A Machine Vision-based *In Situ* Probe for Automated On-Line Measurement of Cell Density and Viability by Dark Field Microscopy, Image Processing and Pattern Recognition

A thesis submitted for obtaining the doctoral

degree of natural science (Dr. rer. nat)

at the

Faculty of Technology

University of Bielefeld

by

Ning Wei

May 2007

Acknowledgements

I show special gratitude to my brother, Chen Po. His inspiration, care and joyfulness are like an oasis in the desert.

I dedicate this work to my parents, and my sister, for all the support they gave to me throughout my life.

Index

Chapter 1

Introduction

Discovery of new biological information and extraction of knowledge from all kinds of biological entities has been a hotspot in recent biomedical research. These entities have included macromolecules (e.g. DNA, RNA, proteins), subcellular structures (e.g., membranes, nucleus, mitochondria), cells, tissues, organs, and so on. Much effort has been made in finding the connections between phenotype and genotype, between function of a biological system (like a cell) and its features (proteome, transcriptome, metabolome, etc.). Obviously, cell viability is one of the basic characteristics indicating the physiological state of the cell. Thus, it has long been one of the major considerations. Recently, many projects have been carried out on studying mechanisms of cell death (Green and Kroemer, 2004; Plas and Thompson, 2002; Madeo et al., 1999; Majno et al., 1995). Extracting information on cell viability will benefit a variety of biological applications, for instance fermentation processes, in which properties related to the state of cell growth are required to be monitored, like cell density, viability, mobility, size, shape, aggregation, etc.. Among these, biomass concentration (or cell density) and viability are the most important, as these values are often directly related to the productivity of the biological system. In consequence, it would be of great value if these quantities could be determined on-line in bioreactors.

Current Techniques

Preliminarily, cell growth is monitored by investigating biomass concentration or cell density. One of the most traditional approaches to measuring biomass is to determine the dry weight of the cells of a chosen volume of cultivation broth through a cascade of processes, like centrifugation, washing, drying and weighing. This approach is, though intuitive and straightforward, hardly put to use for on-line measurement due to its time-consuming nature. In the last 30 years, various methods for on-line estimation of biomass have been presented, including measurement of optical density, fluorescence, thermal properties, pH value, *etc*. (For details see Harris and Kell, 1985; Scheper, 1991; Sonnleitner et al., 1992).

However, these techniques are rarely applied in industry due to rigid requirements in industrial practice. An alternative of monitoring cell growth is to measure cell density. Using a hemocytometer under a laboratory microscope, it can be determined manually by experienced examiners. With progresses in modern technologies, tasks like cell counting, cell detection or cell density estimation can be accomplished by means of automated image processing and machine learning (Thomas and Paul, 1996; Huls et al., 1992; Vecht-Lifshitz und Ison, 1992; Pons et al., 1993; Nattkemper et al., 2001&2003; Wei et al., 2005).

Apart from cell density, cell viability is also directly connected to the productivity of the microorganisms. Consequently, it is regarded as one of the critical parameters in growth control. Microorganisms in a bioreactor can be physiologically active (live) or inactive (dead). As a result, biomass can be classified as viable biomass (in which cells are physiologically active) and necromass (in which cells are no longer physiologically active).

In common methods for viability determination, reagents are normally added to the cells, and viability of a cell is detected according to its reaction to the reagents. A variety of assays are available for determining viability of yeast, plant and mammalian cells. (For a survey see Castro-Concha et al. (2006), Cook and Mitchell (1989), and Heggart et al. (2000)). In general, viable cells can be distinguished from dead ones according to either their physical properties (such as membrane integrity and cytoplasmic streaming), or their metabolic activities (such as cellular energy capacity, macromolecule synthesis capacity, and hydrolysis of fluorogenic substrates). Applicable reagents for assessing cell membrane integrity include methylene blue, methylene violet, evans blue, trypan blue, propidium iodide and other permeability reagents. In order to assess metabolic activities, derivatives of fluorescein, radioactive isotope labelled thymidine, etc. have been used. These reagents are reliable, however, some of them might be invasive and even toxic to the target cells.

There are two main categories of techniques available for the measurement of cellular properties: off-line and on-line techniques. In the former case, a portion of the culture broth is taken out for further analysis; while in the latter case, cell properties are measured directly in the bioreactor.

One typical off-line system is a CASY cell count analyser (Schärfe System GmbH, Reutlingen, Germany). This is an efficient, sophisticated instrument based on flow cytometry technique. It has in total 512,000 channels, resulting in more accurate and rapid measurements. It possesses comprehensive measurement functionalities, including measurement of cell density, cell diameter, cell size, and cell volume. Electric pulse measurement technique is used in analysis of membrane integrity to avoid the toxic effects of staining on cells, thus enabling the

measurement of cell viability. Using a corrector for cell agglomerate, the size of cell agglomerate can be measured.

Another example of an off-line system is Nucleo Counter developed by ChemoMetec Inc. (Allerod, Denmark). Propidium iodide (PI) is a common fluorescence dye for labelling nuclei. It cannot pass the intact cell membrane, however, it can get through membranes of dead cells and redden the nuclei. PI is normally excited by green light (540 nm wavelength) and emits a bright fluorescence at 600 nm (red light). The intensity of the fluorescence released by PI combined with nuclei is normally 20~30 times larger than that not combined with nuclei. The core of this system is a fluorescence microscope, which includes a light emitting diode (LED) light source, an optical system (lens, filters, etc.), and a charge-coupled device (CCD) camera. The LED gives off green light for excitation. An excitation filter separates the green light, a fluorescence filter separates red light from PI-DNA complex. The CCD camera records red light, and the signals are converted to cell count.

Apart from off-line techniques, nowadays more and more on-line systems are developed and implemented for biological applications. Currently available on-line instruments for cell density measurement are based on turbidity (e.g. Aquasant Messtechnik AG, Bubendorf, Switzerland), optical density (e.g. generic Photometer with dip probe or bypass) or fluorescence (Bioview, Delta Light & Optics, Lyngby, Denmark), and so on. One of the on-line instruments for cell viability monitoring is an Aber detector. It is an *in situ* viable cell detector based on capacitance measurement (Watson et al., 1994; Kronlof, 1991; Van Impe, et al., 1998; Claes and Van Impe, 1999). Under the influence of certain electric fields, a cell with intact plasma membrane is polarized and behaves as a tiny capacitor, which results in the alteration of the overall capacitance. In principle, the system only responds to viable cells and is not sensitive to gas bubbles, microcarriers, cell debris, cell blob, dead cells, fermentation liquid foam, solid medium particles, or micro-particles in the suspension. Nevertheless, this kind of probe is normally supposed to be used in conjunction with another probe exclusively for measuring viability of the overall biomass (including both live and dead cells), thus, increasing the system's complexity.

TruCell (Finesse LLC, California, USA) is a real-time on-line cell density monitor with the integration of launcher and sensor. It is based on infrared sensing. Because its near-infrared sensor is only 12 mm in diameter, it can be implemented even in the smallest desktop bioreactor. Meanwhile, it also provides a series of products for middle- or large-scale bioreactors. Sensors provide two forms of communication: the browser interface-based Bluetooth wireless technology and Fieldbus technology. Based on the experimental methods and the individual needs

of customers, the equipment can convert data into OD or cell density. During the use of this sensor, the growth curve, nutrition take-in, change in cultivation phase and harvest cycle can be optimized. Sensor configurations and platform options facilitate and optimize the operation of customers.

Another on-line monitoring system for assessment of cell viability and functions in a bio-artificial liver support system was reported by Xiang et al. (2005). The system can work without undermining the environment of the cells in the bioreactors, and respond quickly to the vital status and functional activity within the liver cells, in order to take timely measures on corresponding control. Different from the traditional methods of large-scale off-line instruments, this system uses a monochromatic light source, an integrated optoelectronic sensor as measuring component and a control system based on a monolithic processor for on-line rapid detection of metabolism. Results show that the system can be used as a testing system to assess the biological activity of the livers, thus enabling the closed-loop control of the production.

The results of all these on-line measurements are verified by offline techniques, for instance, by counting cells with a hemocytometer under microscope.

Due to the fact that biomass concentration (or cell density) and viability are important properties for indicating the state of cell growth, reliable on-line probes for measuring both cell density and cell viability have long been requested by the industries. Up to now, two types of *in situ* probes have been developed for on-line monitoring of cell growth (Suhr et al., 1995; Bittner et al., 1998). Suhr et al. constructed their probe on the basis of epifluorescence microscopy, while Bittner et al. built the probe based on bright field microscopy. Nevertheless, both of these two probes can only measure the biomass or cell density, lacking the capability of assessing viability.

The Goal

The goal of this thesis is to develop an on-line probe that can be implemented to measure the *in situ* cell density and viability in bioreactors. This task includes not only hardware development, but also the development of suitable software that can fast and accurately process the signal generated by the hardware.

In spite of the diversity of the methods for on-line measuring cell density or viability separately, it appeared to be a nontrivial task to build a probe suitable for both properties. Even a simple combination of any two of the techniques would lead to great technical complexity. In order to avoid the system complexity, a special probe based on dark field microscopy has been proposed in this work. A reflective condenser has been designed and built to achieve dark field illumination

and high contrast images. The imaging system of the probe is also designed and configured in accordance with the system's requirements.

In order to obtain accurate cell density and viability, programs have been written to evaluate the images captured by the probe. The core of the programs is implementing classifiers based on supervised machine learning. Different pattern recognition methods have been utilized in order to find the best way of image processing for the system.

Outline of the Manuscript

In the next chapter, details of the hardware of the on-line probe will be given. In Chapter 3, the performance of this probe in measuring yeast cell density and viability in a bioreactor will be shown. Methods of preparation of different type of cell populations, and the principle of training and test of the classifiers will also be introduced. In Chapter 4, viability classification based on wavelet feature computing and selection is proposed to improve the performance of the system. In Chapter 5 and 6, efforts have been made to extract information from time series images in order to distinguish live and dead cells accurately. While principal component analysis is applied in Chapter 5 to extract dynamic information, in Chapter 6, the temporal variances of the wavelet features of the cells are investigated to generate the foundation of distinguishing live and dead cells. In Chapter 7, some conclusions are drawn with respect to the advantages and disadvantages of different strategies of evaluating cell viability. In addition, the future perspective of this work is also discussed.
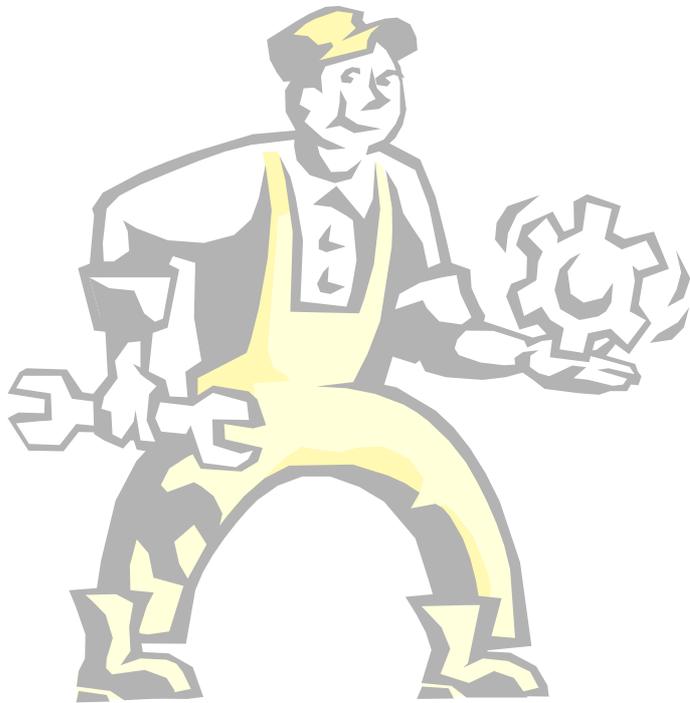
Chapter 2

Hardware Development

In order to monitor cell growth in bioreactors, an *In situ* **B**right field **M**icroscopy **P**robe (IBMP) for determining biomass properties has been devised in the Technical Chemistry Institute (TCI) of Hannover University, Germany (Fig. 2-1). It was termed originally *in situ* microscope, or ISM, but in this work it is referred to as IBMP for the convenience of comparison and description. In spite of its advantages in on-line visual observation of the cells and measurement of cell density, there are still technical obstacles to overcome in order to realize the monitoring of cell viability.

In this chapter an *In situ* **D**ark field **M**icroscopy **P**robe (IDMP) is proposed. It is constructed on the basis of the IBMP developed at TCI of Hannover University, Germany, which is similar to that one used by Joeris et al. (2002). The IBMP fits into a 25 mm standard port and has a retractable housing. The height of the sampling zone and the position of the objective lens can be adjusted by stepping motors during cultivation processes. A customizable program for sampling control and image acquisition has also been developed at TCI. In order to realize dark field microscopy, we have modified the illumination part of the IBMP. A dark field condenser has been developed to replace the normal condenser in the original design. In addition, the imaging system of the IBMP has also been improved for sharpening the images.

Selection of the Schemes

With the IBMP, some important properties of the cells can be determined, like cell size, cell density, etc.. A typical image captured with this kind of probe could resemble that shown in Fig. 2-2. However, due to bright field illumination, the pictures of the cells are of low contrast. What can be clearly seen in these pictures are rather the cell contours than the details of the interior structure. Because of this feature, no information about the cell vital status, namely viability (whether the cells are dead or living) and vitality (how vigorous the cells are), can be extracted.
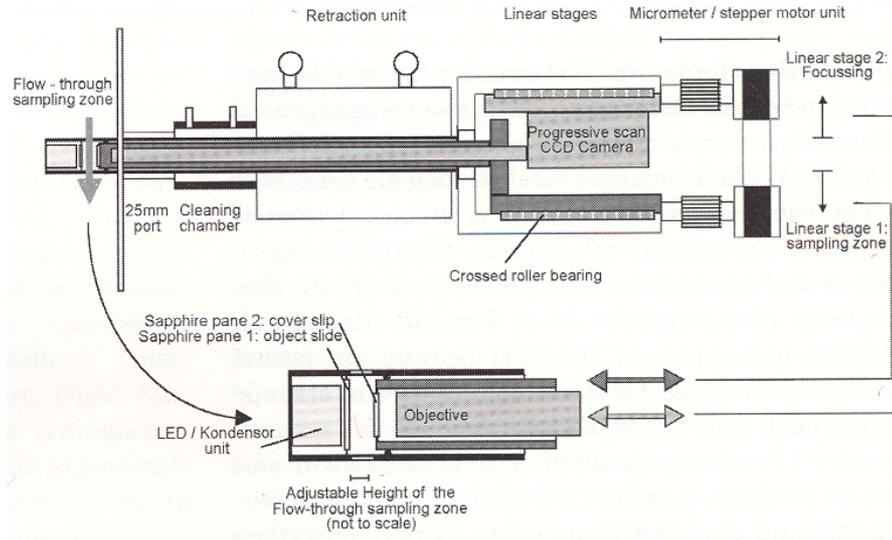
Fig. 2-1: Set-up of the microscopic sensor developed for determination biomass concentration in fermenters, from TCI, Hannover University.
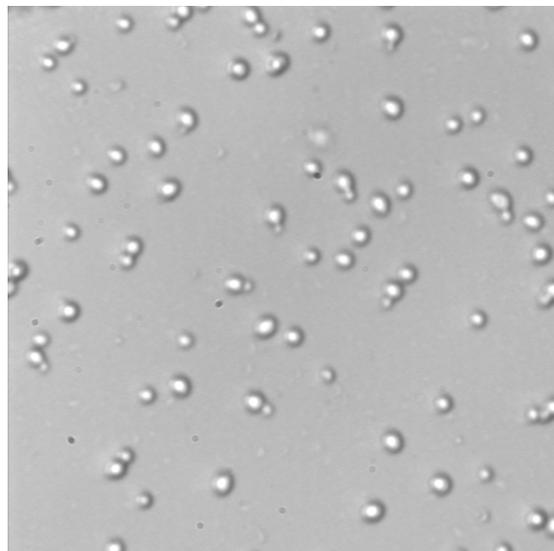
Fig. 2-2: An example of a picture of animal cells taken from the in-situ sensor developed by TCI, Hannover University. Courtesy of Guido Rudolph at TCI, Hannover University.

Despite the disadvantages of the IBMP, it can be improved with minor technical modifications to achieve the goal of measuring cell viability. In general, there are three microscopy-based techniques for increasing contrast of the images and revealing more intracellular information. These are phase contrast, DIC (Differential Interference Contrast) and dark field microscopy.

A typical set-up of a phase contrast microscope is presented in Fig. 2-3 Partially coherent illumination generated by the halogen lamp is directed through a collector lens and focused on a condenser annulus, which is positioned on the front focal plane of the substage condenser. The light collected by the condenser illuminates the specimen, and the resulting wavefronts can be split into two parts. While one part passes through the specimen non-deviated, the other is diffracted and retarded in phase by structures and phase gradients present in the specimen. These two parts of wavefronts are collected by the objective lens, and modulated differently with a phase plate on the rear focal plane of the objective lens, which leads to an interference pattern on the image plane.

Presented in Fig. 2-4 are two images obtained with bright field and phase contrast microscopy. While the subjects in bright field microscopy are regarded as *amplitude objects*, which attenuate the amplitude or intensity of the illuminating wavefronts, phase contrast microscopy is considered as *phase translator*, which translates minute variations in phase into corresponding changes in amplitude, which can be visualized as differences in image contrast.

In a typical DIC microscope (Fig. 2-5), polarized monochromatic light is split by a Wollaston prism into pairs of closely spaced, parallel, orthogonally polarized E (extraordinary) and O (ordinary) rays. The shear distance of the rays is below the resolving power of the objective. As the refractive index of the specimen with regards to E rays is different from that with regards to O rays, the pairs of rays undergo different phase changes, thus, a relative phase shift may be produced. The ray pairs are collected and recombined by the objective and a second Wollaston prism, and approach the analyzer. If there is no relative phase shift between the E and O rays, the recombined rays will be linearly polarized and blocked by the analyser; whereas, if there is some relative phase shift, the resulting rays will be elliptically polarized and pass through the analyser to the image plane, where they interfere with other transmitted rays producing an amplitude contrast image with bright phase-retarding objects on a dark background, as shown in Fig. 2-6.
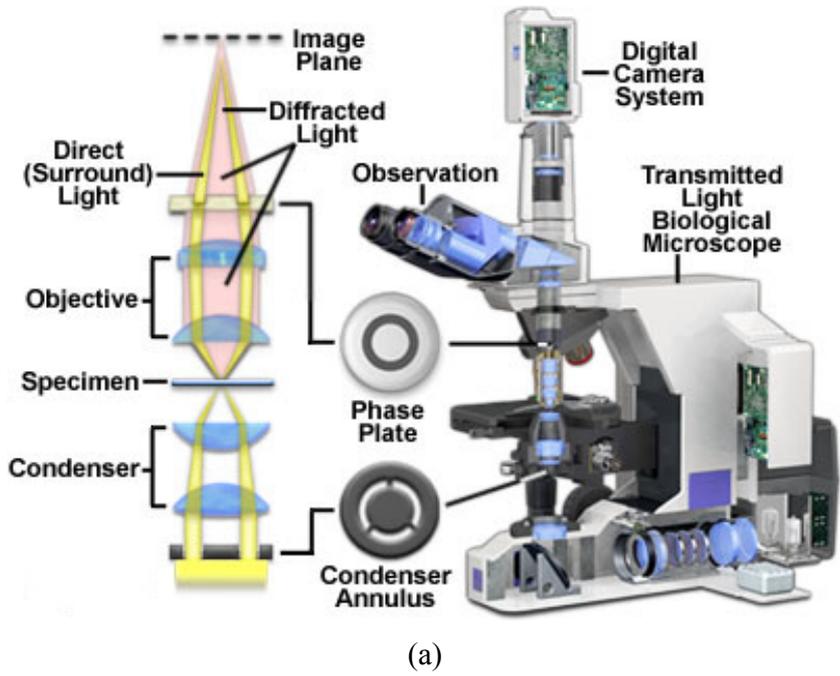
(a)
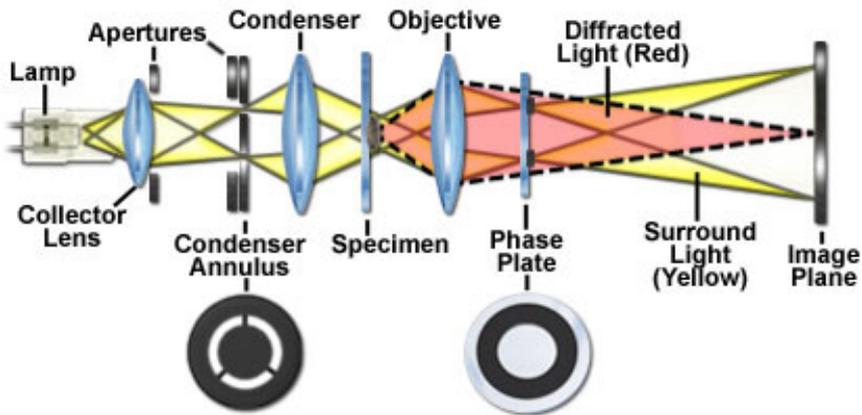
(b)

