

# Comparison of two automatic cell-counting solutions for fluorescent microscopic images

J. LOJK\*, U. ČIBEJ†, D. KARLAŠ†, L. ŠAJN† & M. PAVLIN\*

\*Faculty of Electrical Engineering, University of Ljubljana, Ljubljana, Slovenia

†Faculty of Computer and Information Science, University of Ljubljana, Ljubljana, Slovenia

**Key words.** Automatic cell counting, CELLCOUNTER, cell viability, evolutionary algorithm, fluorescence microscopy, LEARN123, transfection.

## Summary

Cell counting in microscopic images is one of the fundamental analysis tools in life sciences, but is usually tedious, time consuming and prone to human error. Several programs for automatic cell counting have been developed so far, but most of them demand additional training or data input from the user. Most of them do not allow the users to online monitor the counting results, either. Therefore, we designed two straightforward, simple-to-use cell-counting programs that also allow users to correct the detection results. In this paper, we present the CELLCOUNTER and LEARN123 programs for automatic and semiautomatic counting of objects in fluorescent microscopic images (cells or cell nuclei) with a user-friendly interface. Although CELLCOUNTER is based on predefined and fine-tuned set of filters optimized on sets of chosen experiments, LEARN123 uses an evolutionary algorithm to determine the adapt filter parameters based on a learning set of images. CELLCOUNTER also includes an extension for analysis of overlaying images. The efficiency of both programs was assessed on images of cells stained with different fluorescent dyes by comparing automatically obtained results with results that were manually annotated by an expert. With both programs, the correlation between automatic and manual counting was very high ( $R^2 < 0.9$ ), although CELLCOUNTER had some difficulties processing images with no cells or weakly stained cells, where sometimes the background noise was recognized as an object of interest. Nevertheless, the differences between manual and automatic counting were small compared to variations between experimental repeats. Both programs significantly

reduced the time required to process the acquired images from hours to minutes. The programs enable consistent, robust, fast and accurate detection of fluorescent objects and can therefore be applied to a range of different applications in different fields of life sciences where fluorescent labelling is used for quantification of various phenomena. Moreover, CELLCOUNTER overlay extension also enables fast analysis of related images that would otherwise require image merging for accurate analysis, whereas LEARN123's evolutionary algorithm can adapt counting parameters to specific sets of images of different experimental settings.

## Introduction

Quantification is one of the most fundamental analysis tools in biology, medicine and other life sciences (Peer & Corzo, 2007). Several methods exist for quantification of different phenomena, and a substantial part of them use fluorescent dyes in combination with various protocols where the number of fluorescent cells and fluorescence intensity are usually indicators of a specific effect. Fluorescently stained cells can be detected and analysed using flow cytometry (Rieseberg *et al.*, 2001), spectrofluorometry (Lakowicz, 2006) and fluorescent microscopy (Stephens & Allan, 2003). Also other counting methods exist that are not based on fluorescence labelling, like counting with counting chambers (hemocytometer) (Dein *et al.*, 1994; Ongena *et al.*, 2010), but those can be time consuming and also require detachment of the cells, which can result in their loss and damage. Moreover, analysis of plated cells enables direct observation of cell morphology and can sometimes give more accurate counting results (Marjanovič *et al.*, 2014). Despite that, counting is considered tedious, time-consuming, subjected to user-to-user variations and is therefore, whenever possible, avoided in favour of other analysis methods, like flow cytometry or spectrofluorometry.

The idea of automatic counting of various objects exists for almost as long as personal computers do (Dell, 1954; Sharpe *et al.*, 1986), but with time the solutions significantly

J. Lojk and U. Čibej have contributed equally to this work.

Correspondence to: Mojca Pavlin, Group for nano and biotechnological applications, Faculty of Electrical Engineering, University of Ljubljana, Trzaska 25, 1000 Ljubljana, Slovenia. Tel: +386 1 4768 949; fax: +386 1 4264 850; e-mail: mojca.pavlin@fe.uni-lj.si

Luka Šajn, Computer Vision Laboratory, Faculty of Computer and Information Science, University of Ljubljana, Večna pot 113, 1000 Ljubljana, Slovenia. Tel: +386 1 479 8220; fax: +386 1 476 84 98; e-mail: luka.sajn@fri.uni-lj.si

improved. Currently, several programs are available for analysis of microscopic images and counting of adherent cells, both commercial solutions, such as METAMORPH (Molecula Devices, Downingtown, PA, USA), BioQuant (Image Analysis Corporation, Nashville, TN, USA), Image-Pro (Media Cybernetics, Bethesda, MD, USA) and Cellscreeener system (Innovatis AG, Bielefeld, Germany), as well as free software, such as ImageJ (US National Institute of Health, Bethesda, MD, USA) Plugins (Gallagher, 2008) or customized ImageJ macros (Ganias *et al.*, 2010; Cai *et al.*, 2011; Väyrynen *et al.*, 2012), CellProfiler (Carpenter *et al.*, 2006), UTHSCSA ImageTool (University of Texas Health Science Center, San Antonio, TX, USA) and CellC (Selinummi *et al.*, 2005). In addition, several custom programs have been written that are optimized for analysis of a specific image analysis problem (Byun *et al.*, 2006; Faustino *et al.*, 2009; Kachouie *et al.*, 2009). In practice, however, most of the solutions require additional user input regarding various image parameters, such as thresholds, approximate nuclei distances, contrast settings, etc., which have to be precisely tuned for reliable results. These parameters require specific knowledge of image processing, which is not common among life-science experts (Benali *et al.*, 2003; Byun *et al.*, 2006).

However, a complete automatization of cell counting process also faces several technical challenges. The most common problems are (1) a wide variety of assays and cell types, resulting in a range of objects that are hard to generalize, and (2) uneven illumination and other equipment-related factors, such as electronic and/or optical noise, which cause variable contrast and image quality (Haralick & Shapiro, 1985; Wu *et al.*, 1995). Moreover, (3) cells in the cell culture can have a varying density and can even overlap (Li *et al.*, 2006), and (4) extracellular debris and internal cell structures can interfere with the recognition algorithm, giving false results (Wu *et al.*, 1995). Depending on the microscopy technique used, (5) images can have a really low contrast and it can be (6) difficult to detect individual nuclei in multinucleated cells.

In this paper, we present two programs (CELLCOUNTER and LEARN123) for effective cell counting which do not require any specific knowledge of image processing and are incorporated in user-friendly computer applications. The programs were specifically designed for counting fluorescently stained cells, which are used in many biomolecular and biotechnological applications – from cell viability assays (Bregar *et al.*, 2013) to determination of transfection efficiency (Kanduser *et al.*, 2009; Marjanovič *et al.*, 2014; Mars *et al.*, 2014). Both programs are based on the same filter sets, but whereas CELLCOUNTER operates with highly optimized hardcoded parameters, LEARN123's evolutionary algorithm can adapt the counting parameters to a chosen set of images and can thus account for a higher variability among the images of different experimental settings. Moreover, a special extension of the CELLCOUNTER program also enables easy counting on two overlapping images. Both programs are already intensively used and as shown in the

empirical evaluation, have drastically improved researchers' workflow.

## Materials and methods

### *In vitro experiments*

**Cell culturing.** Images used for program validation were taken from experiments performed on Chinese hamster ovary cells and on primary human myoblasts grown in Ham's tissue culture medium for mammalian cells (HAM) (Sigma-Aldrich GmbH, Diesenhofen, Germany) and Advanced MEM (aMEM) (Sigma-Aldrich), respectively, both supplemented with 10% fetal bovine serum (Sigma-Aldrich) at 37°C in 5% CO<sub>2</sub>-enriched atmosphere at saturation humidity. All experiments were performed on cell cultures in exponential growth phase.

**Nanoparticles (NPs) and cell viability (type 1 experiment).** Images used for program validation were taken from nanoparticle (NP) exposure experiments and are a part of not yet published results. The cell viability experiment for different NP concentrations was performed on Chinese hamster ovary cells as described previously (Bregar *et al.*, 2013). Briefly, cells were incubated with an increasing concentration of polycationic polymer coated magnetic NPs for 24 h and stained with two fluorescent dyes: Hoechst 33342 (Hoechst), which stained all cell nuclei, and propidium iodide (PI), which differentially stained only dead cells (Jones & Senft, 1985). At least 15 visual fields were taken at 200 × magnification for each sample and each fluorescent dye using a fluorescent microscope (Zeiss 200, Axiovert, Germany). The images were recorded by METAMORPH imaging system software (Visitron, Germany) and saved in lossless TIF (Tagged Image File) format.

The number of live cells in each sample was obtained by subtracting the number of dead cells ( $N_{(PI)}$ ) from the number of all counted cells ( $N_{(H)}$ ). Cell viability for each sample (% Viability) was determined as a ratio of live cells in each sample ( $N_{(H)} - N_{(PI)}$ ) to all cells in the control sample ( $N_{(CH)}$ ):

$$\% \text{Viability} = \left[ \frac{N_{(H)} - N_{(PI)}}{N_{(CH)}} \right] \times 100. \quad (1)$$

### *Electroporation and transfection efficiency (type 2 experiment).*

The images used for program evaluation were taken from electroporation experiments published elsewhere (Mars *et al.*, 2014). Electrotransfection of cells was performed according to previously described protocol (Kanduser *et al.*, 2009; Pavlin *et al.*, 2010; Mars *et al.*, 2014). Briefly, cells were electroporated in electroporation medium containing pEGFP-N1 plasmid DNA (Clontech, Clontech Laboratories, Mountain View, CA, USA), carrying the gene for green fluorescent protein (GFP). Electric pulses were applied with high-voltage generator (Jouan, France) using wire electrodes. Protocols of eight

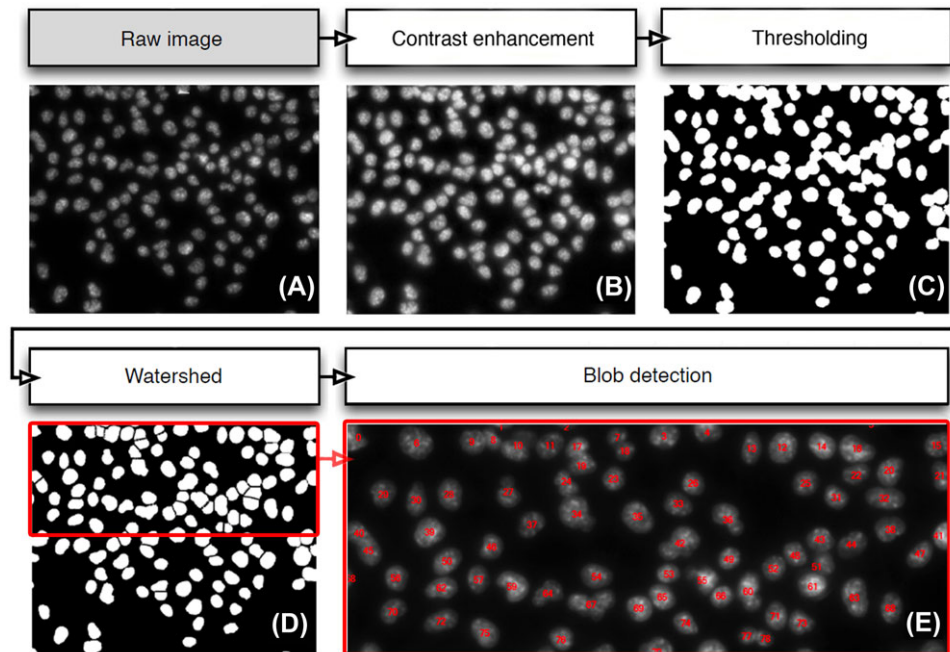


Fig. 1. A schematic overview of the CELLCOUNTER's counting steps with a sequence of sample images.

pulses with a duration of 2 or 5 ms with a repetition frequency 1 Hz were used, applying two amplitudes of electric field ( $E$ ):  $E = 0.8 \text{ kV cm}^{-1}$  and  $E = 0.9 \text{ kV cm}^{-1}$ . Transfection efficiency was determined 24 h after electroporation. Cell nuclei were stained with Hoechst to obtain the total cell number that was compared to the number of GFP expressing cells. At least six visual fields at  $200\times$  magnification were taken of each sample for each used fluorescent dye. The images were recorded by METAMORPH imaging system software and saved in TIF format.

Transfection efficiency of each sample (%Transfection) was determined as the ratio between the number of GFP expressing cells ( $N_{GFP}$ ) and the number of all counted nuclei ( $N$ ) (Pavlin *et al.*, 2012; Mars *et al.*, 2014) in a sample as determined by Hoechst staining:

$$\% \text{Transfection} = 100 * N_{GFP} / N. \quad (2)$$

#### Description of the programs

**CELLCOUNTER: simple out of the box solution.** CELLCOUNTER was written in C# and uses an open source C# framework designed for developers and researchers in the fields of Computer Vision and Artificial Intelligence (AForge.NET). The program is available at <http://lalg.fri.uni-lj.si/CellCounter>. The process used in this program is a four-step image processing algorithm. The sequence of these steps and underlying algorithms were determined by testing several possible combinations and the most accurate detection and counting procedure was chosen based on analysed sets of images.

Step 1 (Fig. 1B): The program first addresses the problem of image contrast, which can be low or unevenly distributed through the image due to uneven illumination or other optical problems. Image contrast is thus improved using the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm (Zuiderveld, 1994; Cheng & Shi, 2004; Ahmad, 2011). In our application, the image is cut into 64 ( $8 \times 8$ ) disjoint regions and histogram equalization is independently applied to each region: within each image region, local maximum and minimum pixel intensity are determined and the intensities are proportionally adjusted to the scale ranging from 0 to 255. The individual regions are then recomposed again using interpolation at region borders. The final image has better contrast and can reveal more local information (Fig. 1B).

Step 2 (Fig. 1C): Grey-scale images are converted to black-and-white. The objects' borders are defined using a thresholding algorithm. In this step, Otsu Thresholding algorithm (Otsu, 1979; Xu *et al.*, 2011) was used, as it gave slightly better results compared to other tested algorithms (Huang, Renyi entropy method and others).

Step 3 (Fig. 1D): In cell cultures, cells tend to grow in close proximity, touch or even overlap. To split the overlapping (touching) objects, the Watershed algorithm (Vincent & Soille, 1991) was applied. A classical method for finding the seeds is used, that is, the seeds are the local maximum points of the distance function. The distance function of an object pixel is the minimal distance to a background pixel. The image is eroded on the watershed lines, which usually correspond to the borders between cells. This enables

detection of individual cells, even when they are multinucleated or overlapping.

Step 4 (Fig. 1E): The connected regions are counted using classical blob detection algorithm (Chang *et al.*, 2004), which counts the objects separated by black background.

CELLCOUNTER program was also extended with an add-on for cell counting on overlaid images. The program aligns the two images of the same visual field but of different fluorescent labels to detect the overlapping regions. On the first image (in our case Hoechst stained cell nuclei), the program detects and counts fluorescent objects using the algorithm steps described previously. The second image (GFP labelled cells) is thresholded with a hard-coded value. The objects' positions from the first image are overlaid on the second image and the program detects whether the objects are positioned on the foreground or background of the second, thresholded image, and recognizes them as positive or negative, respectively.

**LEARN123: automatic adaptation of the counter.** Based on the sequences of filters with hardcoded fine-tuned parameters used in CELLCOUNTER, we created LEARN123 program (using programming language Scala), which is able to adapt the filter parameters to specific variations in the images with (as little as possible) guidance from the user. We used the CELLCOUNTER filter sequence as a general framework, where the parameters of the image filters represent a vector, which needs to be optimized. As the optimization method, we chose an evolutionary approach (Jong & Jong, 2002) because it is very flexible and we can easily change the goal function, stop the process at any point and get feasible solutions, start from existing solutions and improve the solution by interacting with the user. Thus, it enables adaptation to the specific set of fluorescent images. Evolutionary algorithms have been used before for finding good counters for a specific cell counting application (Chan *et al.*, 2010; Ramin *et al.*, 2012). Our application strives to be more general, we provide the user with tools that finds good counters for many different applications.

The application is subdivided into two subapplications; the learning part and the counting part of the application. The program is available online at <http://lalg.fri.uni-lj.si/learn123>.

Step 1 (Fig. 2): The user selects a representative subset of microscopic images (typically around 10 images) that spawns as many different aspects of the entire set (i.e. different sizes of object, different lighting conditions, different shapes, etc.).

Step 2: The user manually counts the objects in the images, which provides the number and approximate centres of the objects. By selecting objects of certain intensities, the user implicitly determines the threshold for fluorescence intensity, which is usually user specific.

Step 3: The user initiates the learning process. The objective of the learning process is to find a counter (specified by a numeric vector) that minimizes the error made by the counter on the learning set of images. This is achieved through a

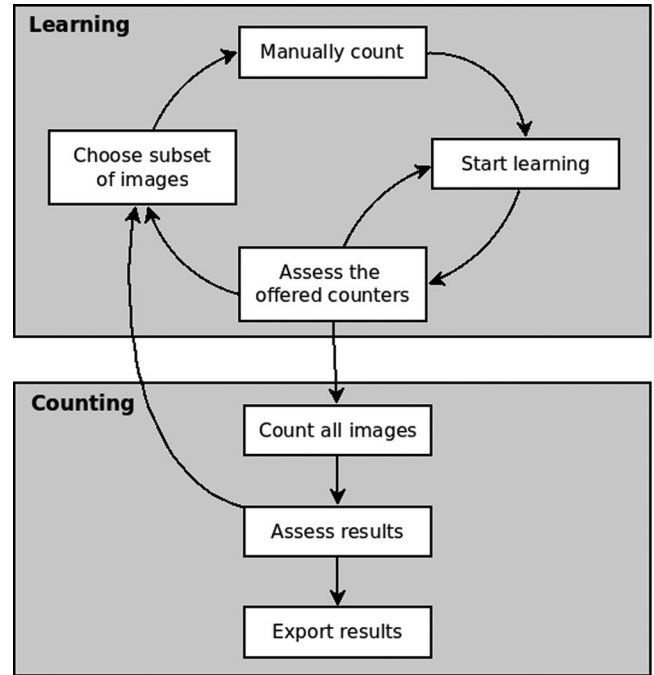


Fig. 2. The user workflow in the LEARN123 program.

standard evolutionary algorithm with some specific adaptation to fit our problem.

To generate a set of most accurate counters, the algorithm uses two intuitive criteria: (1) the difference of the obtained counts (the numbers) should be as small as possible, and (2) the positions of the found objects should be as close to the manual annotations as possible. The goal function for the first criterion is simply the absolute difference between the manual count and the automatic count, defined as:

$$\Delta_{img}(C_{man}(img), C_P(img)) = ||C_{man}(img) - |C_P(img)||, \quad (3)$$

where  $C_{man}$  is the set of manually annotated points on the image  $img$ , and  $C_P$  is the set of points found on the same image by the automatic counter defined by the parameter vector  $P$ .

The second criterion is encompassed in the function which finds the distances between points found by the manual count and points found by the automatic counter:

$$D_{img}(C_{man}(img), C_P(img)) = \sum_{p \in C_{man}(img)} d(p, \text{closest}(p, C_P(img))), \quad (4)$$

where  $d$  is the Euclidean distance between the two points, and  $\text{closest}(p, S)$  returns the closest point in the set  $S$  to the point  $p$ . The cumulative (or average) value of  $\Delta_{img}(D_{img})$  over all images is the quantification of the quality of a given automatic counter. These two goal functions are used interchangeably

during the execution of the evolutionary algorithm; in every generation, the goal function is switched.

Counters, which are used for creating a new generation, are selected based on their rank in the current population. We used a one-point crossover, that is, a random position in the chromosome is chosen and the two chromosomes exchange the parameters to the left of that position. The mutation is not a bit-level mutation (as it is usually the case with genetic algorithms), but we used a mutation on the parameter level, where a uniformly random value is chosen from the parameter domain.

Step 4: When the program finishes its learning process, the set of 10 best counters is presented to the user. The user can visually inspect the counts of the automatic counter and compare them to the manual counts. If none of the counters shows the desired counting accuracy, the user can add new images to the learn set, remove some old images and restart the learning process from the previous learn run.

Step 5: When a suitable counter is found by the above process, it can be used in the second subapplication, where all experimental images are counted. If the algorithm behaves badly on a certain image, the image can be added to the learn set and the new learning process will adapt also to this type of images.

Although in both presented programs the counting process is robust, as we will show in the following sections, misdetections can still occur on more problematic images, such as those with low contrast, heterogeneous background or images with overlaying objects, which are not split by the Watershed algorithm. The programs thus enable the user to remove erroneously detected counting hits or add missed hits with a mouse click. Left mouse click adds a counting hit at the clicked spot whereas upon the right mouse click the program searches for the closest hit and removes it. This will be referred as a semiautomatic counting hereafter.

### Program evaluation

The presented programs were developed and optimized based on microscopic images from several types of experiments and experimental repeats. To demonstrate and evaluate both programs' efficiency, two random experiments were selected.

Cells in the images of the selected experiments were counted using only `CELLCOUNTER` or `LEARN123` and by manually counting using ImageJ software (manual) (v1.45s, National Institute of Health). ImageJ was selected as a reference program because it is openly accessible and widely used by several research groups.

To evaluate the accuracy of automatic counting, three independent NP cell viability experiments were counted manually and automatically by the same user. Moreover, images from one NP viability experiment were independently analysed by three researchers to determine the inter-personal error. Results are expressed as mean  $\pm$  standard error (SE). Images

from the electroporation experiment were analysed only with `CELLCOUNTER` by one user.

### Results and discussion

We have developed two programs (`CELLCOUNTER` and `LEARN123`) for automatic and semiautomatic counting of fluorescently stained objects, usually cells and cell nuclei. Although there are several existing programs for automatic microscopic image analysis, we wanted a simple-to-use cell and nuclei counting method that would not require additional training of the users. Also, most of the existing programs do not allow corrections of the counting results by the user. The `CELLCOUNTER` program uses a hardcoded set of filters with fixed parameters, whereas `LEARN123` uses an evolutionary algorithm that adapts the parameters of the same general filter framework to each individual set of images. This enables the program to better adapt the counting parameters to specific sets of images obtained with different experimental protocols (labels, concentrations, filters, cells...), equipment and possible user-induced variations.

Fluorescence microscopy was chosen as a basis for cell counting because the obtained images have a much higher contrast compared to bright field of phase-contrast microscopy, while retaining the resolution. For specific cases where the total number of cells has to be determined, the fluorescent staining of cell nuclei also mitigates the problems related to detection of cell edges, shape or cell confluency (unclear cell borders between cells in close contact) (Haralick & Shapiro, 1985; Wu *et al.*, 1995). Moreover, the presented solution enables detection of individual nuclei in multinucleated cells, which is difficult to determine on bright field and phase-contrast images (Glory *et al.*, 2004). Although fluorescence microscopy requires some additional experimental work (cell staining) and is only applicable to short-term experiments, it avoids more complex cell counting solutions associated with other microscopic techniques (Usaj *et al.*, 2011). However, if nontoxic cell trackers are used for cell labelling, cells can be analysed over extended time periods (Urish *et al.*, 2013).

As an example to evaluate the program, we show two specific applications (determination of cell viability and transfection efficiency), in which cells were labelled with three fluorescent labels: blue fluorescent dye Hoechst, red fluorescent dye PI and GFP (Kanduser *et al.*, 2009; Mars *et al.*, 2014).

Three independent cell viability experiments were counted by one user (940 images, containing 0–500 cells) as shown in Figure 3. Cells in type 1 experiment (cell viability) were stained with Hoechst, which gave us the total number of the cells (Fig. 3A), and with PI, which only stains dead or damaged cells (Fig. 3B). This combination of dyes can be used to determine cell viability (Bregar *et al.*, 2013). To obtain the number of viable cells, the number of dead cells was subtracted from the number of all cells (Fig. 3C).

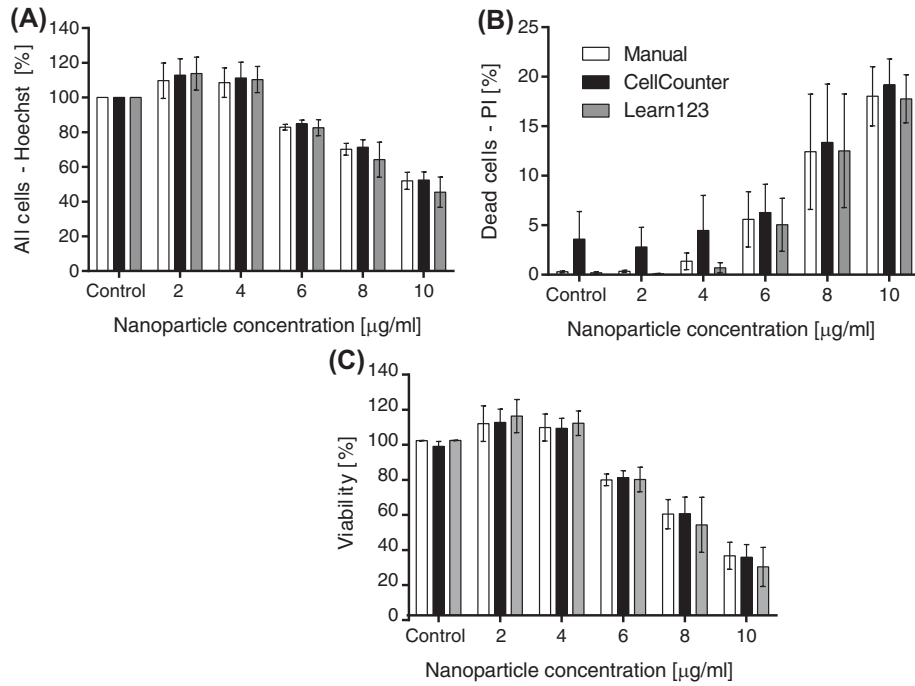


Fig. 3. Comparison of counting results of three independent experimental repeats counted manually and with CELLCOUNTER or LEARN123. Mean and standard deviation are shown ( $N = 3$ ).

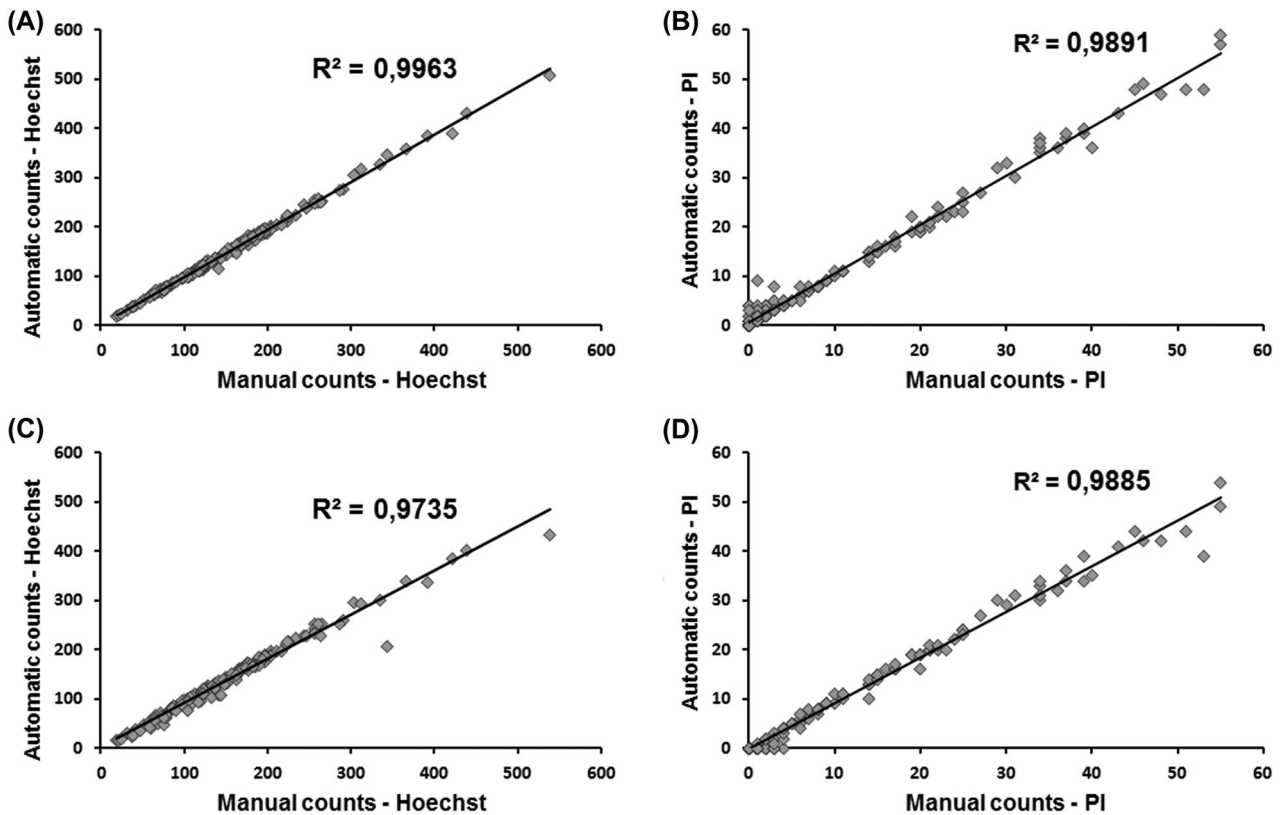


Fig. 4. Correlation of number of counted cells obtained manually and automatically by using (A) CELLCOUNTER for images with Hoechst stained nuclei and (B) propidium iodide (PI) stained dead cells and (C) by using LEARN123 for Hoechst and (D) PI images. Counting results for 170 images are shown for each fluorescent dye. Regression line and correlation coefficient  $R^2$  are shown for each correlation analysis.

The counting results showed reasonably good correlation and small standard errors. The differences between manual and automatic counting were mostly due to two or more objects in substantial contact being counted as one or due to objects positioned predominantly outside the margin of the image (Fig. 4). Generally, with Hoechst staining, we observed better performance in images with fewer and randomly scattered objects, where the comparison gave the correlation coefficient of  $R^2 = 0.9963$  for CELLCOUNTER (Fig. 4A) and  $R^2 = 0.9735$  for LEARN123 (Fig. 4C). Similar results were obtained when counting images with PI stained cells, although, the programs had occasional problems with recognition of objects of interest in images with low contrast, either images with no cells or images with dimly stained cells, resulting in correlation coefficient of  $R^2 = 0.9891$  for CELLCOUNTER (Fig. 4B) and  $R^2 = 0.9885$  for LEARN123 (Fig. 4D). In those images, CELLCOUNTER sometimes detected the background noise as an object of interest thus overestimating the number of objects. In both cases, as the number and fluorescence intensity increased with increasing number of dead cells, the counting accuracy increased as well (Fig. 3).

Additionally, one experiment (370 images in total, containing 0–500 cells) was analysed by three users to obtain the inter-personal error, which was found to be really low due to straightforward nature of the fluorescent objects (results not shown). These deviations, as well as the differences between manual and automatic counting, were small compared to the generally higher variation between independent experimental repeats due to great morphological and physiological variability of cells (Fig. 4)

Both programs work by improving the contrast of bright objects in the images (in our case fluorescently stained nuclei and/or cytoplasm) and convert the obtained greyscale images to binary images in which automatic detection and counting is performed (Fig. 1). As such, both programs do not require objects with clear borders and were found to be quite robust in accurately determining the number of cells even when the quality of images was not ideal, as seen in Figure 5. This allows less accurate and thus faster acquisition of images. Also, because the programs do not rely on object's size or shape, they can be applied to fluorescent images of cells of different shapes and sizes without additional optimization, as already confirmed for several different cell types (results not shown). The problems could emerge with fluorescent stains with uneven distribution inside cells, which could be recognized and counted as more than one cell.

The CELLCOUNTER program was additionally extended with a user friendly solution for automatic and semiautomatic counting of fluorescent objects in two overlapping images, as shown in Figure 6. Here, the solution is presented on an example of a transfection experiment, which requires simultaneous analysis of two fluorescent images of the same visual field to obtain the total cell number and the number of transfected, GFP positive cells. In this particular case, we transfected

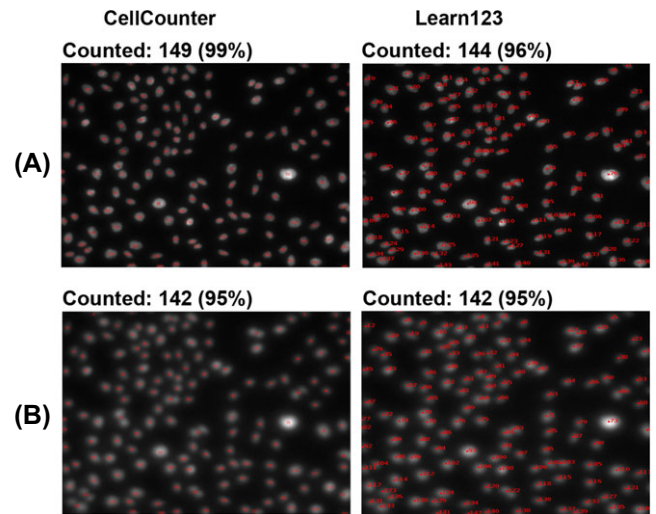
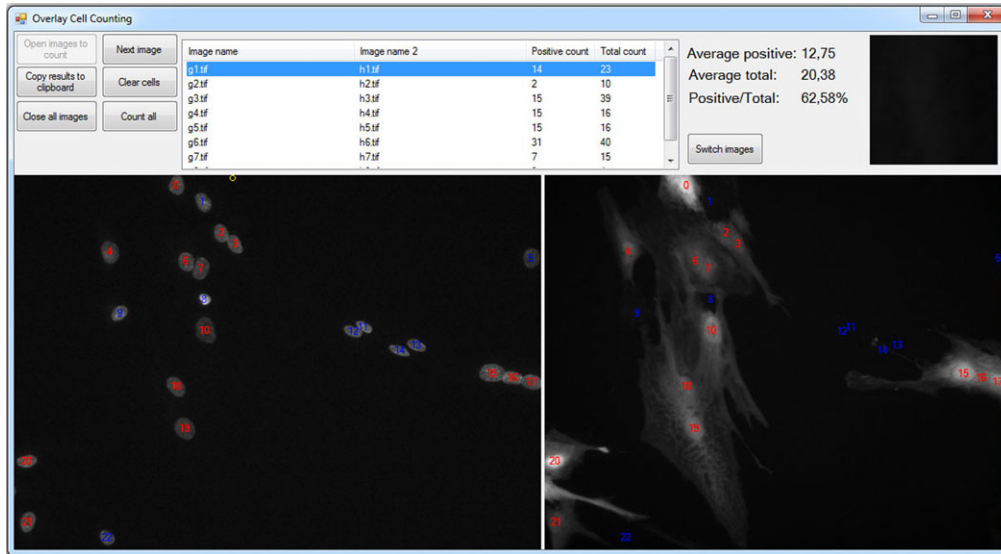


Fig. 5. A demonstration of the programs' robustness. The figures are images of Hoechst stained nuclei on the same visual field obtained in (A) focus plane and (B) out of focus plane. The total number of cells was obtained by manually counting the cells on phase contrast image (150 cells in total; data not shown). Above each image, the total number of automatically counted cells and the percentage of the cells compared to the manual count for CELLCOUNTER and LEARN123 are listed.

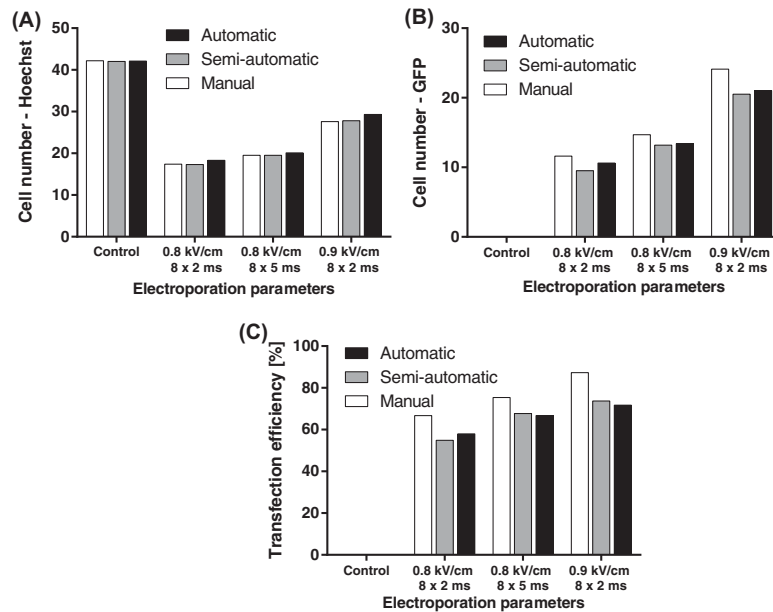
multinucleated myoblasts (Abmayr & Pavlath, 2012; Mars *et al.*, 2014). Multinucleated cells represent an additional challenge when determining the number of nuclei in a single cell on phase-contrast images (Glory *et al.*, 2004), where the nuclei are sometimes not visible. Even when using an image of fluorescently stained nuclei instead of a phase-contrast image, precise counting still requires colocalization (merging) of the two fluorescent images, which is a time-consuming task.

The presented extension automatically overlays the two corresponding fluorescent images, counts the nuclei and also determines the number of nuclei that colocalize with GFP positive cells in the second image. The user interface is arranged to display the fluorescent image with cell nuclei and the GFP image separately, but marks nuclei position in both images. This makes it easier to check and if necessary correct the automatically detected cell nuclei with simple mouse clicks, while on the second (GFP) image, the user can verify the accuracy of the program-determined positive and negative hits, which are labelled with different colours for easier verification (Fig. 6).

The overlay extension was also assessed for counting accuracy. The object counting algorithm again displayed a reasonably good correlation with manual counting (Fig. 7A), comparable to the accuracy obtained in the NP viability application images (Fig. 3). On the other hand, some discrepancies between automatic and manual counting occurred with detection of GFP positive cells (Fig. 7B). The main reason is the difference between program set threshold and the user's subjective judgment when classifying marginally GFP stained



**Fig. 6.** CELLCOUNTER program interface showing the result of automatic detection on overlaying images. The fluorescent image of Hoechst stained cell nuclei is displayed on the left and the image of GFP positive cells is displayed on the right. Each detected object (cell nuclei) is labelled so the user can quickly assert the accuracy of the detection. Cell nuclei colocalizing with GFP positive cells are labelled with red numbers and noncolocalized with blue for easier verification.



**Fig. 7.** Comparison of counting results obtained by counting fluorescently stained objects to determine the (A) number of Hoechst stained nuclei (total number of nuclei) of all multinucleated myoblasts and (B) number of nuclei in transfected GFP positive myoblasts 24 h after electroporation. (C) Transfection efficiency is represented as the ratio between number of nuclei colocalized with GFP positive cells and the total number of nuclei in the same visual field. Three counting methods were used: manually by counting Hoechst positive cells using ImageJ software, semiautomatically by correcting CELLCOUNTER detections and automatically by using CELLCOUNTER only. The results are presented as average cell number (in panels A and B) and average transfection efficiency (panel C) in each sample. Only one user analysed the data.

cells as positive or negative. The discrepancy is not very problematic as long as the same person or program analyses the whole experiment, since there are significant interpersonal differences in judgment among researchers or even definitions

of positive hits between research groups (Lumley *et al.*, 1997; Nattkemper *et al.*, 2003).

CELLCOUNTER and LEARN123 can therefore be used for automatic counting of fluorescent objects in microscopic images.



The programs can be applied to a range of different applications in various fields of life sciences where fluorescent labelling is used for quantification of various phenomena, from assessment of cell viability after certain treatment (Bregar *et al.*, 2013), to transfection efficiency (Mars *et al.*, 2014), yield of cell fusion (Trontelj *et al.*, 2010) and others. The interface of both programs is user friendly and straightforward, and require no additional image manipulation or tuning of parameters (raw images as obtained from the microscope can be used). Also, the entire ImageJ application was integrated in LEARN123 to further manipulate the images as well as the algorithm itself, since most of the filters in the counting algorithm are directly used in ImageJ. Images can be loaded through the drag and drop action. The program automatically analyses the images and allows simple corrections by adding or removing hits with mouse clicks. Both programs also enable to skip corrupted images. The results are automatically ordered and can be easily copied to clipboard and transferred to the editing program of choice. The processing time is negligible.

In CELLCOUNTER, in favour of processing speed and user friendliness, the counting parameters and thresholds are hardcoded and may not be optimal for all experimental protocols and labels, thus requiring more corrections from the user. LEARN123, on the other hand, requires an additional learning procedure, but can account for the variations in the images due to equipment, experimental protocols, fluorescent labels and user variability and thus can be applied in a larger number of experimental protocols. In both cases, the assistance provided by the presented programs made the tedious job of cell counting much easier and significantly faster (from hours to minutes) for several applications where fluorescent dyes like Hoechst, PI or others are used.

## Conclusions

In this paper, we present two easy-to-use solutions for automatic and semiautomatic counting of cells on fluorescent microscopic images. As we demonstrate, both programs (CELLCOUNTER and LEARN123) enable consistent, robust, fast and reasonably accurate detection of fluorescent objects and can therefore be applied to a range of different applications in different fields of life sciences where fluorescent labelling is used for quantification of various phenomena. Moreover, the CELLCOUNTER overlay extension also enables fast analysis of related images that would otherwise require image merging for accurate analysis, whereas LEARN123's evolutionary algorithm can adapt counting parameters to variability among the images with different experimental settings.

## Acknowledgements

The authors would like to thank Klemen Strojjan and Maja Jankovec for their help with the manual cell counting and Rok Mandeljc for the help with the manuscript. We would

like to thank the Institute of Pathophysiology (University of Ljubljana, Faculty of Medicine) for providing the cells for the experiments. This work was supported by Slovenian Research Agency within project J4-4324, J2-6758, J3-6794, young researchers program and MRIC UL IP-0510 Infrastructure program.

For academic performance evaluation, both Mojca Pavlin and Luka Šajn should be considered as lead authors of this paper.

## Conflicts of interest

No benefits in any form have been received or will be received from a commercial party related directly or indirectly to the subject of this paper.

## References

- Abmayr, S.M. & Pavlath, G.K. (2012) Myoblast fusion: lessons from flies and mice. *Development* **139**, 641–656.
- Ahmad, S.A. (2011) Analysis of Compound Enhancement Algorithms (CEA) based on Adaptive Histogram Equalization (AHE) on intra-oral dental radiographs images. *Int. J. New Comput. Archit. Their Appl. IJN-CAA* **4**, 902–916.
- Benali, A., Leefken, I., Eysel, U.T. & Weiler, E. (2003) A computerized image analysis system for quantitative analysis of cells in histological brain sections. *J. Neurosci. Methods* **125**, 33–43.
- Bregar, V.B., Lojk, J., Suštar, V., Veranič, P. & Pavlin, M. (2013) Visualization of internalization of functionalized cobalt ferrite nanoparticles and their intracellular fate. *Int. J. Nanomedicine* **8**, 919–931.
- Byun, J., Verardo, M.R., Sumengen, B., Lewis, G.P., Manjunath, B.S. & Fisher, S.K. (2006) Automated tool for the detection of cell nuclei in digital microscopic images: application to retinal images. *Mol. Vis.* **12**, 949–960.
- Cai, Z., Chattopadhyay, N., Liu, W.J., Chan, C., Pignol, J.-P. & Reilly, R.M. (2011) Optimized digital counting colonies of clonogenic assays using ImageJ software and customized macros: comparison with manual counting. *Int. J. Radiat. Biol.* **87**, 1135–1146.
- Carpenter, A.E., Jones, T.R., Lamprocht, M.R., *et al.* (2006) CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol.* **7**, R100.1–R100.11.
- Chang, F., Chen, C.-J. & Lu, C.-J. (2004) A linear-time component-labeling algorithm using contour tracing technique. *Comput. Vis. Image Underst.* **93**, 206–220.
- Chan, Y.-K., Tsai, M.-H., Huang, D.-C., Zheng, Z.-H. & Hung, K.-D. (2010) Leukocyte nucleus segmentation and nucleus lobe counting. *BMC Bioinformatics* **11**, 1–18.
- Cheng, H.D. & Shi, X.J. (2004) A simple and effective histogram equalization approach to image enhancement. *Digit. Signal Process.* **14**, 158–170.
- Dein, F.J., Wilson, A., Fischer, D. & Langenberg, P. (1994) Avian leucocyte counting using the hemocytometer. *J. Zoo Wildl. Med.* **25**, 432–437.
- Dell, H.A. (1954) Stages in the development of an arrested scan type microscopic particle counter. *Br. J. Appl. Phys.* **5**, S156–S161.
- Faustino, G.M., Gattass, M., Rehen, S. & deLucena, C. (2009) Automatic embryonic stem cells detection and counting method in fluorescence microscopy images. In *IEEE International Symposium on Biomedical*

- Imaging: From Nano to Macro*, Boston, MA, USA, 2009. ISBI '09, 799–802.
- Gallagher, S.R. (2008) Digital image processing and analysis with ImageJ. *Current Protocols Essential Laboratory Techniques* (ed. by Sean R. Gallagher and Emily A. Wiley). John Wiley & Sons, Inc, Wiley, Hoboken, NJ, USA.
- Ganias, K., Rakka, M., Vavalidis, T. & Nunes, C. (2010) Measuring batch fecundity using automated particle counting. *Fish. Res.* **106**, 570–574.
- Glory, E., Faure, A., Meas-Yedid, V., Cloppet, F., Pinset, C., Stamon, G. & Olivo-Marin, J.-C. (2004) A quantification tool to analyse stained cell cultures. *Image Analysis and Recognition, Lecture Notes in Computer Science* (ed. by A. Campilho & M. Kamel), pp. 84–91. Springer, Berlin, Heidelberg.
- Haralick, R.M. & Shapiro, L.G. (1985) Image segmentation techniques. *Comput. Vis. Graph. Image Process.* **29**, 100–132.
- Jones, K.H. & Senft, J.A. (1985) An improved method to determine cell viability by simultaneous staining with fluorescein diacetate-propidium iodide. *J. Histochem. Cytochem. Off. J. Histochem. Soc.* **33**, 77–79.
- Jong, K.A. de D. (2002) *Evolutionary Computation*. 1st edn. A Bradford Book, Cambridge, MA.
- Kachouie, N., Kang, L. & Khademhosseini, A. (2009) Arraycount, an algorithm for automatic cell counting in microwell arrays. *BioTechniques* **47**, x–xvi.
- Kanduser, M., Miklavcic, D. & Pavlin, M. (2009) Mechanisms involved in gene electrotransfer using high- and low-voltage pulses—an in vitro study. *Bioelectrochemistry Amst. Neth.* **74**, 265–271.
- Lakowicz, J.R. (2006) *Principles of Fluorescence Spectroscopy*. 3rd edn. Springer, New York, US.
- Li, K., Miller, E.D., Weiss, L.E., Campbell, P.G. & Kanade, T. (2006) Online tracking of migrating and proliferating cells imaged with phase-contrast microscopy. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*. p. 65. IEEE Computer Society Press, Los Alamitos.
- Lumley, M.A., Burgess, R., Billingham, L.J., McDonald, D.F. & Milligan, D.W. (1997) Colony counting is a major source of variation in CFU-GM results between centres. *Br. J. Haematol.* **97**, 481–484.
- Marjanovič, I., Kandušar, M., Miklavčič, D., Keber, M.M. & Pavlin, M. (2014) Comparison of flow cytometry, fluorescence microscopy and spectrofluorometry for analysis of gene electrotransfer efficiency. *J. Membr. Biol.* **247**, 1259–1267.
- Mars, T., Strazisar, M., Mis, K., Kotnik, N., Pegan, K., Lojk, J., Grubic, Z. & Pavlin, M. (2014) Electrotransfection and lipofection show comparable efficiency for in vitro gene delivery of primary human myoblasts. *J. Membr. Biol.* **248**, 273–283.
- Nattkemper, T.W., Twellmann, T., Ritter, H. & Schubert, W. (2003) Human vs machine: evaluation of fluorescence micrographs. *Comput. Biol. Med.* **33**, 31–43.
- Ongena, K., Das, C., Smith, J.L., Gil, S. & Johnston, G. (2010) Determining cell number during cell culture using the sceptor cell counter. *J. Vis. Exp.* **45**, 1–5.
- Otsu, N. (1979) A threshold selection method from gray-level histograms. *Syst. Man Cybern. IEEE Trans. On* **9**, 62–66.
- Pavlin, M., Flisar, K. & Kanduser, M. (2010) The role of electrophoresis in gene electrotransfer. *J. Membr. Biol.* **236**, 75–79.
- Pavlin, M., Pucihar, G. & Kandušar, M. (2012) The role of electrically stimulated endocytosis in gene electrotransfer. *Bioelectrochemistry Amst. Neth.* **83**, 38–45.
- Peer, P. & Corzo, L.G. (2007) Local pixel value collection algorithm for spot segmentation in two-dimensional gel electrophoresis research. *Comp. Funct. Genomics*, pp. 1–91.
- Ramin, M., Ahmadvand, P., Sepas-Moghaddam, A. & Dehshibi, M.M. (2012) Counting the number of cells in immunocytochemical images using genetic algorithm. *IEEE Int. Conf. Hybrid Intell. Syst.* **2012**, 185–190.
- Rieseberg, M., Kasper, C., Reardon, K.F. & Scheper, T. (2001) Flow cytometry in biotechnology. *Appl. Microbiol. Biotechnol.* **56**, 350–360.
- Selinummi, J., Seppälä, J., Yli-Harja, O. & Puhakka, J.A. (2005) Software for quantification of labeled bacteria from digital microscope images by automated image analysis. *BioTechniques* **39**, 859–863.
- Sharpe, A.N., Diotte, M.P., Peterkin, P.I. & Dudas, I. (1986) Towards the truly automated colony counter. *Food Microbiol.* **3**, 247–270.
- Stephens, D.J. & Allan, V.J. (2003) Light microscopy techniques for live cell imaging. *Science* **300**, 82–86.
- Urish, K.L., Deasy, B.M. & Huard, J. (2013) Automated classification and visualization of fluorescent live cell microscopy images. *J. Microsc.* **249**, 206–214.
- Usaj, M., Torkar, D., Kanduser, M. & Miklavcic, D. (2011) Cell counting tool parameters optimization approach for electroporation efficiency determination of attached cells in phase contrast images. *J. Microsc.* **241**, 303–314.
- Usaj M., Trontelj K., Miklavcic D. & Kanduser M. (2010) Cell-cell electrofusion: Cell-cell electrofusion: optimization of electric field amplitude and hypotonic treatment for mouse melanoma (B16-F1) and Chinese Hamster ovary (CHO) cells. *J Membr. Biol.* **236**, 107–116.
- Väyrynen, J.P., Vornanen, J.O., Sajanti, S., Böhm, J.P., Tuomisto, A. & Mäkinen, M.J. (2012) An improved image analysis method for cell counting lends credibility to the prognostic significance of T cells in colorectal cancer. *Virchows Arch. Int. J. Pathol.* **460**, 455–465.
- Vincent, L. & Soille, P. (1991) Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Trans. Pattern Anal. Mach. Intell.* **13**, 583–598.
- Wu, K., Gauthier, D. & Levine, M.D. (1995) Live cell image segmentation. *IEEE Trans. Biomed. Eng.* **42**, 1–12.
- Xu, X., Xu, S., Jin, L. & Song, E. (2011) Characteristic analysis of Otsu threshold and its applications. *Pattern Recognit. Lett.* **32**, 956–961.
- Zuiderveld, K. (1994) Contrast limited adaptive histogram equalization. *Graphics Gems IV*, pp. 474–485. Academic Press Professional, Inc, San Diego, CA, USA.