Cellprofiler: image analysis software for identifying and quantifying cell phenotypes, *Genome Biol.* 7 (10) (2006) R100.
- [8] A. Allalou, F. van de Rijke, R. Tafrechi, A. Raap, C. Wählby, *Image Analysis*, Springer, Berlin/Heidelberg, 2007, 631–640.
- [9] M. Abramoff, P. Magalhaes, S. Ram, Image processing with Image, *J. Biophotonics Int.* 11 (7) (2004) 36–42.
- [10] O. Söderberg, M. Gullberg, M. Jarvius, K. Ridderstråle, K. Leuchowius, J. Jarvius, K. Wester, P. Hydring, F. Bahram, L.

- Larsson, U. Landegren, Direct observation of individual endogenous protein complexes in situ by proximity ligation, *Nat. Methods* 3 (12) (2006) 995–1000.
- [11] M. version 7.3.0, The MathWorks, Inc., Natick, MA, 2003.
- [12] N. Otsu, A threshold selection method from gray-level histograms, *IEEE Trans. Syst. Man Cybernet.* 9 (1) (1979) 62–69.
- [13] R.C. Gonzales, R.E. Woods, S.L. Eddins, *Digital Image Processing using MATLAB*, Pearson Prentice Hall, 2004.
- [14] C. Lantuéjoul, S. Beucher, On the use of geodesic metric in image analysis, *J. Microsc.* 121 (1981) 39–49.
- [15] P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag, 1999.
- [16] R. Tafrechi, F. van de Rijke, A. Allallou, C. Larsson, W. Sloos, M. van de Sande, C. Wählby, G. Janssen, A. Raap, Single cell A3243G mitochondrial DNA mutation load assays for segregation analysis, *J. Histochem. Cytochem.* 55 (2007) 1159–1166.