Fig. 2-3: Principle of phase contrast microscopy. (a) Configuration; (b) Optical train. (source: http://www.microscopyu.com/articles/phasecontrast/phasemicroscopy. html).
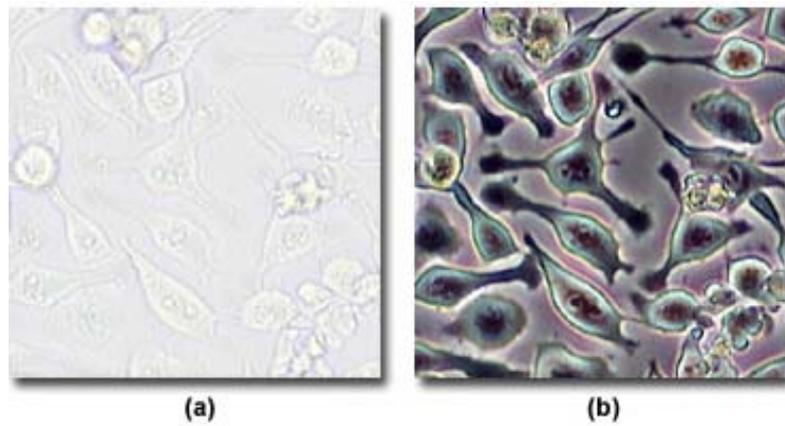
Fig. 2-4: Image of cells in (a) bright field and (b) phase contrast microscopy. The cells are human glial brain tissue grown in monolayer culture bathed with a nutrient medium containing amino acids, vitamins, mineral salts, and fetal calf serum. (source: http://www.microscopyu.com/articles/phasecontrast/phasemicro-scopy.html)
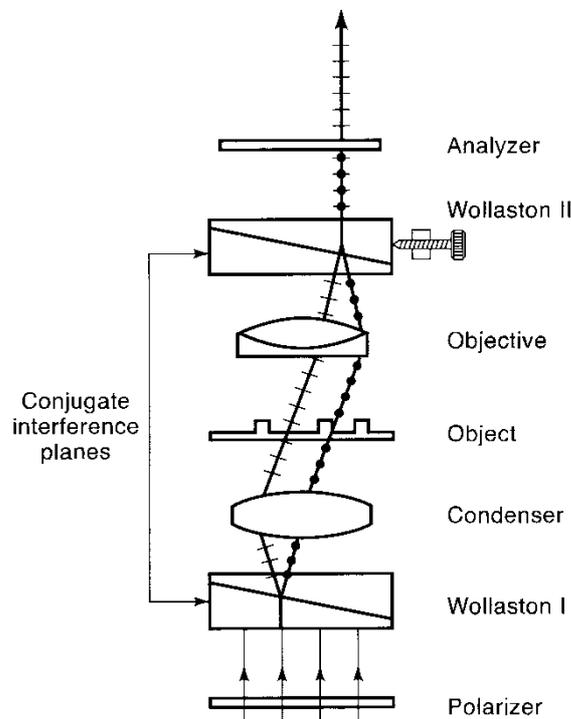
Fig. 2-5: Principle of a DIC light microscope (Heath, 2005).
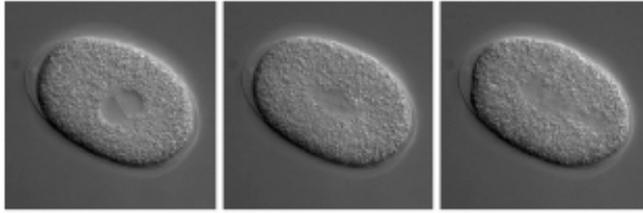
Fig. 2-6. An example of DIC microscope images of Caenorhabditis elegans embryos (Hamahashi et al., 2005) at different time points: (from left to right) 0, 160, and 320 seconds.

Following Heath (2005), the principle of dark field condensers is that it makes the illuminating light (non-diffracted) miss the aperture of the objective lens. On the contrary, it allows only diffracted light (from the target cells) to be collected by the objective lens. In this way, background of the images is made darker than the cells to enhance the contrast. Its principle is depicted in Fig. 2-7.
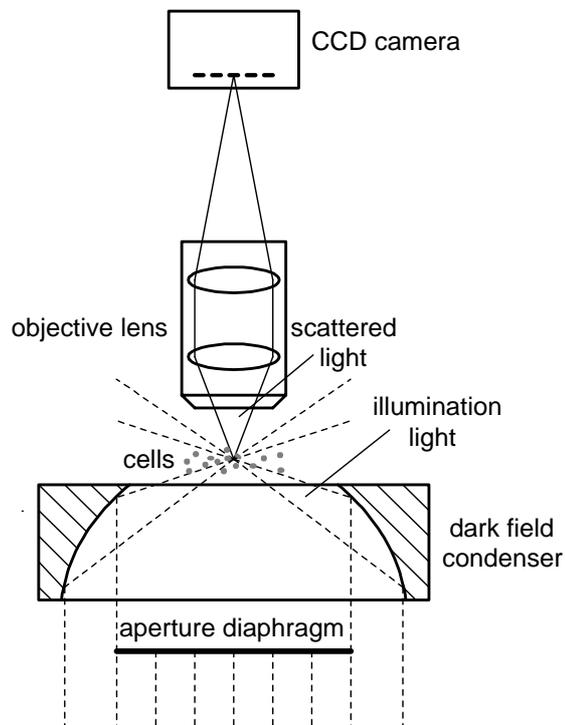
Fig. 2-7: Principle of dark field microscopy.

The parallel illuminating light from the collimating lens is segmented into two parts: in the middle, the light is blocked by an aperture diaphragm; while at the periphery, the light is focused by the interior surface of the dark field condenser, and cast onto the cells. Due to the large propagation angle, this part of light does not proceed into the objective lens directly. On the contrary, only the scattered light from the illuminated cells is able to enter the objective lens, which forms the image on the CCD pixel array. As a result, the images captured have a dark background and bright cells, thus enhancing the contrast of the images. Depending on magnification, not only the contour of the cells, but also some fine structures in the cells can be seen in these images. The realization of observing those structures may provide a means of inspecting the intracellular properties of the cells, which might provide valuable information for classification of live and dead cells (Wei et al., 2005).

It is clear that dark field microscopy is of less technical complexity compared to phase contrast and DIC microscopy, and is easier to be realized in an on-line probe. For this reason, we propose that a dark field microscopy probe (IDMP) should be developed to achieve our goal.

Structure of the IDMP

As shown in Fig. 2-8, the IDMP consists of an illumination unit, an optical imaging unit and a control unit.
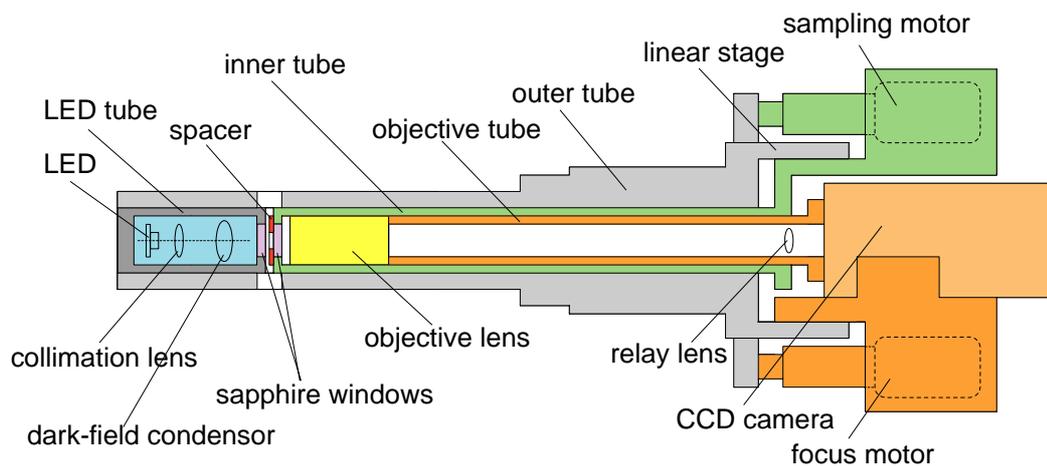
Fig. 2-8: Schematic diagram of the IDMP.

The light source of the illumination unit is an LED (Laser Emitting Diode) with a typical power of 340 mW and a divergence angle of 160 degree (LXHL-LR3C, Philips Lumileds Lighting Company, San Jose, USA. See Fig. 2-9). For the purpose of using a maximum of the power, a collimating lens (Fig. 2-10) is used to reshape the diverging beam from the LED into a parallel one. In order to focus the light beam we have designed a dark field condenser (Fig. 2-11), which is in principle similar to that one shown in the Figure 9.3A of a document at http://www.med.unc.edu/microscopy/Ch9_Dark_Field.pdf.

Fig. 2-9: The applied LED light source.

Fig. 2-10: Two views from different angles of the implemented collimator lens.

Fig. 2-11: The dark field condenser lens that has been designed and manufactured.
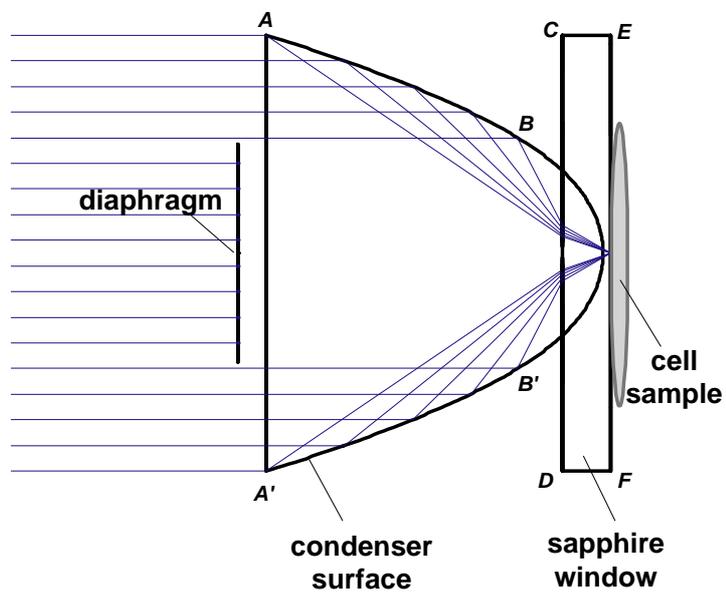
Fig. 2-12: Layout of the dark field condenser plus the other related elements in the Zemax design.

The interior surface of the dark field condenser is designed using a standard optical design software, Zemax® (Zemax Development Corporation, USA). We manufactured the condenser with aluminium alloy on a numerical control lathe. The layout of the dark field condenser plus the other related elements in the Zemax design is shown in Fig. 2-12. The diameter of the diaphragm that blocks the central rays of the incoming beam is 9 mm. The entrance (at the plane *AA'*) is 18 mm in diameter. The interior surface of the condenser (surface at *AB* and *A'B'*) is aspheric, it reflects and focuses the collimated beam to the rear surface of the sapphire window, on which the cell sample is located. The thickness of the sapphire window, namely, the length of line *CE* or *DF*, is 2cmm. The dark field illumination unit has been tested on a laboratory microscope (Fig. 2-13) and high contrast images have been achieved (Fig. 2-14).

Fig. 2-13: Set-up for the validation of the manufactured dark field illumination unit on a normal microscope.
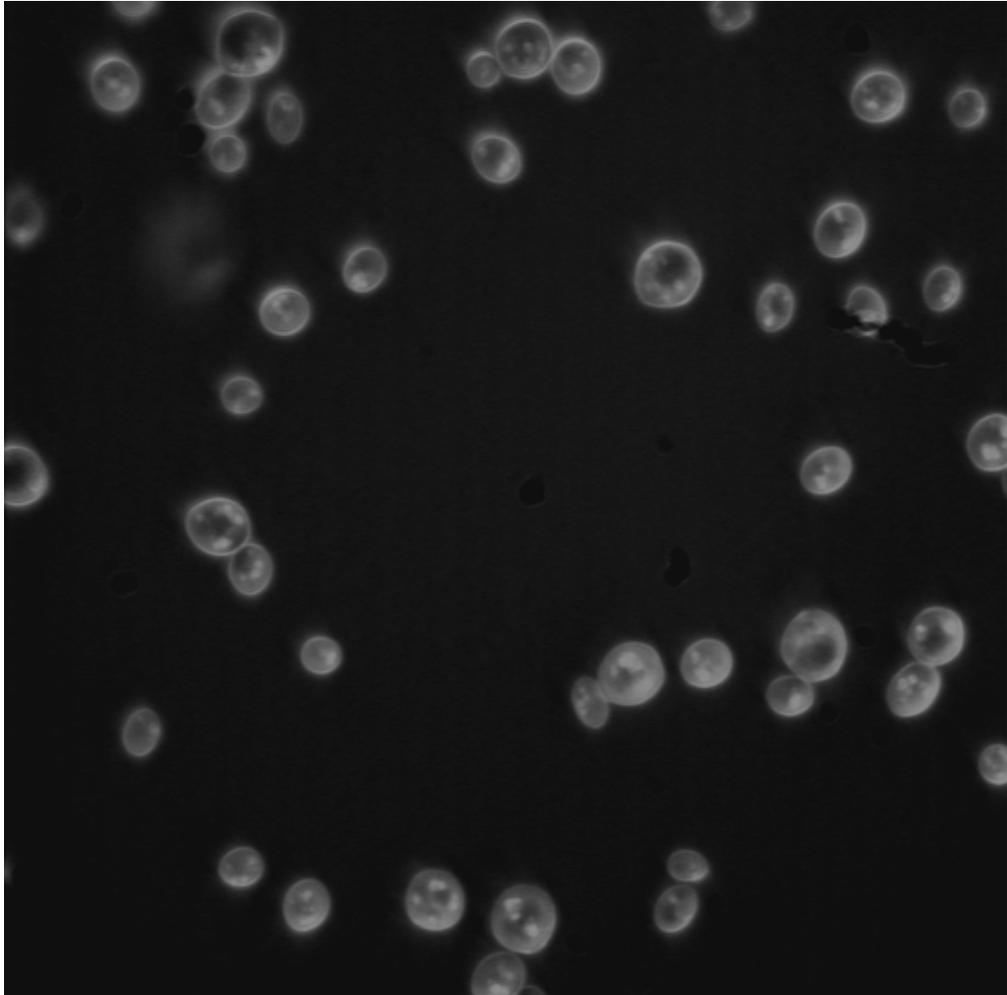
Fig. 2-14: The micrograph of the yeast, *Saccharomyces cerevisiae*, captured with the set-up shown in Fig. 2-13.

At both the exit of the illumination unit and the entrance of the optical imaging unit, there is a sealed sapphire window, which withstands high temperatures and maintains high transmission. Between these two sapphire windows is a sampling zone. A slit is made on the outer tube so that the cells can flow through it into the sampling region. A spacer is applied to form an isolated chamber to avoid large cell movements caused predominantly by the stirring in bioreactors, which might deteriorate the sharpness of the images. The thickness of the spacer has been chosen carefully. If it were too thick, multi-layers of cells

could have been caused, leading to low image quality. If it were too thin, it would be difficult to form an isolated chamber, because the spacer would get wrinkled when glued onto the sapphire window.

An objective lens, a relay lens and a CCD camera compose the optical imaging unit. The objective lens is a standard lens used in normal microscopes with a magnification of 10. It has a light-collecting angle of about 20°.
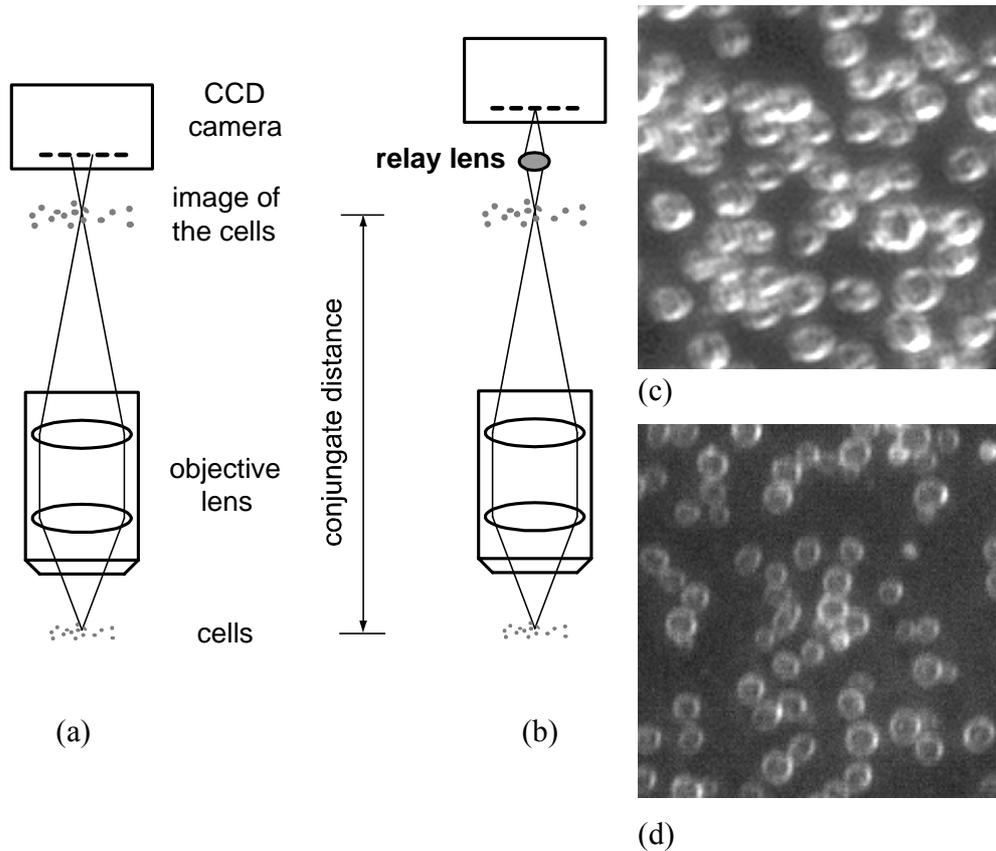
Fig. 2-15: Principle of image relaying. (a) old system without relay lens. (b) new system with a relay lens. (c - d) An image of the yeast, *Saccharomyces cerevisiae*, obtained in the absence (c) and in the presence (d) of a relay lens.

With the objective lens alone, no sharp images can be acquired, because the original design of the IBMP from TCI, Hannover Univerisity is not a standard layout. It is known that the conjugate distance (distance between the object plane and the image plane) is fixed for a certain standard objective lens, meaning that the distance from the sampling region (between two sapphire windows) to the pixel array of the CCD camera must be in principle the same as the standard one. However, the standard port on the bioreactor for holding the IBMP has a larger length than the conjugate distance. As a result, the distance between CCD camera and the sampling region had to be increased, or otherwise the sampling region could not have reached the culture broth in the bioreactor. This positional deviation of the CCD camera causes a certain blurring of the images. This is not a problem for the IBMP, since it is used mainly for cell counting, in which a little blurring will not lead to great difficulties. However, in the development of the IDMP, sharp images are required, as sharp images are supposed to provide more essential morphological details than blurred ones for evaluating cell viability.

The most straightforward way of solving this problem is image relaying, the principle of which is shown in Fig. 2-15. In the system without relay lens (Fig. 2-15(a)), image is formed a certain distance away off the pixel array of the CCD camera, which causes blurring of the image. In the new system, a relay lens is added between the CCD camera and the objective lens. In this way, the image is formed on the CCD pixel array by a second imaging via the relay lens.

At a first attempt, a singlet double convex spherical relay lens has been implemented in order acquire sharp images. This lens has a 1:1 zooming scale, and is shown in Fig. 2-16. An example of the micrographs of yeast cells using this singlet relay lens is shown in Fig. 2-17.
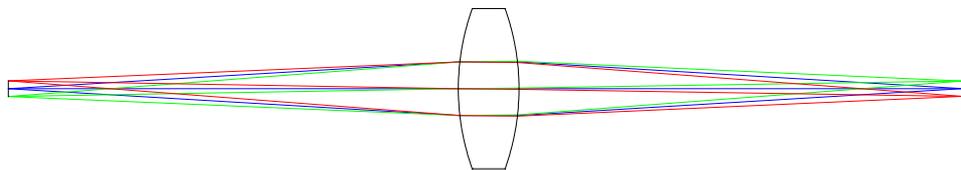
Fig. 2-16: The optical path of a singlet double convex spherical lens for 1:1 imaging.
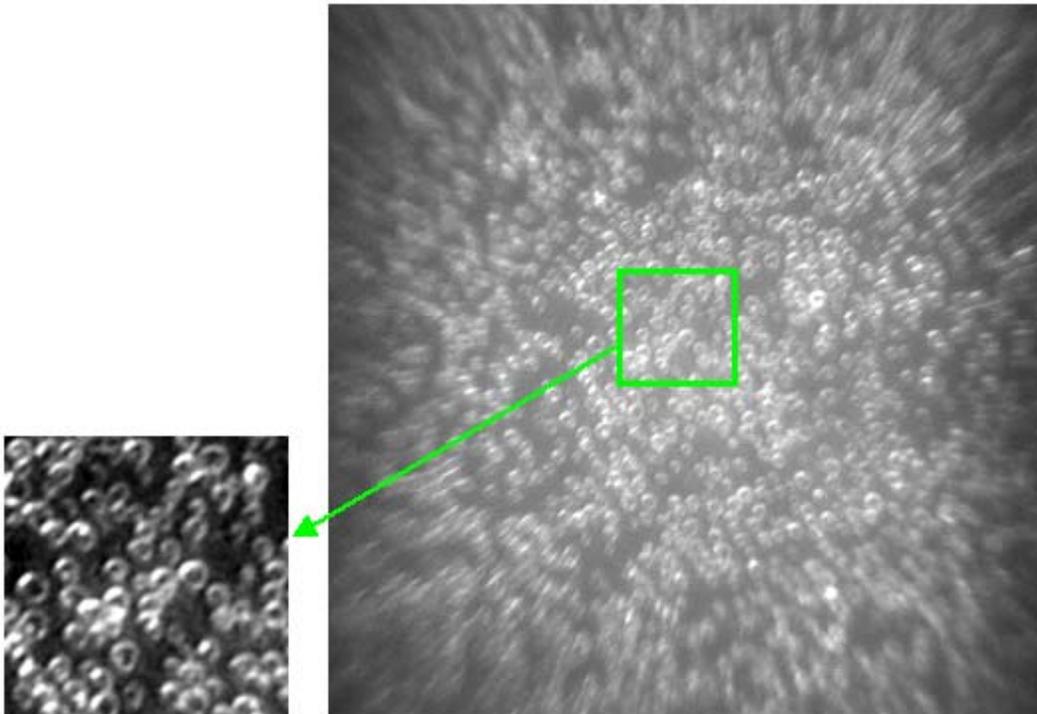
Fig. 2-17: The performance of the 1:1 imaging with a singlet double convex spherical lens: though in the center of the image the quality has been increased, however, the periphery is still blurred due to optical aberration.

It can be seen from Fig. 2-17 that the image is clear only at the central part, while the image quality degrades to a great extent at the periphery of the image as the optical aberrations of the system is greatly increased with the distance to the center of the image. In order to restrain the effective diameter of the lens, a small aperture (diameter $D = 2.3$ mm) is placed in front of the lens, and the micrograph turns into that one shown in Fig. 2-18. Though image quality in the center could be somehow improved, however, as the aperture size is too small (we could only find a standard gasket, which is fit for our system, with a diameter of the inner hole being 2.3 mm), the periphery of the image is "cut" completely.
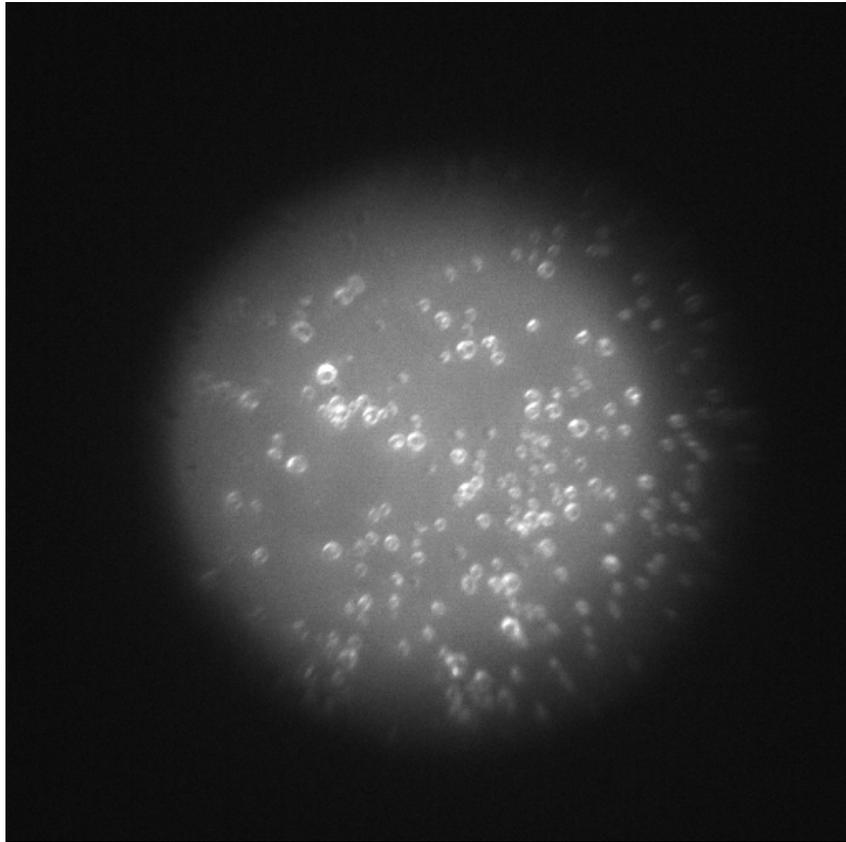
Fig. 2-18: The compromise of improving image quality and increasing the effective image size with a singlet lens as relay lens.

In order to achieve both image quality and effective size of the image, we have turned to using a multi-element imaging relay lens (Stock number: L45-761; Edmund Optics GmbH, Karlsruhe, Germany), which is designed exclusively for standard CCD applications. This relay lens has a magnifying factor of 1:1, and a focal length of 15 mm. Although this lens is well designed and manufactured, the implementation of it in the system is not straightforward, because the relay lens should be placed at a correct axial position, or otherwise the quality of the obtained images could not be sufficiently high. One example of the effect of a wrong axial position of the relay lens is shown in Fig. 2-19.
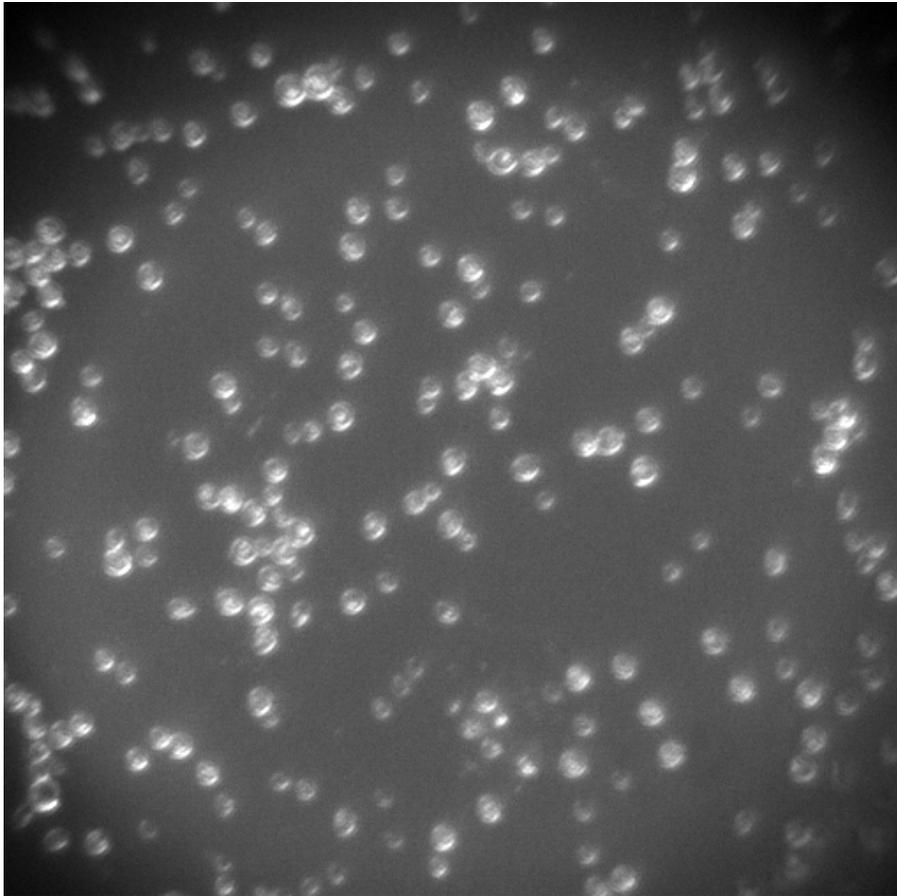
Fig. 2-19: Low-quality imaging with the multi-element relay lens from Edmund Company due to a non-proper axial position. The organism shown here is the yeast, *Saccharomyces cerevisiae*.

In order to position the relay lens properly, we have tried numerous axial positions of the CCD camera and the objective lens relative to the relay lens, and finally found out the best position. One of the resulting high quality micrographs is shown in Fig. 2-20. Compared with Fig. 2-19, it is seen that more details of the cells can be observed in Fig. 2-20. Even very tiny pieces of solids around the cells can be recognized clearly.
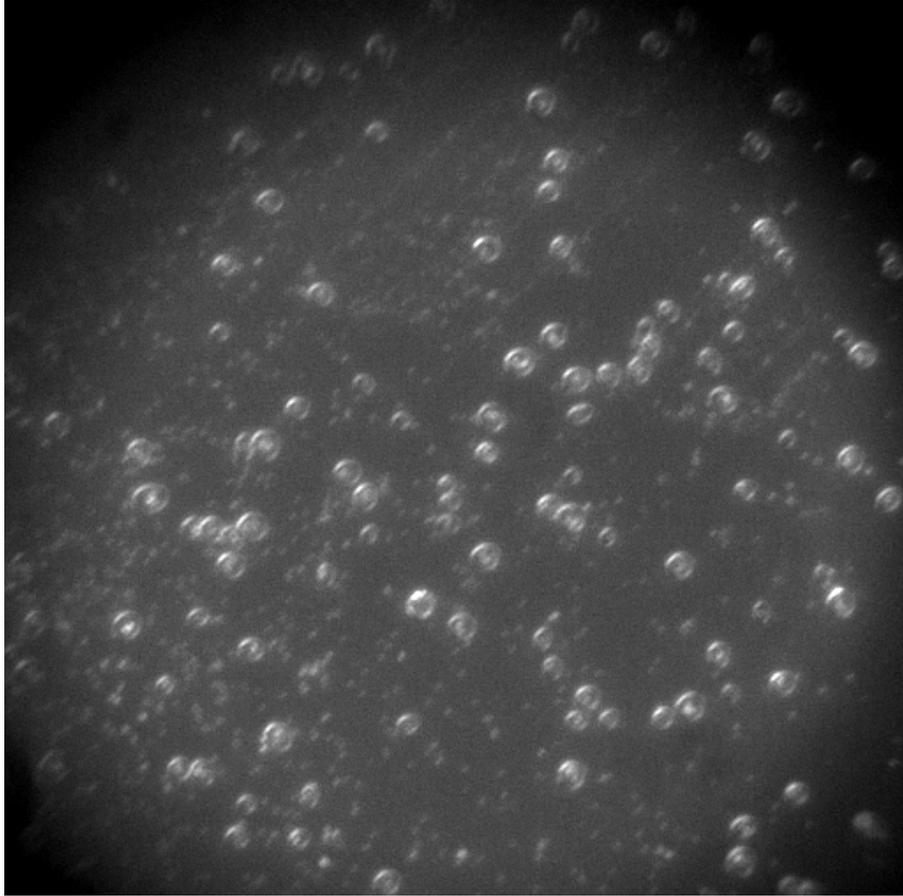
Fig. 2-20: An example image taken with the optimal layout of the relay lens. The organism shown here is the yeast, *Saccharomyces cerevisiae*.
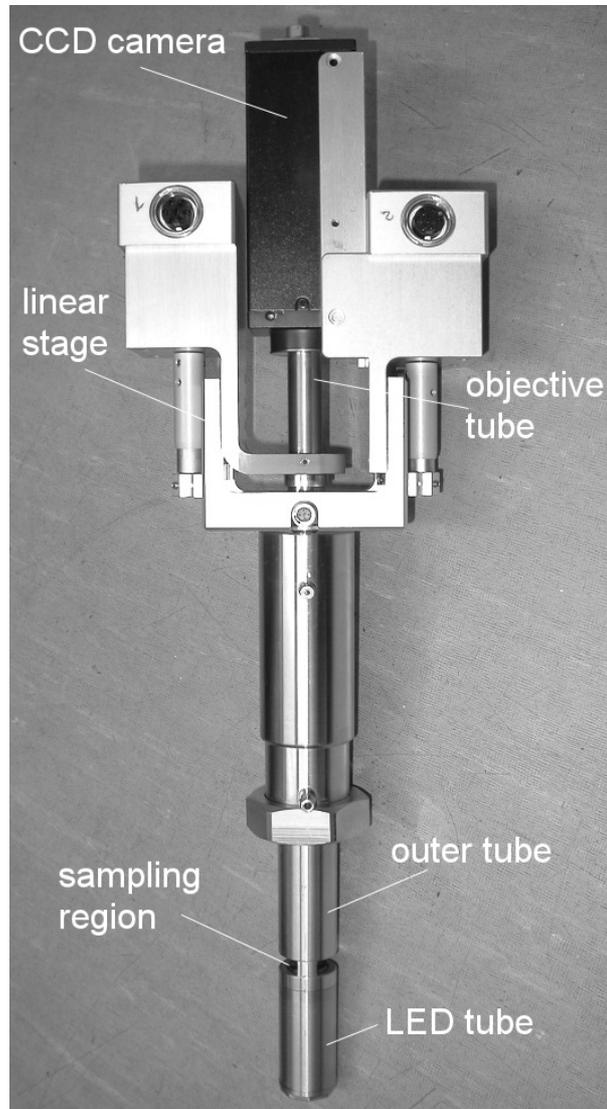
Fig. 2-21: The in situ dark field microscopy probe.

The CCD camera (XCD-X700, Sony Inc., Tokyo, Japan) is a black and white camera with a 1/2" chip. This camera delivers uncompressed, high-resolution digital 8-bit monochrome images. The typical output resolution of the XCD-X700 is 0.8 million pixels (1024 x 768) at 15 frames per second. An IEEE-1394 interface is integrated, thus, no frame grabber is needed. Normally a shutter time of 1/32 second has been used for taking images. High contrast images with dark background and bright cells are formed by the objective and relay lens on the CCD pixel array, which resemble those obtained with laboratory microscopes under dark

field settings. These high contrast images are sent to a connected personal computer for further analysis and processing.

The control unit of the IDMP consists of two stepping motors integrated in the IDMP. One of them is called sampling motor, responsible for the control of cell sampling; the other one, a focus motor, responsible for searching the optimal focus position of the objective lens. The movable parts are moving along two linear stages.

Fig. 2-21 shows the appearance of the IDMP. The CCD camera, motor-driven linear stages, objective tube, sampling region, outer tube and LED tube can be well identified referring to Fig. 2-8.

Automated Sampling Control and Image Acquisition

Since the stepping motors are controlled by the computer, the user can automate the process of cell sampling and image acquisition by a program (developed at TCI) with numerous customizations. More concretely, different parameters such as the brightness of the LED, frame rate of the camera, number of cycles, number of images taken in one cycle and time interval between two images can be defined.

One typical cycle of image acquisition in a batch process of cell monitoring is shown in Fig. 2-22. Before the cycle starts, the inner tube is located at a position of a certain distance from the front surface of the LED tube (sampling region open), and the cells can flow into the sampling region through the slit on the outer tube. When the sampling cycle starts, the inner tube is driven by the sampling motor and keeps on moving toward the LED tube, until the spacer touches the front surface of the LED tube, so that an isolated sampling chamber is formed (sampling region closed). In this way, the cells are separated from the outside culture broth, and will not be influenced by the turbulence caused by the stirrer in the bioreactor, which simplifies the subsequent image processing to a large extent. After the sampling region is closed, the objective motor drives the objective lens to the proper focus position. This position is predefined by the user through a prior search of the best image quality. It is normally fixed before a batch process, so that only the opening and closing of the sampling region are required for each cycle.
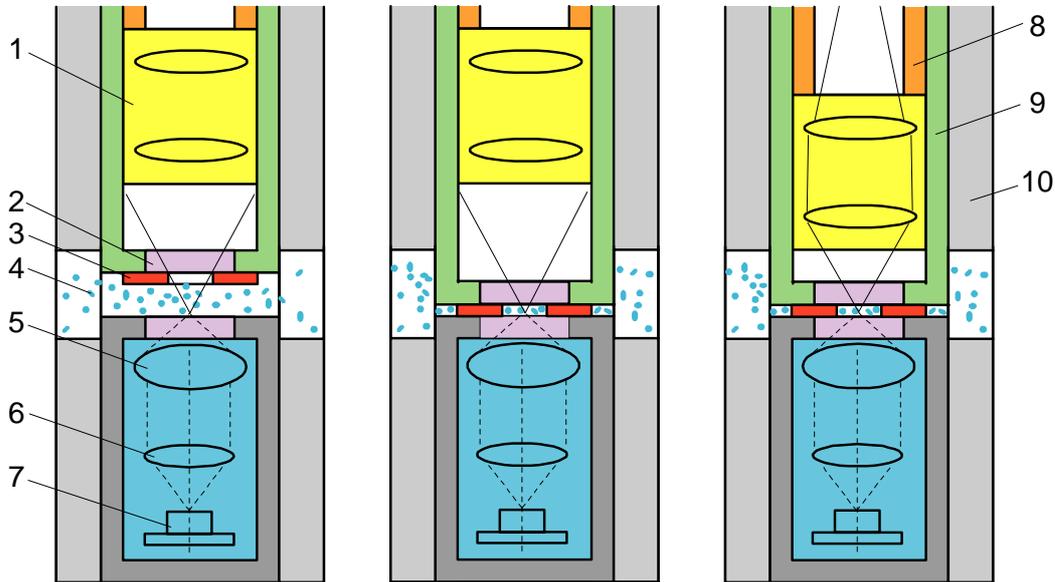
Fig. 2-22: Sketch of the operation principle of the IDMP in a typical cycle of cell sampling and image acquisition. 1 - objective lens, 2 - sapphire window, 3 - spacer, 4 - cells, 5 - dark field condenser, 6 - collimating lens, 7 - LED, 8 - objective tube, 9 - inner tube, 10 - outer tube.

Experimental Set-up

Since the IDMP is in part aiming at *in situ* evaluation of cell viability, growth experiments have been carried out firstly in a simple prototype bioreactor that consists of a 5 L cylindrical vessel and a four-leaf stirrer operating in the range of 0 to 1000 rpm. The diameter of the vessel is 180mm, and the height of it is 220mm. The length and averaged width of each leaf of the stirrer is 32mm and 15mm, respectively. In order to avoid the occurrence of too many air bubbles, the highest speed has been restricted to 300 rpm. This bioreactor is open and the cultures are exposed to the air. The whole set-up of the system is shown in Fig. 2-23. Brewer's yeast, *Saccharomyces cerevisiae*, is selected as the target microorganism. Before the *in situ* imaging experiments start, the yeast populations have been pre-cultivated in sterile flasks until a certain cell density is reached. After that, those populations are poured into the open vessel, and the IDMP is applied in the bioreactor to capture the images of the cells. These images are transferred to a personal computer for monitoring and further processing. The sampling process

can be well controlled using a program developed at the TCI, Hannover University, Germany.
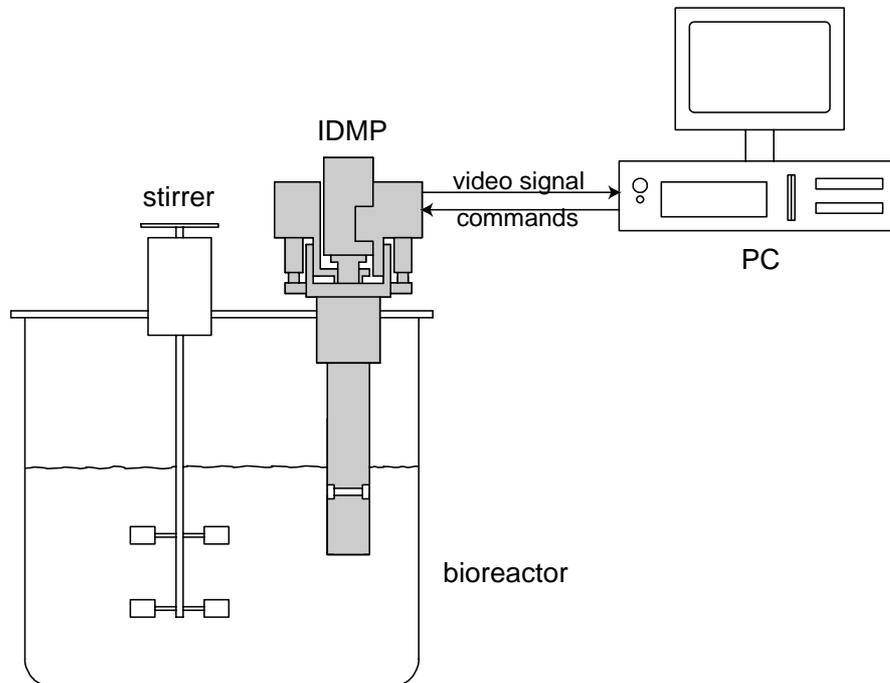
Fig. 2-23: Set-up of a prototype bioreactor equipped with the IDMP.

Example Images

Dark field images have been taken by the IDMP from yeast cultures at different cell densities, as shown in Fig. 2-24 and Fig. 2-25. These images resemble those obtained with normal laboratory microscopes under dark field settings. It can be seen that the contrast of the images are higher than those taken by an *in situ* bright field microscope (Joeris et al., 2002). Clear micrographs of some animal cells have also been captured with the IDMP at TCI, Hannover Univerisity, as shown in Fig. 2-26.
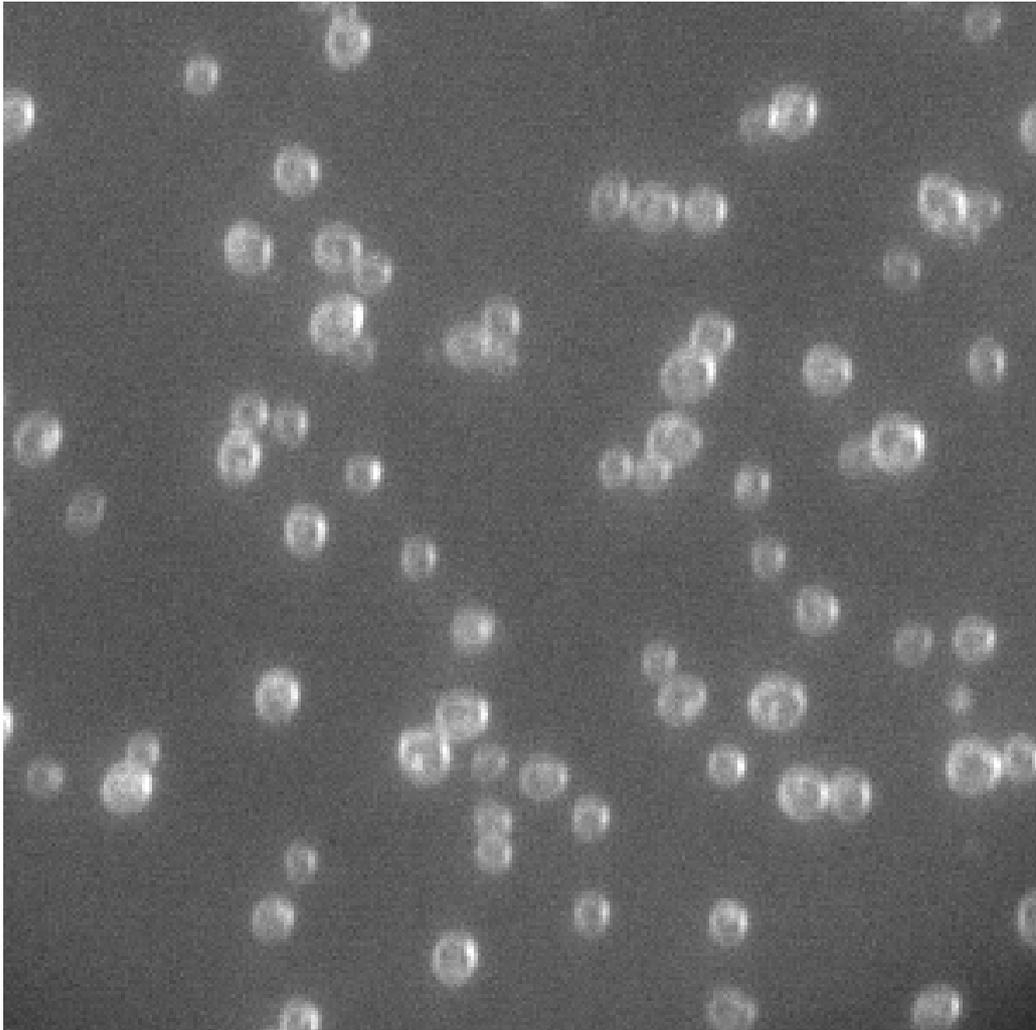
Fig. 2-24: A dark field image of the yeast, *Saccharomyces cerevisiae*, taken in the bioreactor by the IDMP at the cell density of $0.95 \cdot 10^{8}$ mL^{-1}.
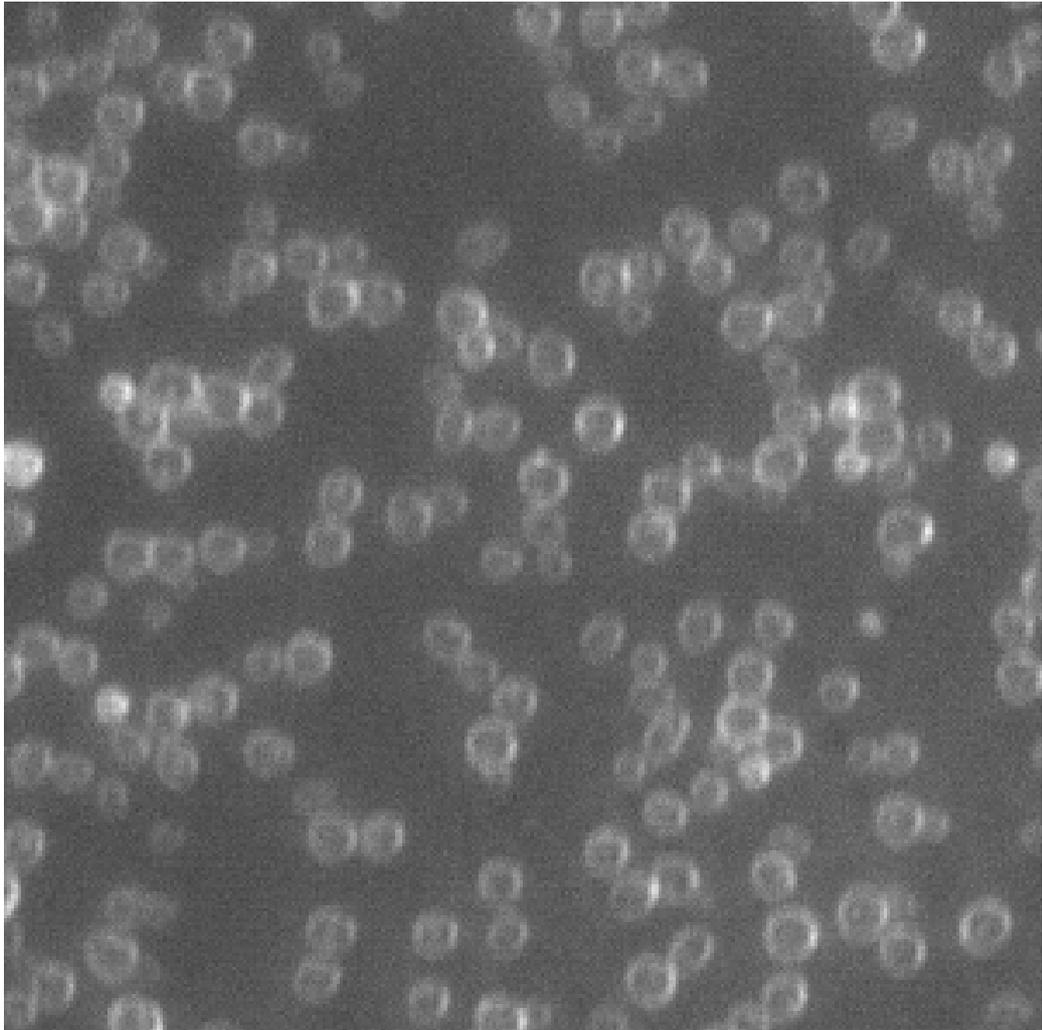
Fig. 2-25: Dark field image of the yeast, *Saccharomyces cerevisiae*, taken in the bioreactor by the IDMP at the cell density of $2.51 \cdot 10^{8}$ mL^{-1}.
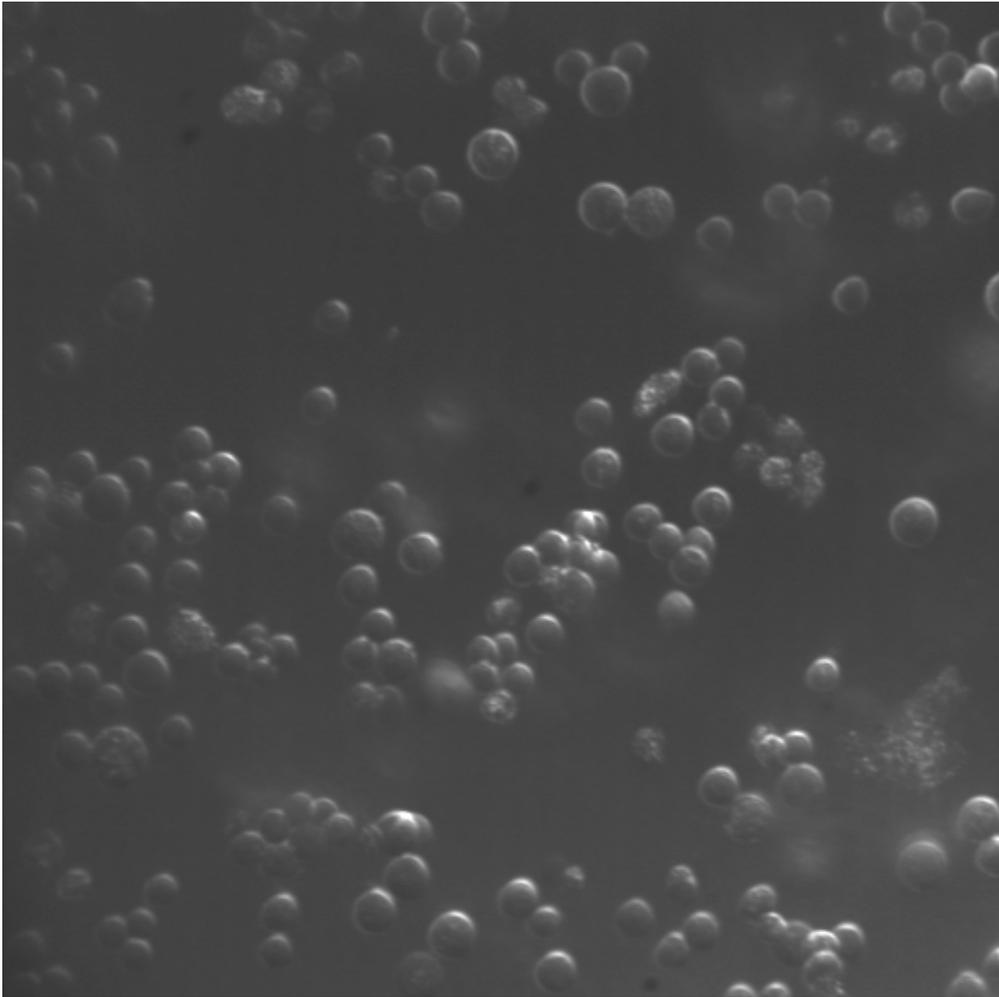
Fig. 2-26: A dark field image of some animal cells taken by the IDMP. Courtesy of Thomas Brückerhoff at TCI, Hannover University.

Abbreviations

DIC: Differential Interference Contrast
IBMP: *In situ* Bright field Microscopy Probe
IDMP: *In situ* Dark field Microscopy Probe
ISM: *In situ* Microscope
LED: Laser Emitting Diode
TCI: Technical Chemistry Institute

Chapter 3

Cell Detection and Viability Classification

To obtain the information about cell density and viability from the images recorded with the IDMP (*In situ* Dark field Microscopy Probe), Support Vector Machine (SVM) classifiers have been used to perform cell detection and viability classification. In the cell detection phase, examples of cell and background image patches are gathered and used to train an SVM classifier. In the cell viability classification phase, image patches of live and dead cells are used to train another SVM classifier. It will be demonstrated that cell density and viability in a bioreactor can both be correctly measured by means of the IDMP in conjunction with the SVM classifiers.

Material and Methods

Experimental Set-up

The same system set-up as introduced in Chapter 2, which includes a 5 L vessel and a stirrer as a prototype bioreactor, has been used. The speed of the stirrer ranges from 0 to 1000 rpm. In order to avoid the occurrence of too many air bubbles, the speed has been restricted to 300 rpm. This bioreactor is open and the cultures are exposed to the air. The IDMP is applied in the bioreactor to capture the images of the cells. These images are transferred to a personal computer for monitoring as well as a subsequent analysis.

Microorganism Strain

As a test organism, brewer's yeast, *Saccharomyces cerevisiae* (strain Tokay), is chosen as the subject of investigation. The advantages of brewer's yeast are fast reproduction and simple cultivation. YM medium (glucose: 10 g L^{-1}, peptone: 5 g L^{-1}, yeast extract: 3 g L^{-1}, malt extract: 3 g L^{-1}, pH 6.2±0.2) is used to cultivate the yeast. The media were autoclaved at 121ºC for 15 min.

Preparation of Various Types of Cultures

To automate image evaluation, we propose a special software design for determination of cell density and viability, which utilizes supervised machine learning for classification (The principle of the learning paradigm will be explained in posterior subchapters). Since the system is trained by a labelled image dataset, images are required, in which the physiological state (live or dead) of any of the cells is known. One straightforward way for obtaining such images is, to micrograph special cultures containing only live or dead yeast. Thus, the following protocols for preparing cultures that contain only live (all-live cultures) or dead (all-dead cultures) yeast have been applied. For preparing all-live cultures, yeast cells are grown aerobically at 25 °C in Erlenmeyer flasks (500 mL) on a rotary shaker (speed: 120 rpm, eccentricity: 25 mm), and harvested in the middle of the exponential phase (at the 36th hour). The filling volume is 20% of the volume of the Erlenmeyer flasks, namely, 100 mL. In order to obtain all-dead yeast cultures, yeast cells are incubated in a water bath at 70 °C for 2 hours. In some cases, starvation cultures are required for comparison with all-dead cultures. A starvation culture is prepared in following steps: first, yeast cells are harvested by centrifugation at 3,000xg for 5 minutes and washed three times in sterile deionised water, and then cells are resuspended to a final concentration of $3 \cdot 10^{8}$ mL^{-1} and incubated in sterile flasks for a couple of days until the viability of the culture decreases to a certain value, typically 60%.

The main difference between all-dead cultures and starvation cultures is that in the former case cells are killed due to high temperature or toxicity (normally with a viability of 0), while in the latter case, due to shortage of nutrients (normally with a viability between 0 and 100%).

The real viability of any of these cultures can be confirmed by means of a fluorescence stain, which will be introduced in the following subchapter.

Gold Standard of Cell Density and Viability Assessment

In order to validate the IDMP, it should be compared with other methods, especially with commonly used standard methods. A commercial fluorescence probe (FUN 1 cell stain, Invitrogen Ltd, Karlsruhe, Germany) for live/dead yeast viability evaluation is used as the gold standard for assessing the performance of the IDMP on viability determination. With FUN 1, live cells are marked clearly with orange fluorescent intravacuolar structures, while dead cells exhibit extremely bright, diffuse, green-yellow fluorescence, as shown in Fig. 3-1.
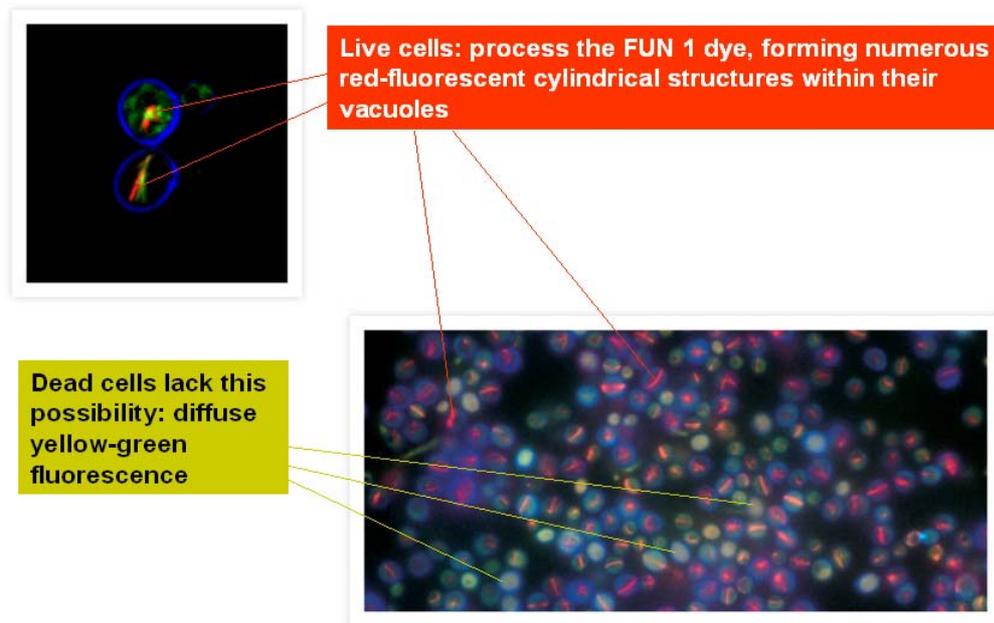
Fig. 3-1: The visual difference of the fluorescent stain, FUN 1, in live and dead cells.

The protocol of viability assessment with FUN 1 is:
1. Add FUN 1 stain to a yeast suspension at a concentration of 0.5 mM.
2. After incubating yeast for 30 minutes in a dark room, trap 10 μL of the yeast suspension between a microscope slide and coverslip.
3. Examine the stained yeast cells under a laboratory fluorescence microscope and assess manually the ratio of live to dead cells according to the distinguishing intracellular form and color of the fluorescence.

In this work, the used fluorescence microscope is Nikon Optiphot-2. In addition, in order to evaluate the IDMP's performance on determining cell density, a commonly used method, manual cell counting by hemocytometer (http://en.wikipedia.org/ wiki/Hemocytometer) is used.

Principal Component Analysis

Principal Component Analysis (PCA) is one of the most commonly used tools for dimensionality reduction, feature extraction and data analysis. The fundamental idea of PCA (Bishop, 1995) is to minimize the error in linear transformation of

vector $\mathbf{x}$ ($\mathbf{x} \in \Re^d$) from a d-dimensional space to an M-dimensional space where $M \ll d$.

Without loss of generality, note that $\mathbf{x}$ can be represented as a linear combination of a set of d orthogonal vectors $\mathbf{u}_i$

$$\mathbf{x} = \sum_{i=1}^{d} z_i \mathbf{u}_i$$

where the coefficients z_i can be found by

$$z_i = \mathbf{u}_i^{\mathrm{T}} \mathbf{x}$$

If a data set of K vectors $\mathbf{x}^n$ ($n = 1, 2\ldots, K$) is considered, and a dimensionality reduction from d to M is to be performed, each of $\mathbf{x}^n$ is approximated by an expression of the form

$$\widetilde{\mathbf{x}}^n = \sum_{i=1}^{M} z_i^n \mathbf{u}_i + \sum_{i=M+1}^{d} b_i \mathbf{u}_i$$

where b_i is independent to n, i.e., for all $\widetilde{\mathbf{x}}^n$, b_i are the same. The error in the vector $\mathbf{x}^n$ caused by the dimensionality reduction is given by

$$\mathbf{x}^n - \widetilde{\mathbf{x}}^n = \sum_{i=M+1}^{d} (z_i^n - b_i) \mathbf{u}_i$$

The best approximation is defined to be that which minimizes the sum of the squares of the errors over the whole data set. Thus we minimize

$$E_M = \frac{1}{2} \sum_{n=1}^{K} \left\| \mathbf{x}^n - \widetilde{\mathbf{x}}^n \right\|^2 = \frac{1}{2} \sum_{n=1}^{K} \sum_{i=M+1}^{d} (z_i^n - b_i)^2$$

The derivative of E_M with respect to b_i is

$$E_M'\big|_{b_i} = \frac{1}{2} \sum_{n=1}^{K} \left\| \mathbf{x}^n - \widetilde{\mathbf{x}}^n \right\|^2 = -\sum_{n=1}^{K} z_i^n + K b_i$$

If this derivative is set to zero, that yields

$$b_i = \frac{1}{K} \sum_{n=1}^{K} z_i^n = \mathbf{u}_i^{\mathrm{T}} \overline{\mathbf{x}}$$

where

$$\overline{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^{K} \mathbf{x}^n$$

Then the sum-of-squares error is rewritten as

$$E_M = \frac{1}{2} \sum_{i=M+1}^{d} \mathbf{u}_i^{\mathrm{T}} \Sigma \mathbf{u}_i$$

where

$$\Sigma = \sum_{n=1}^{K} \left(\mathbf{x}^n - \bar{\mathbf{x}} \right)\left(\mathbf{x}^n - \bar{\mathbf{x}} \right)^{\mathrm{T}}$$

The minimization of E_M with respect to $\mathbf{u}_i$ is found when $\mathbf{u}_i$ satisfy

$$\Sigma \mathbf{u}_i = \lambda_i \mathbf{u}_i$$

and the value of the error criterion at the minimum is in the form of

$$E_M = \frac{1}{2} \sum_{i=M+1}^{d} \lambda_i$$

Thus, the minimal error is obtained by choosing the *d-M* smallest eigenvalues, and their corresponding eigenvectors as the ones to discard.

Support Vector Machine

In recent years SVM has been proved to be powerful in the applications of pattern classification. The principle and applications of SVM can be found in a variety of publications (Cortes and Vapnik, 1995; Haykin, 1999; Rätsch et al., 2004; Long et al., 2006) since its invention in the 1990s.

Consider an *m*-element training set $\left\{ (\mathbf{x}_i, k_i) \right\}_{i=1}^{m}$, where $\mathbf{x}_i$ is the *i*-th input, and k_i is the corresponding desired output (i.e. class label). For simplicity, suppose that the subset labeled with $k_i = +1$ and that labeled with $k_i = -1$ are linear separable (a hyperplane can be found for discriminating two data sets having different class labels). The equation of the discriminating hyperplane can be expressed as

$$\mathbf{w}^T \mathbf{x} + b = 0 \tag{1}$$

where $\mathbf{w}$ is an adjustable weight vector, and b is a bias.

With a given problem, there could exist more than one hyperplane that can separate two classes from each other, and for any hyperplane defined in the form of Eq. (1), the distance from it to the closest data point is called the *margin of separation,* denoted by ρ. The goal of a support vector machine is to find the particular hyperplane so that the margin of separation is maximized. Under this condition, the found hyperplane is referred to as the *optimal hyperplane*. In Fig.

3-2 the construction of an optimal hyperplane in a two-dimensional input space is illustrated. The particular data points ($\mathbf{x}_i$, k_i) located exactly at the margin (i.e. closest to the decision surface) are called support vectors. These vectors play a prominent role in the operation of an SVM.
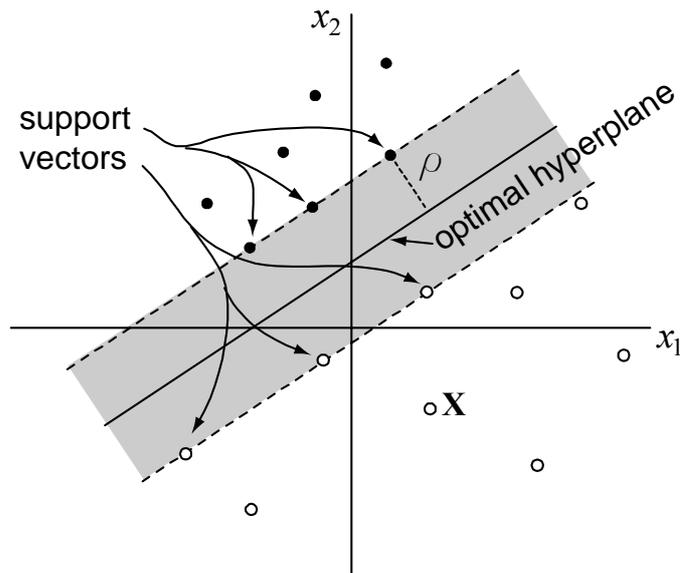
Fig. 3-2: Principle of Support Vector Machine: maximization of the margin of the separation.

The second main idea of SVM is the utilization of a so-called kernel function to transform the original data from the original space into a feature space. In this feature space, inseparable data become more separable depending on the applied kernel function. The most prominent is the Gaussian kernel that allows solving non-linear classification problems.

Image Processing

The image analysis process is composed of the following steps:
1. Image labeling.
2. Cell detection (determination of the positions of cells and cell density).
3. Cell viability classification (determination of the viability of the cells).

Image Labelling

The motivation for this work is to develop an imaging based probe, which automates the determination of cell density and viability in a bioreactor. To perform these two tasks, two SVM classifiers have been used. Both classifiers need to learn from labelled image data. Thus, all-live and all-dead images are collected and labelled manually using different colour to represent different type of objects. For instance, red (RGB code: 255, 0, 0) stands for live cells, green (RGB code: 0, 255, 0) for dead cells, yellow (RGB code: 255, 255, 0) for background. With these definitions, a red pixel placed on the images means that at this position a live cell is centred, while a green pixel labels the centre of a dead cell, and the meaning of a yellow pixel is that there is no cell at that position, or, it is background.

After labelling, the training sets can be obtained by picking out patches from the raw images with an N x N-sized window around the positions given by those labelling pixels. All three categories of image patches (live cell, dead cell and background) are considered differently for two classification tasks. In the task of cell detection, live and dead cell patches are combined as the training set for "cells" (positive set), and background patches are treated as the training set for "background" (negative set), naturally. While in the task of cell viability determination, live and dead cell patches are treated as positive and negative training sets, respectively.

Cell Detection

By means of SVM, cells in the images can be automatically recognized and counted, and then cell density can be determined by relating the number of cells to the volume of the isolated sampling chamber.

In the cell labelling phase, the raw images are labelled manually with pixels of different colours representing different kinds of objects, as outlined above. The N x N-sized patches around those positions are picked out as the items of the training sets. Each N x N-sized image patch can be depicted as a vector $\mathbf{P}_v \in \mathrm{R}^{N^2}$.

The parameter N should be chosen as a rough estimate of the average pixel diameter of the cells. In our applications, it is set to $N = 15$ for a typical image of 768 x 768 pixels. v denotes the index of the vectors. As mentioned above, live and dead cell patches are combined as the training set for "cells", hence, assigned a class label $k_v = 1$. Background patches are treated as the training set for "background", and assigned a class label $k_v = -1$.

The training set containing n_{train} pairs of (input, output) is denoted as

$$\mathbf{\Gamma}_{train} = \left\{ \left(\mathbf{P}_v, k_v \right) \right\}, \quad v = 1, 2, ..., n_{train}$$

The training set is composed of two subsets $\Gamma_{train} = \Gamma_{train}^{+} \bigcup \Gamma_{train}^{-}$, in which the positive subset (consisting of only cell (live and dead) patches) is interpreted as follows:

$$\Gamma_{train}^{+} = \left\{ \left(\mathbf{P}_{v^{+}}, 1 \right) \right\}, \quad v^{+} \in \left\{ v \mid \mathbf{P}_{v} \; is \; a \; cell \; patch \right\}$$

and the negative subset (consisting of only background patches) is interpreted as follows:

$$\Gamma_{train}^{-} = \left\{ \left(\mathbf{P}_{v^{-}}, -1 \right) \right\}, \quad v^{-} \in \left\{ v \mid \mathbf{P}_{v} \; is \; a \; background \; patch \right\}$$

Using these configurations, the SVM classifier is trained and a hyperplane is determined for distinguishing cells and background. The trained SVM classifier maps any input image patch to an output value y of a normalized output space, i.e. $y \in [0, 1]$. The value y can be regarded as the confidence whether the patch contains a cell or not. For this reason it is called confidence value.

To process an entire image the SVM is applied in a pixel-by-pixel manner: at each pixel, an $N \times N$-sized patch is read out as the input $\mathbf{x}$, and mapped with the classifier to a confidence value $y = C(\mathbf{x}) \in [0, 1]$. Thereby, at any pixel of the input image, a corresponding confidence value is given, and that forms a confidence map.

Let $\mathbf{p}$ be a pixel on the confidence map, $\mathbf{x}$ be the $N \times N$-sized patch centred at $\mathbf{p}$, $\mathbf{x'}$ be an $N \times N$-sized patch within a distance of $N/2$ from $\mathbf{x}$, and $C(\mathbf{x})$ be the confidence value of $\mathbf{x}$, then $\mathbf{x}$ is regarded as a cell (or a cell is supposed to be found at $\mathbf{p}$), if

$$\left[C(\mathbf{x}) > t \right] \wedge \left[C(\mathbf{x}) = \max_{d(\mathbf{x},\mathbf{x'}) < (N/2)} \left\{ C(\mathbf{x'}) \right\} \right] \qquad (2)$$

In other words, if pixel $\mathbf{p}$ has a local maximal confidence value within a radius of $N/2$, and this value is greater than a threshold t, it is considered as a centre of a cell.

The cell detection results can be divided into three groups:

TP (True Positive): The item is positive and classified as positive, namely, a cell position is correctly found.

FP (False Positive): The item is negative but classified as positive, namely, a background is falsely classified as a cell.

FN (False Negative): The item is positive but classified as negative, namely, a cell falsely reported as background.

Two ROC (Receiver Operator Characteristics) values (Nattkemper et al., 2003) are computed to evaluate the cell detection results:

Sensitivity (SE): The proportion of real cells that are correctly recognized. It is depicted as:

$$SE = \frac{TP}{TP + FN}$$

Positive Predictive Value (PPV): The proportion of detected cells that are real cells. It is calculated as:

$$PPV = \frac{TP}{TP + FP}$$

There is a threshold parameter t in Eq. (2), which gives the minimal confidence value that any pixel has to possess to be considered as a cell position. Therefore this threshold plays an important role in cell detection.

According to Eq. (2), a position is regarded as the center of a cell, only when two prerequisites on its confidence value are satisfied: being a local maximum and greater than the threshold. For all the local maxima on the confidence map, the lower is the threshold, the more will be regarded as cells. For illustration we consider the extreme case of a threshold $t = 0$: all the local maxima are counted as cell positions. On one hand, TP reaches a high value, meaning that the system will find most cells; on the other hand, such a value will lead to the largest number of FP. As a result, a high SE and a low PPV is observed in this case, as shown in Fig. 3-3 (a). When the threshold is increased, more local maxima will be screened out, thus, FP is considerably reduced (See Fig. 3-3(b)). When the threshold is further increased, more FN appear, and the results turn out to be more conservative, as shown in Fig. 3-3 (c). The situation shown in Fig. 3-3(b) is considered as the best, because not only TP is high, but also FP and FN are low, namely, both SE and PPV are high.

Faster Cell Recognition through PCA

In the above subchapters a cell detection scheme has been described, in which raw $N \times N$ sized patches are used directly. For instance, with $N = 15$, the dimensionality of the datasets is 225, which leads to an excessive computational expense. In order to process the images faster, a PCA is performed to reduce the dimensionality. First, the principal components, the variances, as well as the transform matrix from the old coordinate system to the new one are determined for the positive training set containing cell patches. Thereafter, the principal components that show most of the variances of the data are selected to represent the original data. Normally, the number of the selected principal components (say, 10) is much less than the original dimensionality of the data, thus, the dimensionality can be consequently reduced. In the classification phase, each patch of the cells or the background is

processed before it is used by the classifier with the transform matrix that is determined through PCA. In this way, the computational expenses of classification is greatly reduced due to a lower dimensionality. This scheme is illustrated in Fig. 3-4. The gray blocks are modules corresponding to the PCA operation and dimensionality reduction. The whole process can be divided into training and testing phases.

(a)

(b)

(c)
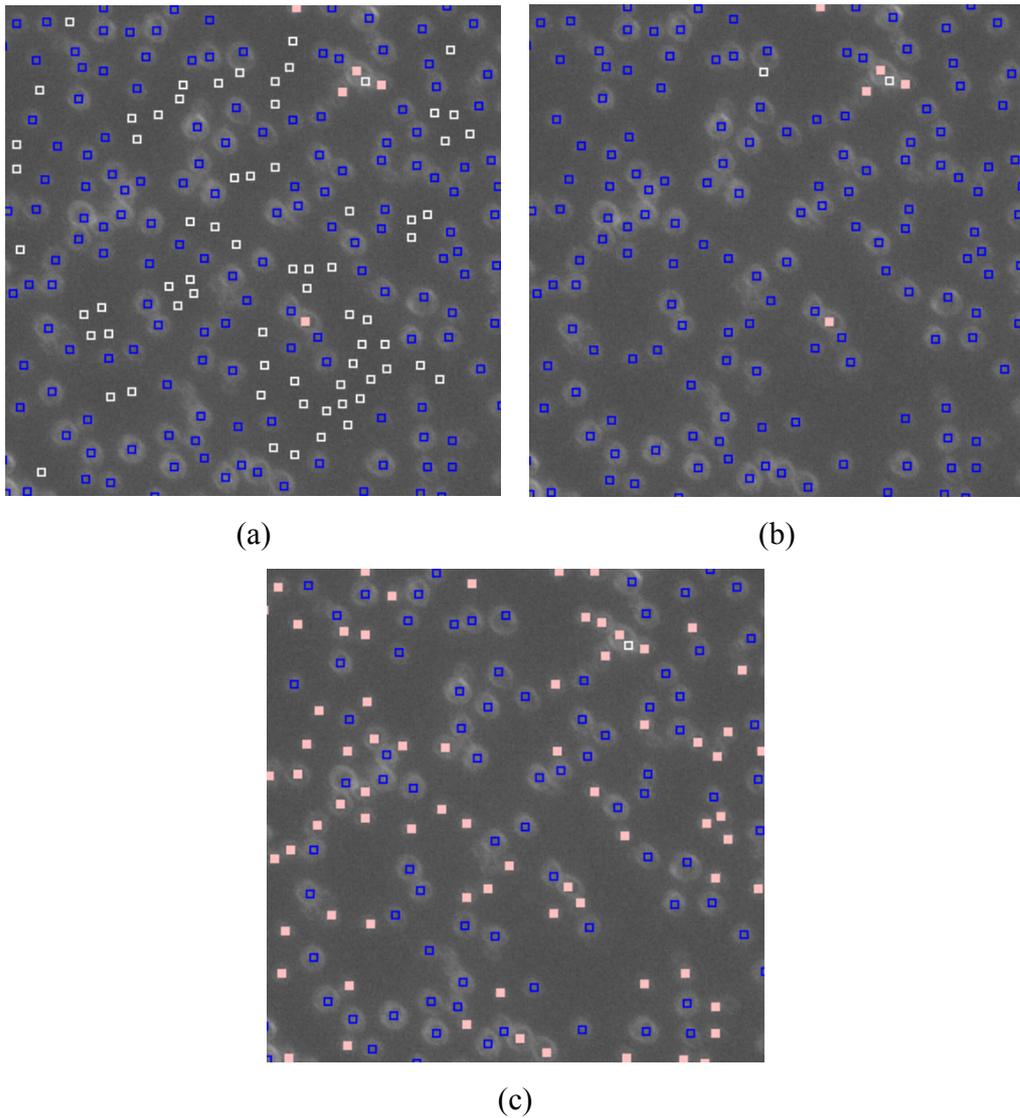
Fig. 3-3: Cell detection at (a) threshold t = 0.1; (b) t = 0.5; (c) t = 0.7. Blue hollow squares: True Positive; white hollow squares: False Positive; pink filled squares: False Negative.
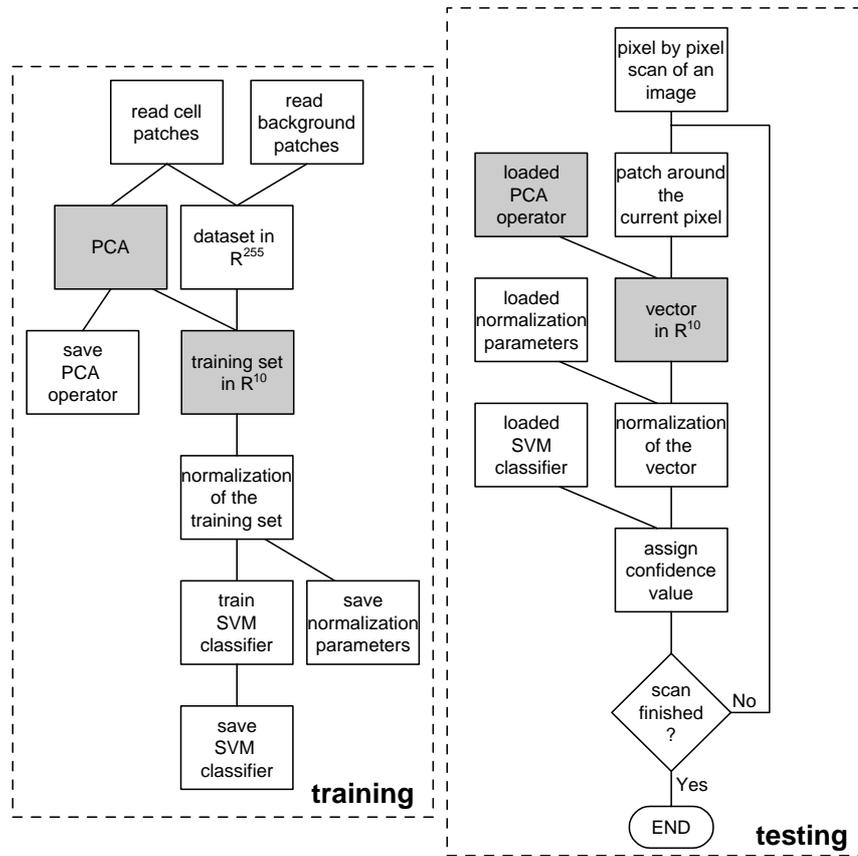
Fig. 3-4: Workflow of the cell counting program.

Cell Viability Classification

In order to obtain a system for automated determination of cell viability, the second SVM classifier is required to be trained to classify cells into live and dead cells. To this end, yeast cultures composed of only live or only dead cells are used for training the system.

As mentioned, in the phase of determining cell density, all the cell positions can be found using the confidence map. With this automated process, the training set of live or dead cell patches is generated: In an image containing only live cells, the cell-detector is applied to find cell positions. Afterwards, N x N-sized patches are picked out and added to the positive training set with a class label "1". Similarly, dead cell patches are added to the negative training set with a class label "-1". Thus, the whole training set can be interpreted as follows (suppose there are n'_{train} training cells):

$$\Psi_{train} = \left\{ \left(\mathbf{V}_\omega, u_\omega \right) \right\}, \quad \omega = 1, 2, ..., n'_{train}$$

The whole training set can also be divided into two subsets $\Psi_{train} = \Psi^+_{train} \cup \Psi^-_{train}$, with positive subset

$$\Psi^+_{train} = \left\{ \left(\mathbf{V}_{\omega^+}, 1 \right) \right\}, \quad \omega^+ \in \left\{ \omega \mid \mathbf{V}_\omega \text{ is a live cell patch} \right\}$$

and the negative subset

$$\Psi^-_{train} = \left\{ \left(\mathbf{V}_{\omega^-}, -1 \right) \right\}, \quad \omega^- \in \left\{ \omega \mid \mathbf{V}_\omega \text{ is a dead cell patch} \right\}$$

In consequence, the SVM classifier is trained to map any tested cell patch $\mathbf{V}$ to a value $u \in \{-1, 1\}$. It gives the class label "1" to a patch that resembles a live cell, and "-1" to a patch that resembles a dead cell.

If a ground truth is available, the viability classification results of one run can be divided into four groups:

TP (True Positive): A live cell is classified correctly as live.
FP (False Positive): A dead cell is classified falsely as live.
TN (True Negative): A dead cell is classified correctly as dead.
FN (False Negative): A live cell is classified falsely as dead.

To assess the classification accuracy, we compute again the following ROC values:

Sensitivity (SE): The proportion of live cells correctly classified as live:

$$SE = \frac{TP}{TP + FN}$$

Specificity (SP): The proportion of dead cells correctly classified as dead:

$$SP = \frac{TN}{TN + FP}$$

Results and Discussion

Accuracy of Cell Density Measurement

A training set containing 783 background patches and 312 cell patches from 4 training images (2 all-live and 2 all-dead) was used for cell detection in 16 tested images (4 all-live, 4 all-dead, 8 mixed) showing in total more than 900 cells. The plot of threshold (t, see Eq. (2)), SE and PPV is displayed in Fig. 3-5. It can be seen that when t ranges from 0.1 to 0.4 (point A), too much background is regarded as cells, yielding a low PPV (0.71). The best threshold is supposed to be 0.5 (Point

B), because both SE and PPV are high (0.95 and 0.98). A threshold of 0.6 (point C) might increase the PPV to some extent (0.99), however, at the cost of a considerable reduction of SE value (0.90). When the threshold is greater than 0.7 (Points D, E, F), the SE decreases too much, which makes the classifier more conservative and less practical.
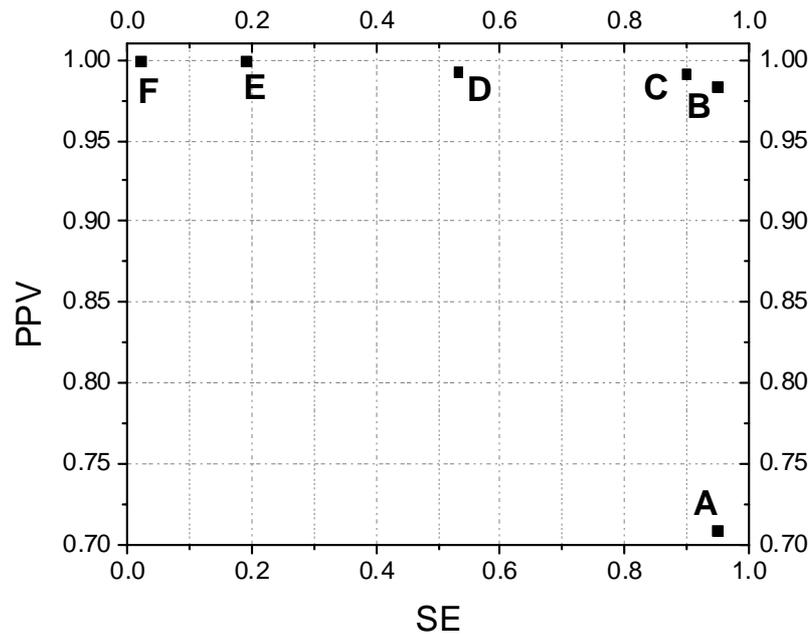
Fig. 3-5: Evaluation of the cell detection results at an SE-PPV plot. A: threshold $t = 0, 0.1, 0.2, 0.3, 0.4$; B: $t = 0.5$; C: $t = 0.6$; D: $t = 0.7$; E: $t = 0.8$; F: $t = 0.9$.

With a proper selection of the threshold ($t = 0.5$), the cells in each image can then be counted automatically. Manual counts using a hemocytometer are applied in parallel for evaluating the results. Cell samples were taken out of the yeast cultures (ranging from $0.3 \cdot 10^{8}$ mL^{-1} to $3.5 \cdot 10^{8}$ mL^{-1}) and counted with a hemocytometer under a laboratory microscope. For each culture, five manual counts are performed, and the average is taken as gold standard.

In comparison, for each of these yeast cultures, three images are taken by the IDMP and the number of cells is computed. A variance of ± 10 cells/image has been observed. By averaging, an estimate of cell quantity is then obtained. The density is computed based on the volume of the chamber obviously. To determine this value, a linear regression program is applied to survey the correlation between the

cell densities determined by the IDMP and those determined by hand count using hemocytometer. In Fig. 3-6, a linear relationship can be identified, and the slope of the straight line indicates the volume of the IDMP's sampling chamber, and it is determined as $3.98 \pm 0.04 \cdot 10^{-7}$ mL. The correlation factor R^2 is 0.983, showing a good performance of cell density determination.
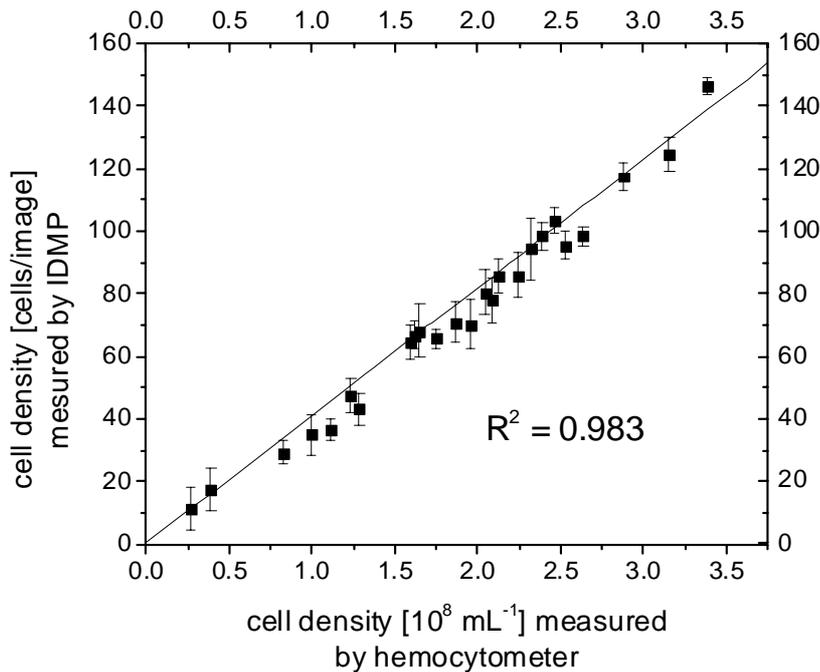
Fig. 3-6: Correlation of the cell density measured by hand counts and by the IDMP.

Fast Cell Detection by Means of PCA

As described above, in order to speed up the cell detection procedure, a PCA can be performed in advance to reduce the dimensionality of the datasets. In this scheme, an advanced method of collecting negative training set (background patches) is used, which extracts background patches in the regions that are a certain distance (say, 7 pixels) away from those hand-labeled cells. The applied background to cell ratio is $m = 10$. In this way, a training set that contains 312 cells and 3120 background (all are of 15 x 15 pixels) is generated, and a PCA is then performed on the 312 cell patches.

PCA computes eigenvectors and eigenvalues of the covariance matrix of 312 vectors in $\Re^{225}$. Fig. 3-7 shows the first 10 principle components (PC's) that have the highest eigenvalues in descending order. From the accumulated fraction curve, it can be seen that a majority (about 85%) of the variance in the image data is covered by the first 10 PC's. Thus, these 10 PC's can be taken for re-presenting the data points in the $\Re^{10}$ space formed by the corresponding eigenvectors. According to these results, all the data points in $\Re^{225}$ (original cell patches) can be transformed into a $\Re^{10}$ space formed by the eigenvectors of the first ten PC's that have the highest eigenvalues, so that the dimensionality is reduced from 225 to 10.
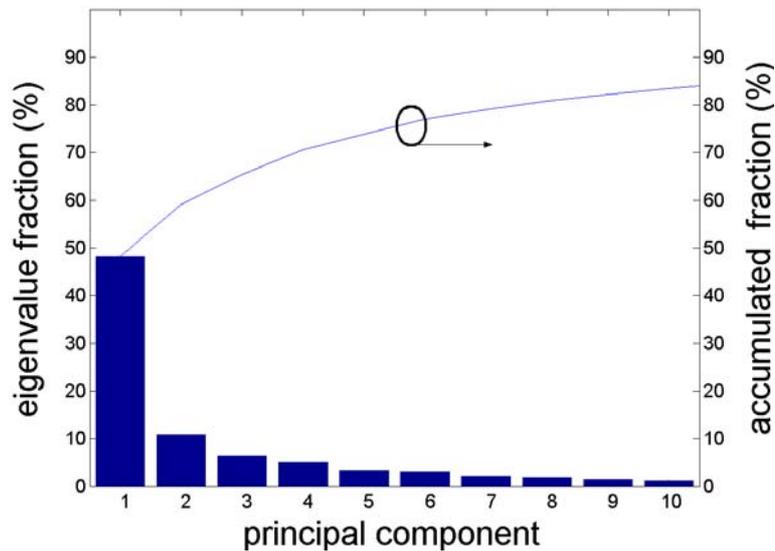
Fig. 3-7: Variance of each principal component of the vectors of the cell patches.

The eigenvectors of the first 10 PC's can be visualized in the form of eigencells, as seen in Fig. 3-8. Any original images of yeast cells can be depicted as the linear combination of these eigencells.

The size of each micrograph is 768 x 768 pixels. Without PCA processing, it corresponds to about 580,000 data points in a $\Re^{225}$ space. Processing such a huge, high-dimensioned dataset requires a great deal of computation effort. Without dimensionality reduction, the counting process takes in average 617 seconds on a laptop (operating system: Suse Linux professional 9.2; CPU: Pentium IV 2.8 GHz; memory: 512 MB). In comparison, after dimensionality reduction, the computation time is remarkably reduced to 105 seconds, only one sixth of the former, without impairing the cell detection accuracy. Apart from that, another benefit of PCA

processing is that an SVM classifier with linear kernel can be used to generate the confidence map. In this way the complicated process of choosing the appropriate parameter of a Gaussian kernel can be avoided.

| 1st eigencell | 2nd eigencell | 3rd eigencell | 4th eigencell | 5th eigencell |
| 6th eigencell | 7th eigencell | 8th eigencell | 9th eigencell | 10th eigencell |
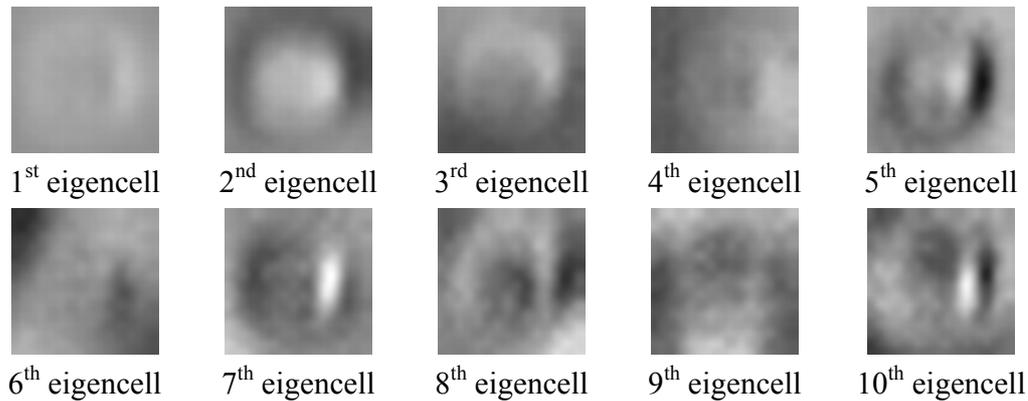
Fig. 3-8: Visualization of the eigencells.

Accuracy of Viability Classification

There are three scenarios of evaluating IDMP's performance on viability assessment:
1. Tests in all-live or all-dead cultures.
2. Tests in mixtures of all-live and all-dead cultures
3. Tests in mixtures of all-live and starvation cultures.

   In the first scenario, the classification accuracy can be evaluated on each single cell in the images, so that quantities like TP, FP, TN, FN, SE, SP can be determined; while in the other two scenarios, only viability of the whole culture (not of any single cell) can be compared with the viability reported by FUN 1 stain.

Scenario I: Viability Classification of All-live or All-dead Cultures

In this scenario, cultures containing only live or dead cells (It has been confirmed in advance that they are really all-live or all-dead cultures with FUN 1 stain) are tested. The IDMP is applied in either all-live or all-dead cultures, and gives its determination of viability of each cell in the images.

   Using a training set that contains 615 live cells (from 8 all-live images) and 627 dead cells (from 8 all-dead images) to train the SVM classifier, and applying this trained classifier to a test set that contains 1332 live cells and 1431 dead cells (from 16 all-live images and 16 all-dead images), an ROC (Receiver Operator Characteristics) curve (Nattkemper et al., 2001&2003) can then be drawn showing

the performance of the IDMP, as shown in Fig. 3-9. In this figure, different data points represent different thresholds of the distance from the vectors to the discriminating hyperplane. These thresholds control the fractions of cells that are classified into different classes. The ROC curve is very close to the ideal, as the area under the curve is 0.99, approximately equals the ideal value, 1.0. That means for all-live or all-dead cultures, the IDMP can give nearly perfect determination of viability.
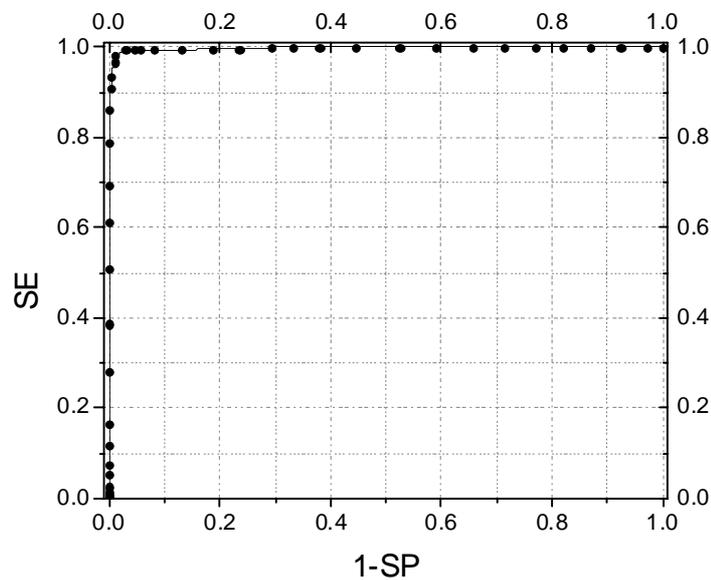
Fig. 3-9: ROC curve of the SVM-based classifier indicating its performance for cell detection.

Scenario II: Viability Classification of All-live and All-dead Mixture

In this scenario, yeast cultures with a variety of viabilities are obtained by mixing the all-live culture and all-dead culture at different ratios. For each mixture, viability is measured by taking the average of five manual counts based on FUN 1 stain. This value is then regarded as the gold standard and compared with the viability determined by the IDMP (for each mixture, three samples are investigated and averaged). The same training set as used in scenario I is used again. The correlation of these two determined values are shown in Fig. 3-10. The correlation factor of $R^2 = 0.99$ shows a high accuracy of the viability determination by the IDMP. The viabilities reported by the IDMP have a standard deviation within

±0.05, indicating the stability of the IDMP's performance.
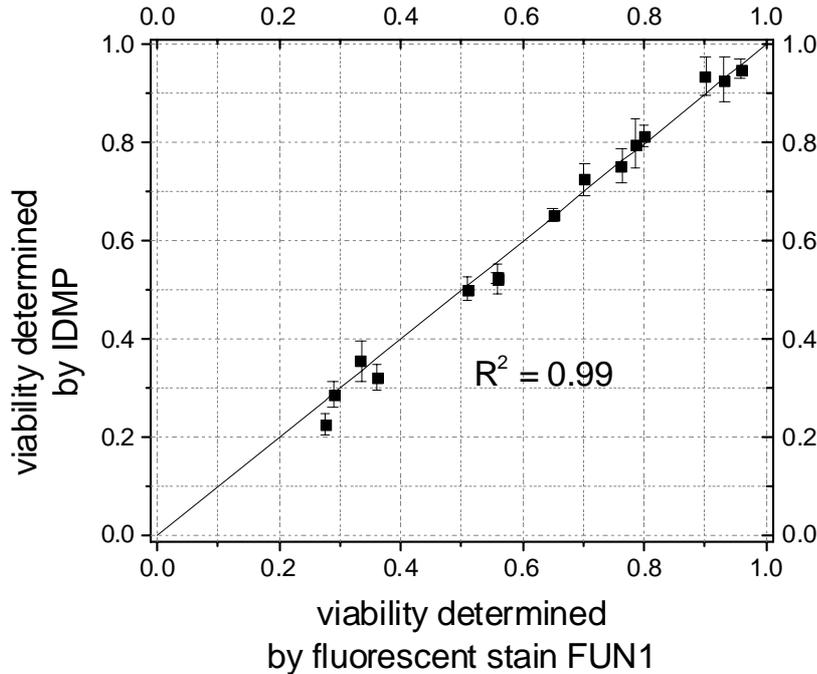
Fig. 3-10: Correlation of the viability determined by hand counts and by the IDMP of the yeast cultures mixed with all-live and all-dead cultures.

Fig. 3-11 shows the labels assigned by the IDMP to the cells. A circle with dashed line represents a live cell, while a square with solid line stands for a dead one. In Fig. 3-11 (a), the cell population was mixed at a ratio of 1:1 (all-live to all-dead culture), and the viability reported by the FUN 1 method was 50.6%. It can be seen from the image that the classification result by the IDMP (48.2%) is nearly identical; in Fig. 3-11 (b), the cell population was mixed at a ratio of 4:1 (all-live culture to all-dead culture), the viability reported by the IDMP was 80.0%, which resembles that reported by FUN 1 assay (79.2%) again.
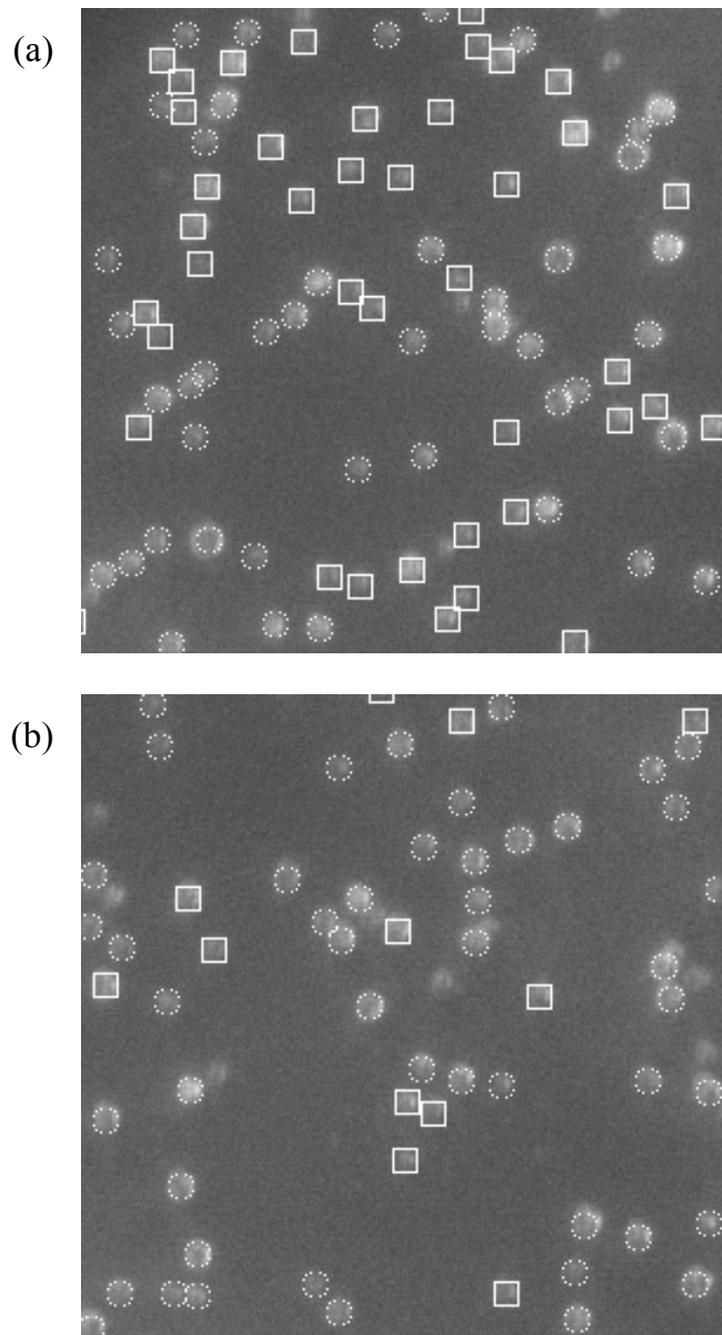
Fig. 3-11: Visualization of the cell classification results based on the IDMP for a yeast population with an expected viability value of (a) 50.6% and (b) 79.2%. Solid squares: live cells; dash circles: dead cells.

Scenario III: Viability Classification of All-live and Starvation Mixture

The negative training sets (dead cells) used in scenario I and II consist of only cells killed by heating in a water bath, and inspiring results have been achieved. However, all tested dead cells are obtained also by heating, therefore, it is yet unclear whether the system is also reliable when dead cells are collected in a different way. For instance, if the system can also give the correct viability for mixtures with all-live and starvation cultures, the result would be more convincing.

To this end, experiments have been carried out on mixtures of all-live and starvation cultures at different ratios. For comparison, the reference viability of any mixed culture is again determined via a manual counting by means of FUN 1 stain. For each mixture, an average of 5 manual counts is taken, and it is regarded as the gold standard. In parallel, the viability of the same mixture is then assessed using the IDMP with 3 samplings. The same training set is used as in scenario I and II. The correlation of these two determinations is shown in Fig. 3-12. The correlation factor $R^2 = 0.938$ and standard deviation within ±0.05 show the excellent performance of the IDMP.
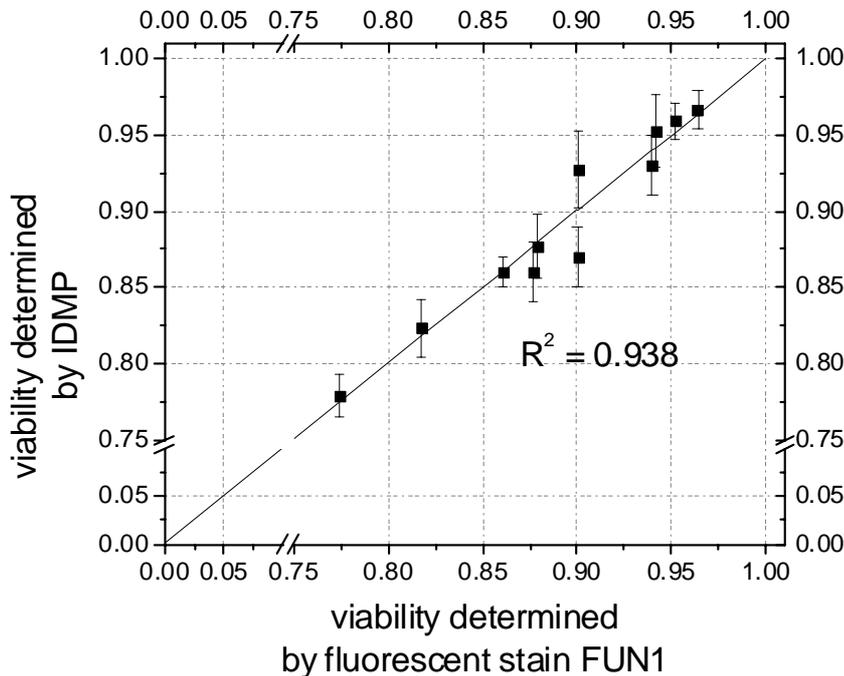
Fig. 3-12: Correlation of the viability determined by hand counts and by the IDMP of the yeast cultures mixed with all-live and starvation cultures.

Discussions

The *in situ* experiments show that the technique of dark field microscopy in conjunction with image processing and support vector machine is suitable for measuring cell density and viability of brewer's yeast, *Saccharomyces cerevisiae*, in a bioreactor. The results are so accurate and stable that a prospect of this probe's application in industry can be promisingly expected.

Advantages of this IDMP are not only that it realizes the *in situ* morphological observation of the cells and the measurement of both cell density and viability, but also that it achieves the goal without irreversibly perturbing the culture system. Using this probe, no auxiliary reagents are required for marking viable cells and nonviable cells in advance. Moreover, the training process itself is very simple as well: prior to implementing this probe, the classification system can be trained using only images captured from all-live and all-dead cultures in the bioreactor.

Abbreviations

FN: False Negative
FP: False Positive
IDMP: *In situ* Dark field Microscopy Probe
PCA: Principal Component Analysis
PPV: Positive Predictive Value
ROC: Receiver Operator Characteristics
SE: Sensitivity
SP: Specificity
TN: True Negative
TP: True Positive

List of Symbols

$\mathbf{x}$:   a sample of the dataset
d: original dimensionality in PCA
M: reduced dimensionality in PCA
$\mathbf{u}_i$:   orthogonal vectors
z_i: coefficients of the orthogonal vectors in representing $\mathbf{x}$
K: size of the dataset in PCA
$\widetilde{\mathbf{x}}$:   approximation of $\mathbf{x}$
b: bias
λ: eigenvalues of the covariance matrix
E_M: sum of the squares of the errors in the approximating $\mathbf{x}$ with $\widetilde{\mathbf{x}}$
N: width or height of the image patches
$\mathbf{\Gamma}_{train}$:   training set in cell detection

$\mathbf{\Gamma}^{+}_{train}$:   training set of cells in cell detection

$\mathbf{\Gamma}^{-}_{train}$:   training set of background in cell detection

n_{train}:   number of samples in the training set in cell detection

$\mathbf{P}$:   an image patch in cell detection

k: class label of an image patch in cell detection

v: index of the image patches in cell detection

v^{+}:   index of the cell patches in cell detection

v^{-}:   index of the background patches in cell detection

y: confidence value

$\mathbf{p}$:   a pixel on the confidence map

$C(\cdot)$:   confidence function

$\mathbf{\Psi}_{train}$:   training set in cell viability classification

$\mathbf{\Psi}^{+}_{train}$:   training set of live cells in cell viability classification

$\mathbf{\Psi}^{-}_{train}$:   training set of dead cells in cell viability classification

n'_{train}:   number of samples in the training set in cell viability classification

$\mathbf{V}$:   an image patch in cell viability classification

u: class label of an image patch in cell viability classification

Chapter 4

Wavelet Feature Selection for Viability Classification

In chapter 3, the foundation of cell viability classification is that live cells exhibit different visual appearances from dead ones. Thus, raw image patches of the cells have been used directly to determine cell viability by means of an SVM (Support Vector Machine) classifier. In spite of the high accuracy obtained in this way, one of the drawbacks of using raw image patches is that the dimensionality of the data is very high, so that it prevents the image from being rapidly processed. In this chapter, a Machine Vision System (MVS) is proposed for noninvasive classification of cell viability. This MVS, instead of making use of the raw image patches, is based on feature computation and selection.

Material and Methods

Principle of the MVS

The main idea of the system is to train the MVS with cell samples, the viability of which are known, in order that the MVS learns from the example images some criterion for distinguishing cell viability. In this learning process, image features are extracted and selected in order to support the classification.

The MVS is composed of two main modules: training (Fig. 4-10-a) and test module (Fig. 4-10-b). In the training module, two special kinds of cultures are used to generate the training dataset. The first kind, the all-live cultures (Fig. 4-10-a(1a)) are cell populations in which each cell is alive; while for the second kind, the all-dead cultures (Fig. 4-10-a(1b)), each cell is dead. Micrographs of these cultures are captured with a laboratory microscope (Phenix XSP 12, Hengtech Optische Instrumente Inc., Germany) plus a dark field illumination device (KL 1500 LCD, Schott, Germany). The images are recorded with the same CCD camera (Sony XCD-X700) as used in Chapter 3. The same cell detection program as used in Chapter 3 is used to find the positions of the cells on the micrographs (Referring to

Fig. 4-10-a(3)).

After the stage of cell detection, the image patch of each cell is collected within a window of N x N pixels around the detected cell centre. These cell patches compose the training set (Fig. 4-10-a(4)). Thereafter, features are computed for the N x N sized image patches prior to performing a feature selection algorithm (Fig. 4-10-a(5)), which determines the best subset of these features according to certain criteria so that the best performance can be achieved. The feature vector **x** of each live cell is assigned with a class label $y = 1$; while that of any dead cell is assigned with a class label $y = -1$. All of these labelled feature vectors are used to train a classifier (Fig. 4-10-a(6)) based on Support Vector Machine (SVM) technique.
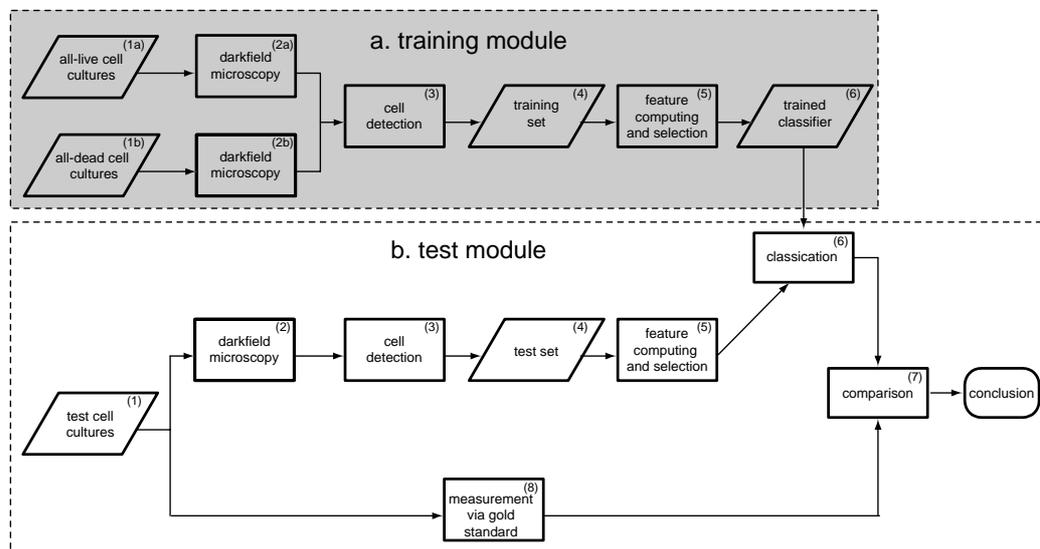
Fig. 4-1: The Machine Vision System (MVS) composed of the training module (shadowed region) and the test module (non-shadowed region).

After training the classifier, it can be applied to investigate new cell cultures, in which the cell viability is unknown (Fig. 4-1-b(1)). The test images are processed in a similar way as in the training module: capture of micrographs under a dark field microscope (Fig. 4-1-b(2)), cell detection (Fig. 4-1-b(3)), and selected feature computation (Fig. 4-1-b(5)). The subset of features is the same as in the training module. After that, the viability of each tested cell is determined with the SVM classifier (Fig. 4-1-b(6)). The results are then compared (Fig. 4-1-b(7)) with a gold standard (Fig. 4-1-b(8)) in order to evaluate the system's performance.

Strain, Medium and Culture Preparation

The same strain, brewer's yeast, *Saccharomyces cerevisiae* (strain Tokay), as applied in Chapter 3, is chosen to be the target microorganism. In cultivating yeast, a YM medium (glucose: 10 g L^{-1}, peptone: 5 g L^{-1}, yeast extract 3 g L^{-1}, malt extract 3 g L^{-1}, pH 6.2±0.2) is used.

As shown in Fig. 4-1-a, training the SVM classifier in the MVS needs all-live and all-dead cultures. The same protocols of preparing all-live and all-dead cultures as used in Chapter 3 are followed.

Gold Standard of Cell Viability Assessment

Gold standard for determining cell viability is a commercial fluorescence probe for live/dead yeast viability evaluation (FUN® 1 cell stain, Invitrogen Ltd, Karlsruhe, Germany), which is the same as that used in Chapter 3.

Wavelet Packet Feature Analysis

In the MVS, wavelet transform is performed for feature computation. Wavelet transform as an approach to multi-scale analysis of signals and images has been widely used in image compression, noise removal, texture segmentation, face recognition, medical image processing, and so on (Pichler et al., 1996; Charnbolle et al., 1998; Garcia et al., 2000; Soltanian-Zadeh et al., 2004).

In Wavelet analysis, a 1D continuous signal $f(x)$ can be expanded into the following form:

$$f(x) = \sum_k W_{\varphi,j_0}(k)\varphi_{j_0,k}(x) + \sum_{j=j_0}^{\infty}\sum_k W_{\psi,j}(k)\psi_{j,k}(x) \tag{1}$$

where $\{\varphi_{j,k}(x)\}$ and $\{\psi_{j,k}(x)\}$ are sets of scaling functions and wavelet functions, respectively. This series of functions have two parameters: the width, j, and the position, k:

$$\varphi_{j,k}(x) = 2^{j/2}\varphi(2^j x - k) \tag{2}$$

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k) \tag{3}$$

The coefficients $W_{\varphi,j_0}(k)$ and $W_{\psi,j}(k)$ are determined with following relationships:

$$W_{\varphi,j_0}(k) = \int f(x)\varphi_{j_0,k}(x)\,dx \tag{4}$$

$$W_{\psi,j}(k) = \int f(x)\psi_{j,k}(x)\,dx \tag{5}$$

Any of the scaling or wavelet functions can be represented as a weighted sum of scaling functions that have a double frequency:

$$\varphi(x) = \sum_n h_\varphi(n)\sqrt{2}\varphi(2x-n) \tag{6}$$

$$\psi(x) = \sum_n h_\psi(n)\sqrt{2}\varphi(2x-n) \tag{7}$$

in which h_φ and h_ψ are called scaling and wavelet vectors.

If $f(x)$ is a discretized function (x = 0, 1, 2, …, M-1), then Eq. (4) and (5) should be modified to:

$$W_{\varphi,j_0}(k) = \frac{1}{M}\sum_x f(x)\varphi_{j_0,k}(x) \tag{8}$$

$$W_{\psi,j}(k) = \frac{1}{M}\sum_x f(x)\psi_{j,k}(x) \tag{9}$$

Eq. (8) and (9) are called Discrete Wavelet Transform (DWT), which is performed through operating $f(x)$ with scaling and wavelet functions. In the Fast Wavelet Transform (FWT) algorithm, a relationship between DWT coefficients in adjacent levels is discovered, and the operation is performed with scaling and wavelet vectors (h_φ and h_ψ):

$$W_{\varphi,j-1}(n) = [h_\varphi(-n) * W_{\varphi,j}(n)]_{\downarrow 2}$$

$$W_{\psi,j-1}(n) = [h_\psi(-n) * W_{\varphi,j}(n)]_{\downarrow 2} \tag{10}$$

where * denotes the convolution operator, $\downarrow 2$ denotes sub-sampling. It is evident that h_φ plays a role as a low-pass filter and h_ψ as a band-pass filter, and the original signals can be split into approximations (W_φ) and details (W_ψ).
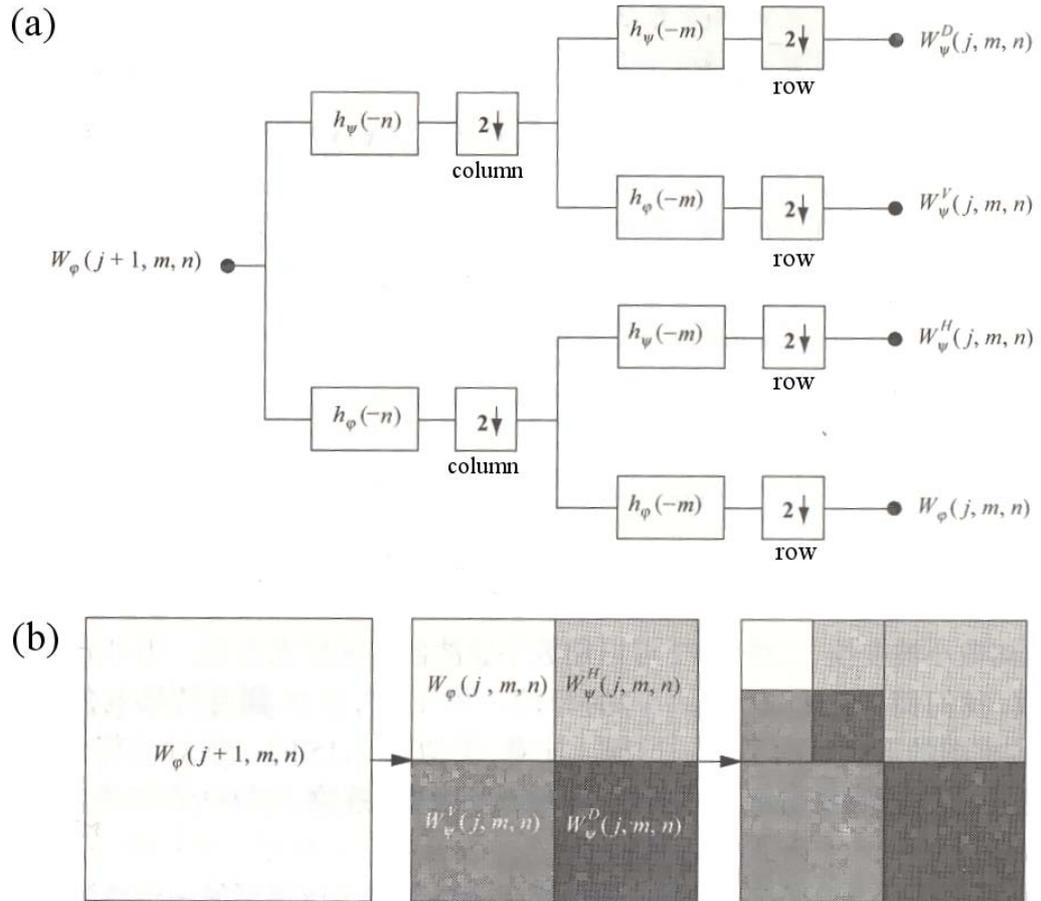
Fig. 4-2: 2D wavelet decomposition. (a) Flow diagram of the algorithm. (b) Subimages of the classical 2D wavelet decomposition.

In 2D cases, such as in classical wavelet decomposition of images, each image is split into approximations and details. The approximations are further split into approximations and details with a 2D FWT (as can be seen in Fig. 4-2):

$$W_{\varphi,j-1}(m,n) = [h_{\varphi}(-m) * [h_{\varphi}(-n) * W_{\varphi,j}(m,n)]_{\downarrow 2(c)}]_{\downarrow 2(r)}$$

$$W_{\psi,j-1}^{H}(m,n) = [h_{\psi}(-m) * [h_{\varphi}(-n) * W_{\varphi,j}(m,n)]_{\downarrow 2(c)}]_{\downarrow 2(r)}$$

$$W_{\psi,j-1}^{V}(m,n) = [h_{\varphi}(-m) * [h_{\psi}(-n) * W_{\varphi,j}(m,n)]_{\downarrow 2(c)}]_{\downarrow 2(r)} \qquad (11)$$

$$W_{\psi,j-1}^{D}(m,n) = [h_{\psi}(-m) * [h_{\psi}(-n) * W_{\varphi,j}(m,n)]_{\downarrow 2(c)}]_{\downarrow 2(r)}$$

where $\downarrow 2(c)$ ($\downarrow 2(r)$) denotes sub-sampling along the columns (rows). If the original level of the signals is J, then $W_{\varphi,J}(m,n) = f(m,n)$ is the original image.

$W_{\varphi,J-i}(m,n)$ denotes the approximations subimage at scale i, and $W_{\psi,J-i}^{t}(m,n)$ denotes the details subimages at scale i ($t = H, V, D$ for horizontal, vertical and diagonal details information).

Fig. 4-3 shows an example of the classical wavelet 2D decomposition of an image. In Fig. 4-3 (a) the original image is displayed. The approximation information (AI), horizontal, vertical, and diagonal detail information (HI, VI, DI) of the original image are given at the upper left, upper right, lower left, and lower right corner of Fig. 4-3 (b). Subsequently, at the upper left corner of Fig. 4-3 (c), the AI, HI, VI, DI of the AI shown in Fig. 4-3 (b) have been computed. Analogously, in Fig. 4-3 (d) the AI, HI, VI and DI are further computed at the upper left corner.

In a *wavelet packet decomposition*, both the approximations and details are split, which provides richer information for signal analysis. An example of a complete wavelet packet decomposition of a fingerprint image is shown in Fig. 4-4.
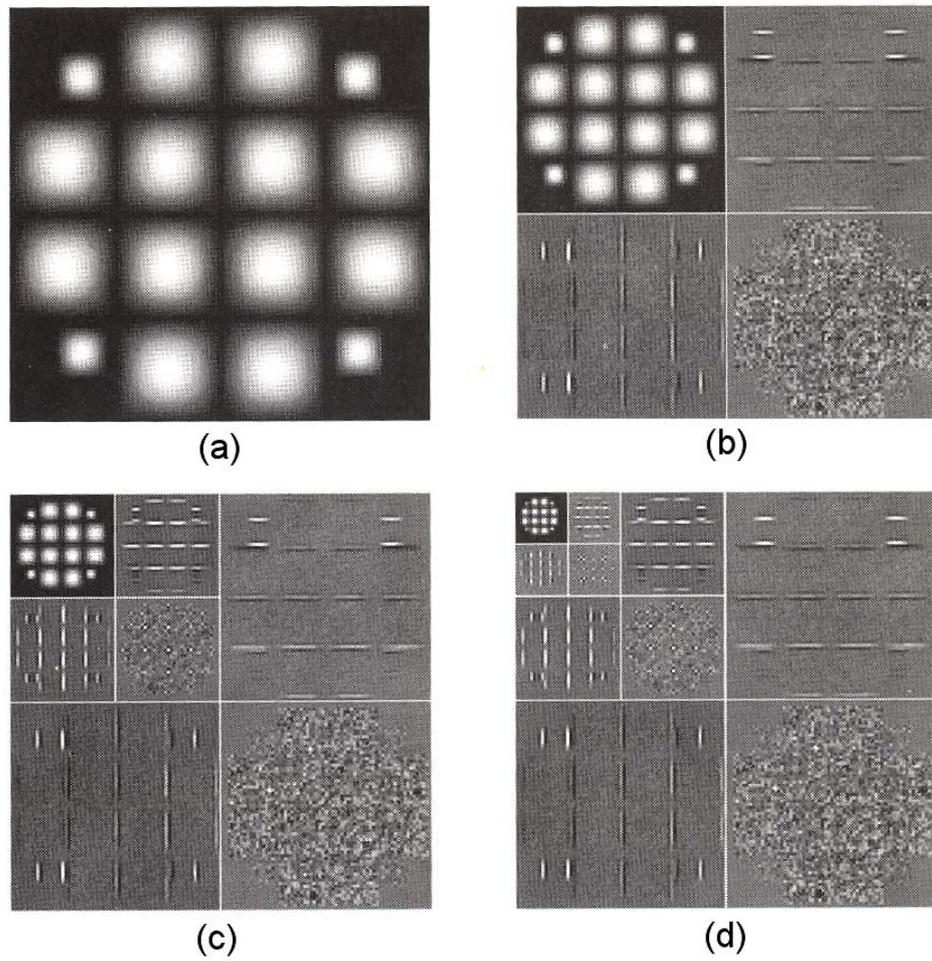
Fig. 4-3: An example of the classical 2D wavelet decomposition of an image.
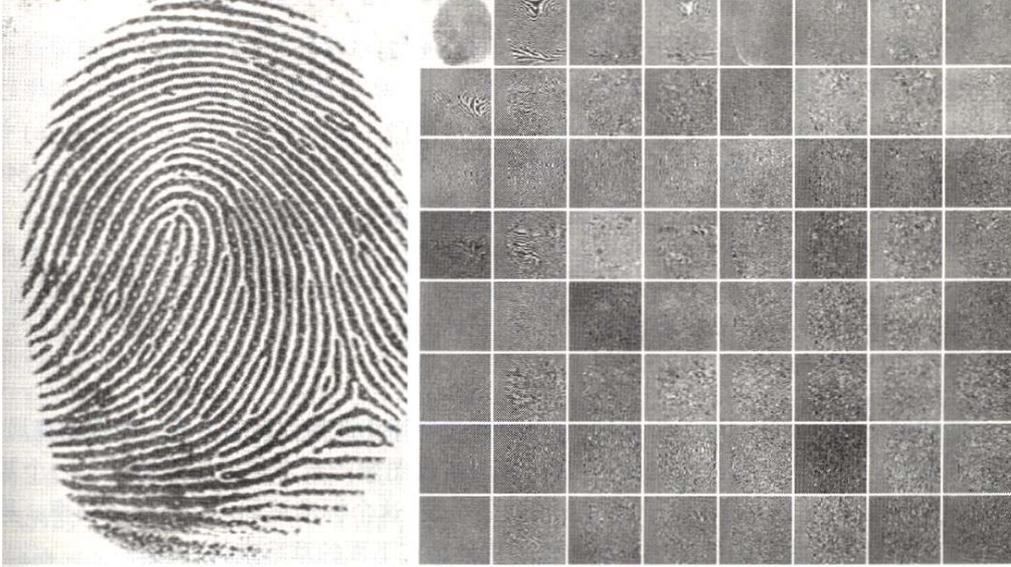
Fig. 4-4: An example of a complete 2D wavelet packet decomposition of a fingerprint image at level 3. In total there are 64 subimages generated. Left: original image of the fingerprint; Right: the subimages of the complete decomposition.

In the proposed scheme, a two-level wavelet packet decomposition is performed to the cell images, as shown in Fig. 4-5. For simplicity, in this figure W_φ and W_ψ are denoted as V and W, respectively. First, FWT is used to decompose the original image, V_J, into four subimages: V_{J-1} (approximation), W^H_{J-1} (horizontal detail), W^V_{J-1} (vertical detail), and W^D_{J-1} (diagonal detail). Then a second FWT is used to decompose V_{J-1}, W^H_{J-1}, W^V_{J-1} and W^D_{J-1} into corresponding approximation and detail subimages.

A Daubechies wavelet with four taps is used for filtering the images. The scaling and wavelet function are shown in Fig. 4-6.
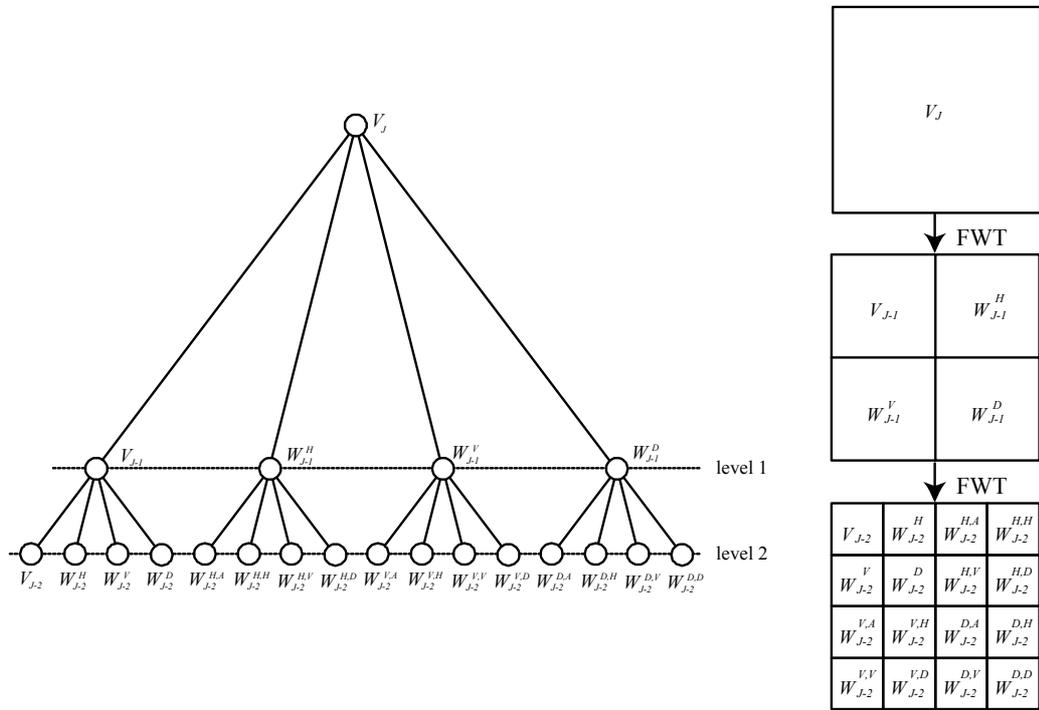
Fig. 4-5: A two level complete wavelet packet decomposition by means of fast wavelet transform (FWT).
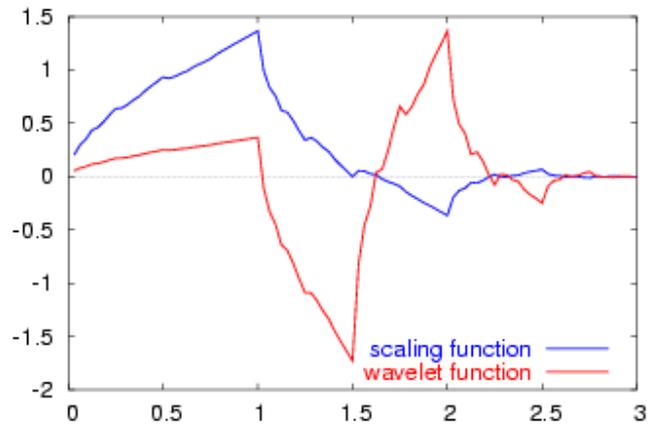
Fig. 4-6: The Daubechies wavelet with four taps.

At each level of the decomposition, the frequency space is split into four sub-spaces, which leads to a total of 4^2 sub images at the second level, including one approximation and 15 details. Provided that each sub image has a size of $N \times N$ pixels, its energy (E) and entropy (E_p) are computed as follows:

$$E = \frac{\sum_i \sum_j u_{ij}^2}{N^2} \tag{12}$$

$$E_p = -\sum_i \sum_j p(u_{ij}^2) \log(p(u_{ij}^2)) \tag{13}$$

in which u_{ij} is the ij-th pixel value of the subimage, and $p(\cdot)$ denotes the probability of the occurrence of value u_{ij}^2. Here the values u_{ij}^2 are quantized into 50 bins in the histogram. In Fig. 4-7 one typical histogram of u_{ij}^2 in a subimage is shown. In this figure, the maximum of u_{ij}^2 is 70.82. Thus, width of each pin is 1.42 (namely, 70.82/50). For example, the first pin covers u_{ij}^2 from 0 to 1.42, and the number of occurrences of u_{ij}^2 within this domain is 26. In this work, the original image patches have a size of 25×25 pixels, hence, the size of each subimage is 7×7 pixels. Thus, the probability of any $0 < u_{ij}^2 < 1.42$ is 0.53 (namely, $26/7^2$).
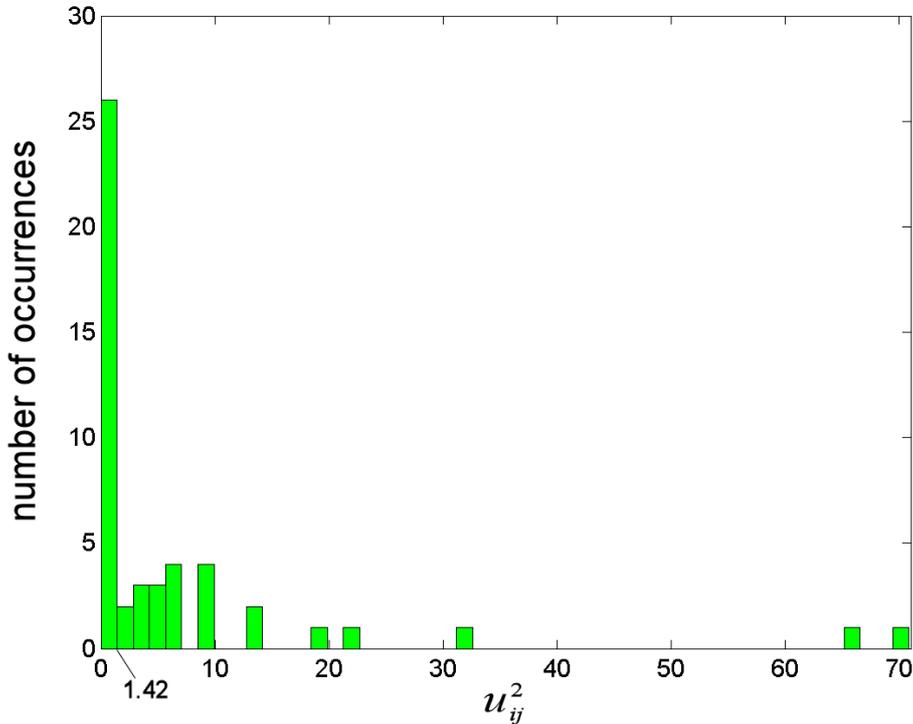
Fig. 4-7: One example of the histogram of the values u_{ij}^2 in a wavelet subimage.

	(0)	(1)	(2)	(3)
(0)	V_{J-2}	W_{J-2}^{H}	$W_{J-2}^{H,A}$	$W_{J-2}^{H,H}$
(1)	W_{J-2}^{V}	W_{J-2}^{D}	$W_{J-2}^{H,V}$	$W_{J-2}^{H,D}$
(2)	$W_{J-2}^{V,A}$	$W_{J-2}^{V,H}$	$W_{J-2}^{D,A}$	$W_{J-2}^{D,H}$
(3)	$W_{J-2}^{V,V}$	$W_{J-2}^{V,D}$	$W_{J-2}^{D,V}$	$W_{J-2}^{D,D}$

subimages

Energy   Entropy

0	1	4	5
3	2	7	6
12	13	8	9
15	14	11	10

16	17	20	21
19	18	23	22
28	29	24	25
31	30	27	26
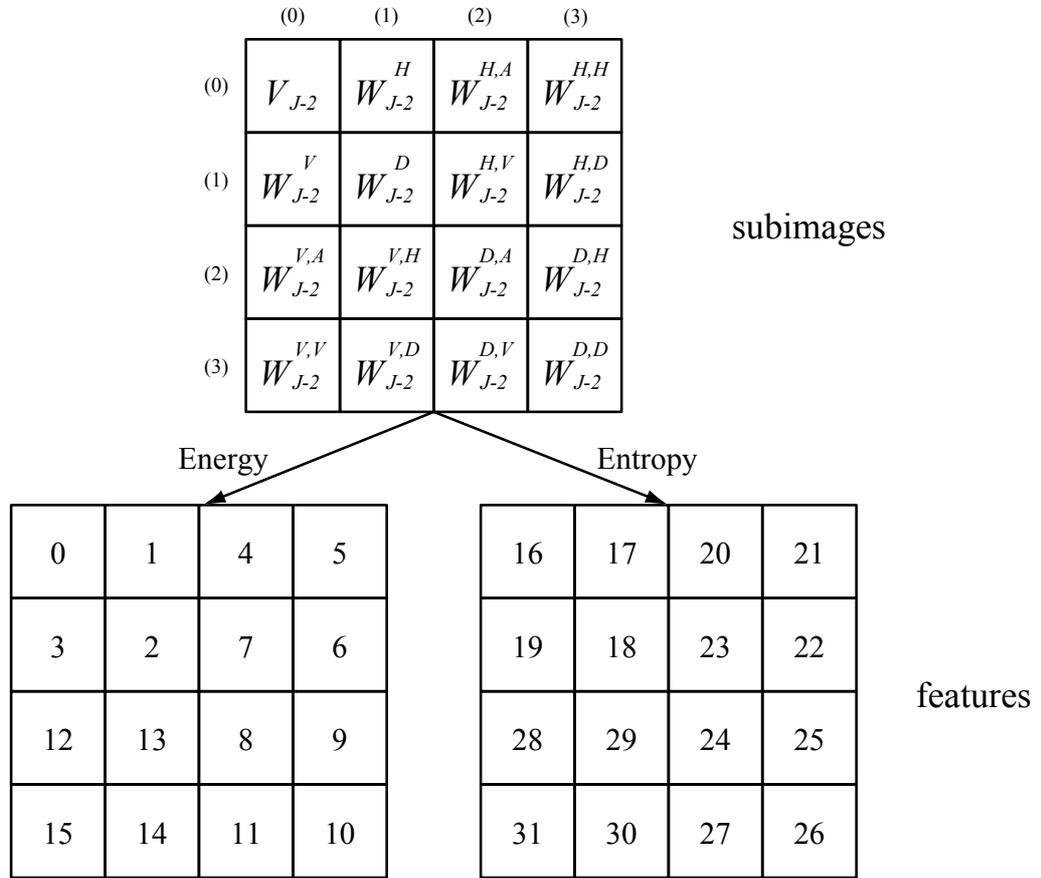
features

Fig. 4-8: Layout of the wavelet packet subimages and their features. For each subimage two features (energy and entropy) are computed.

Energy and entropy are computed for all subimages, therefore, in total 32 features are generated for each image. These features are assigned a number in accordance with a layout shown in Fig. 4-8. The meaning of the feature can also be determined in this figure. For instance, feature 2 is the energy of the subimage (1,1), namely, W^{D}_{J-2}; feature 28 is the entropy of the subimage (0,2), namely, $W^{V,A}_{J-2}$.

Feature Selection

Feature selection has been one of the focuses in pattern recognition because it discovers the subset of features that carries the most discriminative information and abandons those containing more noise than useful information. The advantages of feature selection can be versatile: for instance, reducing dimensionality, enhancing system robustness, increasing recognition rate, and so on.

A large number of algorithms have been proposed for feature selection. Among them, a sequential floating selection algorithm (Pudil et al., 1994; Jain and Zongker, 1997; Kudo et al., 2000) has been shown to be superior to others in comparative studies. This algorithm can be carried out in two different directions - forward (Sequential Forward Floating Selection, or SFFS) and backward (Sequential Backward Floating Selection, or SBFS). In the former case, the program starts with an empty subset, and searches for the optimal solution by iteratively adding features into the subset; while in the latter case, it starts with a complete set of features, and discards features iteratively. A floating selection process has been applied so that previously added or discarded features still own the chance to be discarded or added, which leads to a higher probability of finding the optimum. If a criterion function $F(S)$ can be determined for any feature subset S, then a best subset S^* can be found by using the SFFS or SBFS algorithm. One essential requirement imposed upon the definition of the criterion function is that the better the subset S is, the higher is $F(S)$.

Feature selection in the MVS is performed based on the performance evaluation of the SVM classifier on determining viability of all-live and all-dead cultures. Assume that from the all-live culture K samples are taken, and the corresponding viability values determined by the classifier are denoted as l^S_j ($j = 1, 2, \ldots K$), S being the present feature subset.

Assume that from the all-dead culture also K samples are taken, and the viability values determined by the MVS are denoted as d^S_j ($j = 1, 2, \ldots K$). Assume the real viability of the all-live culture is l_0, and the real viability of the all-dead culture is d_0, the criterion function C with regards to S can be constructed in the following form:

$$C(S) = -\frac{1}{K}\sum_{j=1}^{K}(l_0 - l_j^S)^2 - \frac{1}{K}\sum_{j=1}^{K}(d_0 - d_j^S)^2 \tag{14}$$

It is clear that the higher the criterion function value, the better is the classifier's performance, and consequently, the better is the feature subset S, which satisfies the aforementioned requirement of the SBFS algorithm.

Results

Micrographs are taken for 10 samples from an all-live culture. After the cell detection process, 466 live cells are recognized. Similarly, 491 dead cells are detected from 10 samples in an all-dead culture. From these, 232 live cells and 247 dead cells are used to generate the training set; while the other 234 live cells and 244 dead cells are used to generate the test set. With these datasets, feature selection is performed and the SVM classifier with a linear kernel is trained. The best subsets of wavelet features are selected with the SBFS algorithm, with a criterion function defined in the form of Eq. (14).

The selection results are shown in Fig. 4-9. The global optimum is found when the number of features is 16, with a criterion value of -0.01454. It is evident that a decline of the criterion within 10% is tolerable (the grey region shown in the inset of Fig. 4-9), as a result, the best choice of the feature number should be 12, which leads to a criterion of -0.01516, within the tolerable region. In this 12-featured subset, feature 0, 1, 2, 3, 5, 6, 10, 16, 19, 20, 24, 28 are included. When too many features are discarded the criterion declines significantly, especially when the feature number is less than 6.

In order to evaluate the performance of the MVS with cultures of given viability, mixed cultures are prepared as test sets. Mixed cultures are obtained by mixing all-live and all-dead cultures at a series of ratios (1:4, 2:3, 3:2, 4:1, 17:3, and 9:1). For each mixture, the viability is measured by taking the average of 5 manual counts based on FUN 1 stain. As the cell density of the all-live cultures is a little bit different from that of the all-dead cultures, the outcome viability is slightly deviated from the nominal value of the mixing ratio. For instance, the nominal viability of the 2:3 (all-live to all-dead) mixture is 0.4, while the actual value is nevertheless about 0.37. The viability determined by FUN 1 stain is regarded as the gold standard and compared with that by the MVS. For each mixture, 5 samples are investigated and the viability values are averaged. The total number of the cells in the test sets is 1702.
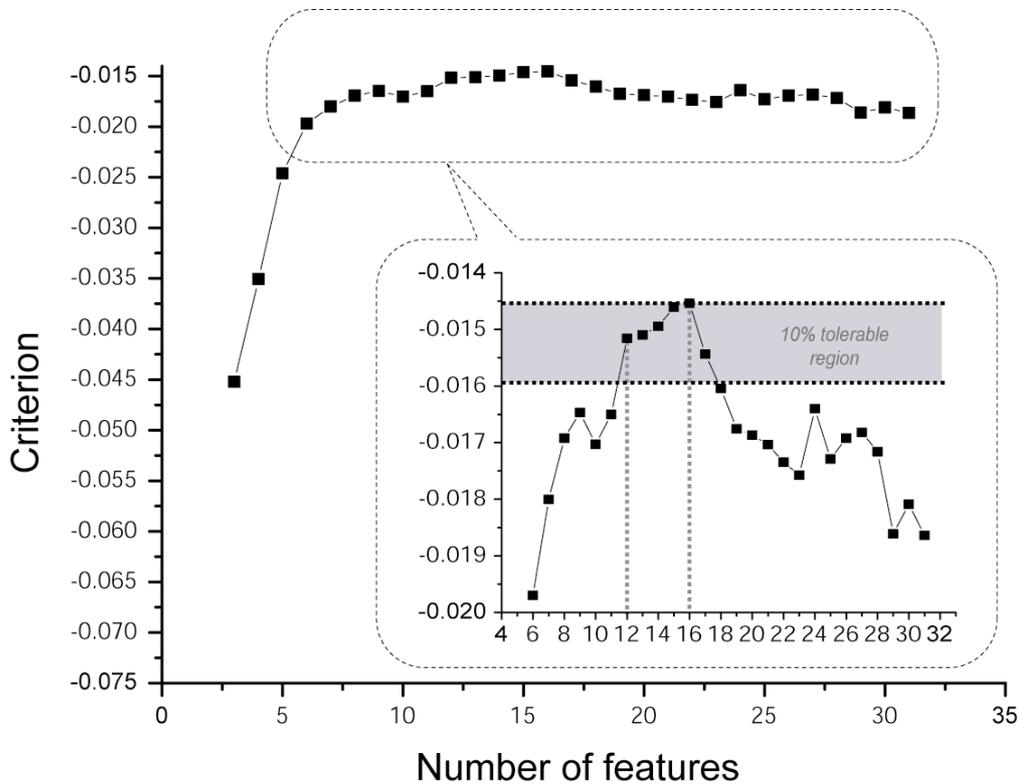
Fig. 4-9: Results of feature selection using Sequential Backward Floating Selection (SBFS). The global optimum is found to be the best 16-featured subset. Given a tolerance of 10%, the best 12-featured subset is also acceptable.

The correlation of the results given by the MVS and by the gold standard is displayed in Fig. 4-10. The system performance is evaluated in three scenarios. In Fig. 4-10 (a), the training set and test set of the classifier are composed of only raw image patches (namely, without feature extraction). In Fig. 4-10 (b) the complete set of 32 wavelet features is used. In Fig. 4-10 (c) a selected subset of 12 features is used. The effectiveness of feature extraction can be recognized in these figures. