10th International IFAC Symposium on
Computer Applications in Biotechnology
Preprints Vol.1, June 4-6, 2007, Cancún, Mexico

APPLICATION OF IN-SITU-MICROSCOPY AND DIGITAL IMAGE PROCESSING IN YEAST CULTIVATIONS

P. Lindner, C. Krabichler, G. Rudolph, T. Scheper, B. Hitzmann

Institut für Technische Chemie
Gottfried Wilhelm Leibniz Universität
Callinstr. 3, 30167 Hannover, Germany

Abstract: In-situ-microscopy is an efficient method to measure cell concentration during cultivations. Furthermore the size distribution of the cells as well as their morphology can be determined on-line. To operate the microscope and acquire sequences of cell images a software system has been developed. This software also allows the on-line analysis of the image data to determine the cell concentration. The software, the image analysis algorithm as well as the analysis results of a yeast cell cultivation are presented. It is demonstrated, that the automated determination of the cell numbers agree very well with its manual count.

Keywords: in-situ microscopy, yeast cell cultivation, image analysis, cell concentration.

1. INTRODUCTION

Important parameters of cell cultivations are the cell concentration, cell size distribution and cell morphology. In-situ-microscopy (Bittner, *et. al.*, 1998; Suhr, *et. al.*, 1995) is a very useful measurement technique that allows inline measurement of these parameters. It can be used for yeast cell cultivations (Bittner, *et al.*, 1998; Camisard, *et al.*, 2002) as well as mammalian cell cultivations (Joeris, *et al.*, 2002; Frerichs, *et al.*, 2002; Guez, *et al.*, 2004). One of the major advantages of in-situ-microscopy is that no sampling is necessary. On one hand a lot of time and money can be saved by using in-situ-microscopy and on the other hand the risk of contaminations can be minimized when there is no need to take samples of the cultivation broth to determine the cell concentration. The cell concentration can be determined from the measurement signal - digital images acquired by a CCD camera - by application of suitable image processing algorithms. In this contribution the software that has been developed to control the microscope as well as to analyze the images is presented.

2. THE IN-SITU-MICROSCOPE

The in-situ-microscope described in this contribution was developed at the Institut für Technische Chemie at the Leibniz Universität in Hannover, Germany. It is a transmitted light microscope and consists of a microscope body with the sampling zone and a linear table with two stepper motors and a CCD digital camera. The microscope can be mounted on a bio-reactor via a 25 mm port.

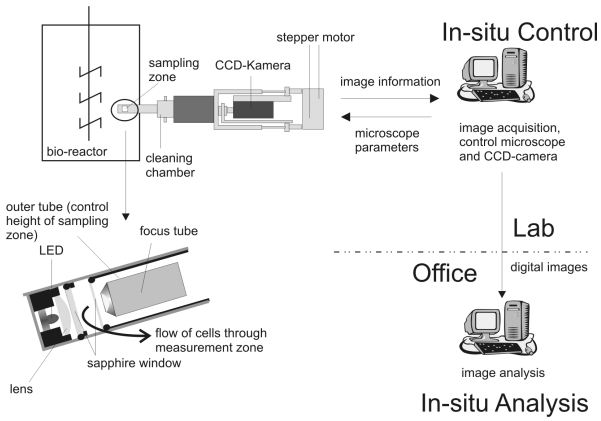

Fig. 1. The in-situ-microscope attached to a bio-reactor and a detailed view of the measurement zone. On the right hand side the function of the programs In-situ-Control and In-situ- Analysis is shown.

The height of the sampling zone, which is dipped in the cultivation broth, can be controlled by moving the inner tube with a stepper motor. Two sapphire windows separate the broth in the sampling zone from the microscope. The sampling zone is lighted by a LED below the zone. To ensure that the content of the measurement zone are representative of the media in the reactor experiments were carried out to compare cell concentrations measured with in-situ microscopy with those of other counting methods (Cedex for example). The results from in-situ microscopy agree well with results from other methods, as has been shown in the literature (Brückerhoff et al. 2004, Brückerhoff 2006).

With a second stepper motor the objective tube can be adjusted. In this fashion the camera can zoom in on the cells. A sketch of the microscope and a detailed view of the sampling zone is shown in Figure 1. This figure also shows the function of the two programs which are needed to operate the microscope and analyze the image data: In-situ-Control and In-situ-Analysis.

3. IN-SITU-CONTROL

The software In-situ-Control was developed with the Delphi programming language (Version 7) from Borland. The connection to the camera is established via the ICImaging Control plugin from The Imaging Source Europe GmbH (Version 1.41). This plugin enables Delphi programs to set and read camera parameters and to obtain the image data. The microscope can be connected via a serial port. A screenshot of the program is shown in Figure 2. In the main window a current image of the sampling zone content is shown. This image is updated 2-15 times per second, depending on the camera settings. Prior to a measurement this so called “live image” can be used to focus on the objects in the sampling zone by moving the objective tube.

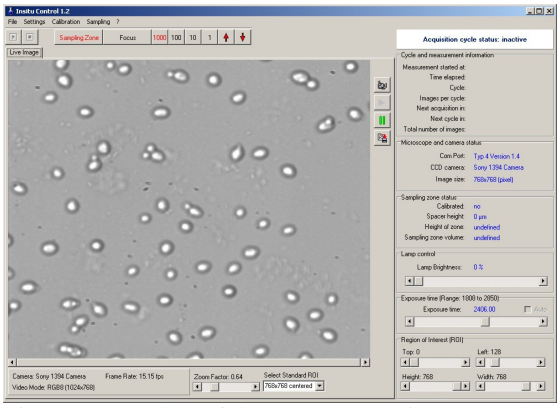

Fig. 2. Screenshot of the program In-situ-Control.

The objective tube as well as the tube which sets the sampling zone height can be moved with buttons that are located above the live image. Also the LED intensity and various camera parameters, like video mode, contrast or exposure time, can be set with graphical elements in the main window. To allow for the acquisition of partial images the region of interest (ROI) can be defined before the measurement. If the goal of the measurement is the determination of the cell concentration the volume of the sampling zone has to be known. Otherwise only the cell number can be obtain from the image data. For this reason the sampling zone can be calibrated: If it is completely closed the height and volume can be set to zero. After that, all tube movements are detected and converted into the current height and volume of the sampling zone.

When all microscope and camera parameters are set the measurement parameters can be defined. This includes the number of images in each cycle and the number of cycles as well as the time intervals between each image and cycle. The advantage of an image acquisition in cycles and not in a continuous fashion is, that if for example the cell concentration should be measured every hour it makes no sense to take only one picture per hour. This single picture does not show the “true” cell concentration but a concentration spread around the true value with a certain variance. It is more accurate in this case to acquire 100 pictures every second in each cycle with a time interval of one hour between each cycle. The cell concentration of each cycle can be determined from the mean value of all pictures in a cycle.

After a measurement has been started it runs completely automatic. The images are saved in bmp format with unique file names and date and time stamp. For each image a parameter file is created with the same name but the extension log. This logfile contains the microscope, camera and measurement settings from the time when the image has been saved. Also an overview file is generated that contains the names of all files that belong to the measurement. This file can be used by other data processing programs to do operations on all images of a measurement.

4. IN-SITU-ANALYSIS

The software In-situ-Analysis also was developed with the Delphi programming language (Version 7) from Borland. This program was designed to analyze single images and whole measurement sequences, that have been acquired with In-situ-Control. The analysis can be made on-line (at the same time the measurement is taken) as well as after the measurement is completed.

The information that is hidden in the image data can be transformed into result values like cell number, cell area, cell volume and a classification of the cells

with a suitable image analysis algorithm. The most important aspect during development of the software was its flexibility. For this reason the image analysis algorithms are not included in the main program but can be loaded dynamically at runtime from a dynamic link library (DLL). If new analysis algorithms become available or have to be developed there is no need to change the main program, one only has to generate a DLL that contains the algorithm.

In-situ-Analysis has a graphical user interface as shown in Figure 3 that provides the general functionality that is needed for image analysis: Display of original and result images, display of the analysis results in tables and diagrams, set display and image analysis options, postprocessing of result data (filter and averaging operations etc.) and export of results for further processing with other software.

The communication between the main program and the DLL is enabled with a flexible data exchange protocol. All display elements and processing functions in In-situ-Analysis can adapt to the amount and type of information that is generated from an input image by the currently loaded algorithm DLL. The program structure is presented in Figure 4. The ellipse represents an external image analysis algorithm.

At the moment image analysis algorithms for yeast, BHK, CHO (all suspension cells) and adherent BHK cells are available.

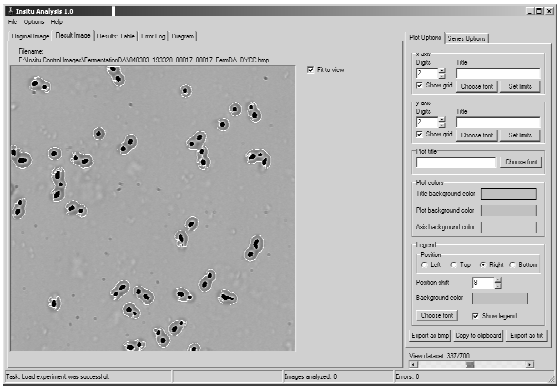

Figure 3. Screenshot of the program In-situ-Analysis. In the main window an analysed image from a yeast cultivation can be seen. Detected cell kernels are coloured black and cell borders are marked with a white line.

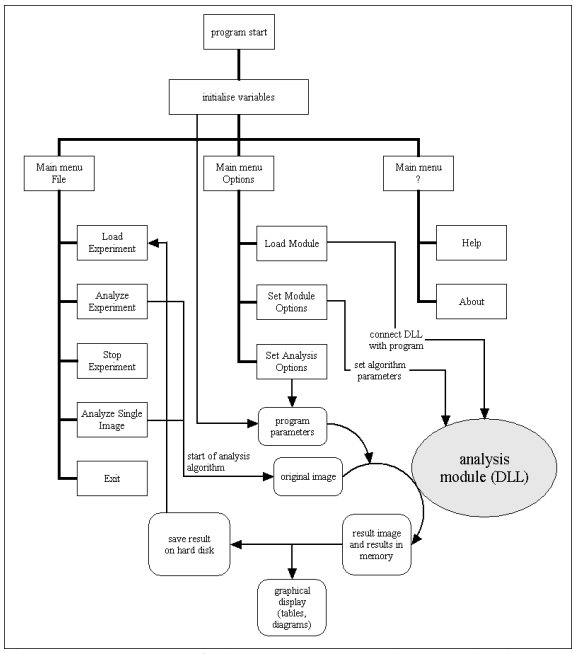

Fig 4. Structure of the program In-situ-Analysis. The analysis module that contains the image analysis algorithm is not part of the program.

5. IMAGE ANALYSIS ALGORITHM FOR YEAST CELL CULTIVATIONS

The image analysis algorithm for yeast cell cultivations has been developed with the Delphi programming language (Version 7) from Borland. Cell images have to be acquired in a defocussed fashion to be suitable for analysis with this algorithm. In this case yeast cells act as lenses and focus light in the middle of the cell. On images each yeast cell has a white kernel and a dark edge, compared to the background, as can be seen in Figure 5, where cells from three different time steps (2 h, 8.5 h and 17.5 h) during the cultivation can be seen.

The algorithm has seven parameters and generates the following result values from an input image: Number of objects (if two or more cells form a cluster, this cluster is treated as one object), number of pixels in objects that contain cells, number of cells, number of single cells, number of double cells, average pixels per cell, cells per mL. It also classifies the cells into the following categories: small, medium or large single cell, double cell, cell cluster and no classification possible.

First the mode value – the most common gray value in the image – is calculated. Then each pixel of the original image that differs from the mode value by at least the gray value threshold (first parameter of the algorithm) is marked with a flag. In this way all cells are detected except a circular area between the kernel and the cell border where the gray value is similar to the background. Pixels in these areas are marked in the next step using a search algorithm.

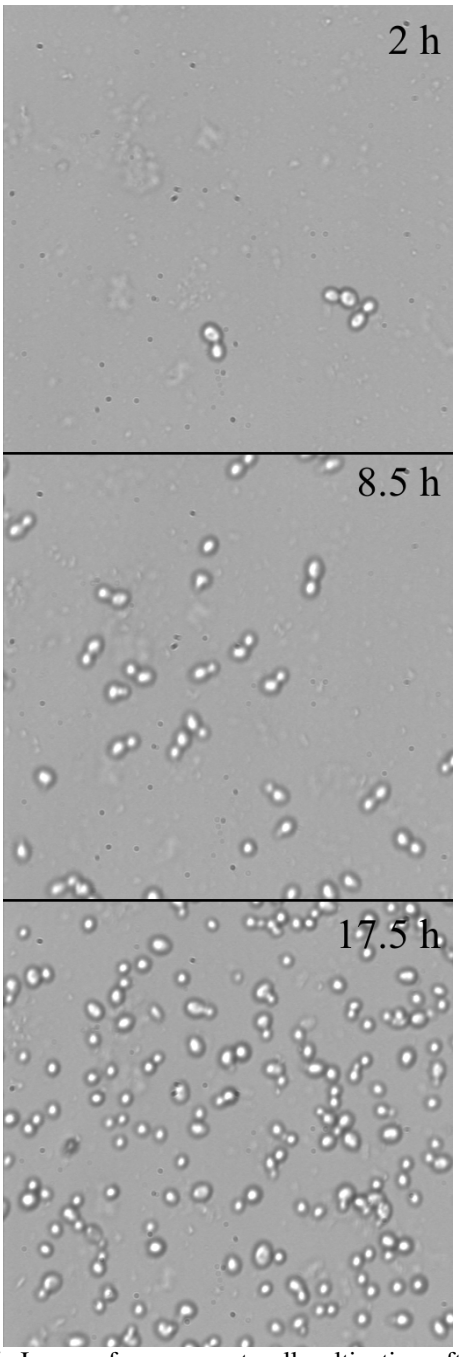

Fig 5. Images from a yeast cell cultivation after 2, 8.5 and 17.5 hours. All images have been collected defocussed so that yeast cells have a white kernel and a dark edge. This simplifies digital image processing.

It scans the image for flagged light pixels that have a flagged dark pixel next to them and one to three unflagged pixels between them.

After the first two steps all pixels belonging to a cell have been marked. Of course, during this procedure non-cell objects that differ significantly from the background and background noise gets marked as well. In the following step all objects are identified and properties are assigned to each one of them to distinguish between cell and non-cell objects. To accomplish this the image is examined with a border detection algorithm: The image is scanned from left to right and row by row until a marked pixel is found. Then the border detection algorithm is started, which operates with the following rules (starting with the direction "right"):

- On a marked pixel turn left and advance one step
- On an unmarked pixel turn right and advance one step
- After three turns in the same direction ignore the other rules, turn to the other direction and advance one step

This principle is illustrated in Figure 6. Within each step the pixel is marked as a border pixel. When the start location is reached again all pixels inside this border are marked with an object flag. Then the scanning process starts again until another marked pixel is found and so on.

In this way all pixels which belong to an object get marked (different flags are used each time so that the objects become distinguishable). By counting the number of object flags for each object its size can be determined. Objects that are smaller than a size threshold value (second parameter of the algorithm) are eliminated.

After this the amount of cells in each object is calculated. This is done by scanning the inside area of an object for pixels than are lighter than the mode value by at least the white kernel threshold value (third parameter of the algorithm). When a kernel pixel has been found the scanning process is interrupted and the kernel border and the inside area are detected by the same algorithm that was used to find the cells in the image (see above).

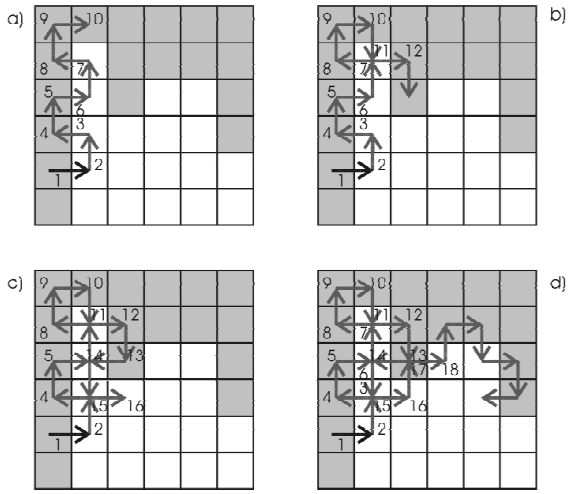

Figure 6. Illustration of the border detection algorithm. The gray squares are background and the white squares are an example object. Starting at the location 1 (black arrow) the algorithm surrounds an object of arbitrary shape as demonstrated with the gray arrows.

Now, in the final step, the information contained in the flags is transformed into the result values stated above: The number of objects is given by the number of different object flags in the image. The number of cells is given by the number of white kernels found in the objects. The average cell area is given by the total number of object flags divided by the number of cells. The cell concentration in cells per mL is calculated from the cell number and the sampling zone volume that can be taken from the log-file of the corresponding image (see above). The classification of the cells into small, medium and large cells is done by using size threshold values, which are parameters of the algorithm.

To verify the results of the algorithm 700 images of a yeast cell cultivation have been evaluated manually. The cell number per image was counted several times to minimize counting errors. In Figure 6 a comparison of the manual count with the algorithm result is shown. The image acquisition during this cultivation was done in 35 measurement cycles with 20 images per cycle. Each dot in Figure 7 represents the mean value of the cell number of all 20 images of a cycle. The match between manual and algorithm results is very good. As one can see from Figure 7 at higher cell numbers per image the variance is increasing. The counting speed of the algorithm equals to 100 images (512x512 pixel, 8 bit color depth) in 11 seconds on a Intel Pentium 4 CPU with 3.2 GHz, 1 MB DDR-SDRAM and Windows XP. The precision and speed of this yeast cell counting algorithm makes it a very usefull tools in the evaluation of yeast cultivations.

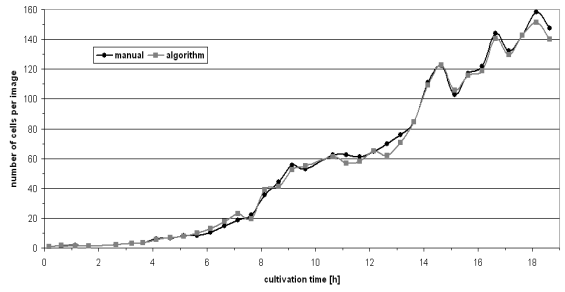

Fig 7. Comparison of the manual count with the algorithm result (number of cells per image versus cultivation time).

6. CONCLUSION

In this contribution a fully automated in-situ-microscope measurement system is presented. The functionality of this system is described and its application for monitoring of a *Saccharomyces cerevisiae* cultivation is reported. Without any sampling the cell concentration has been determined. This demonstrates the power of the system.

REFERENCES

Bittner, C., Wehnert, G., Scheper, T. (1998). In situ microscopy for on-line determination of biomass. *Biotechnol Bioeng* **60**, 24-35.

Brückerhoff, T. et al. (2004). Image analysis based inline monitoring of mammalian and yeast cell cultivation processes. *Technische Systeme für Biotechnologie und Umwelt: 12. Heiligenstädter Kolloquium, Tagungsband,* 125

Brückerhoff, T. (2006). Bildbasiertes Inline-Monitoring von Kultivierungsprozessen mit einem optimierten In-situ Mikroskopsystem. Ph. D. Thesis, Universität Hannover

Camisard, V., Brienne, J.P., Baussart, H., Hammann, J., Suhr, H. (2002). Inline characterization and cell volume in agitated bioreactors using in situ microscopy: application to volume variation induced by osmotic stress. *Biotechnol Bioeng* **78**, 73-80.

Frerichs, J.G., Joeris, K., Konstantinov, K., Scheper, T. (2002). Einsatz eines In-situ Mikroskops zur Online-Beobachtung von Tierzellkultivierungen. *Chem Ing Tech* **74**, 1629-1633.

Guez, J.S., Cassar, J.P., Wartelle, F., Dhulster, P., Suhr, H. (2004). Real time microscopy for animal cell-concentration monitoring during high density culture in bioreactor. *J Biotechnol* **111**, 335-343.

Joeris, K., Frerichs, J.G., Konstantinov, K., Scheper, T. (2002). In-situ microscopy: online process monitoring of mammalian cell cultures. *Cytotechnology* **38**, 129-134.

Suhr, H., Wehnert, G., Schneider, K., Bittner, C., Scholz, T., Geissler, P., Jähne, B., Scheper, T. (1995). In situ microscopy for on-line characterization of cell-populations in bioreactors, including cell-concentration measurements by depth from focus. *Biotechnol Bioeng* **47**, 106-116.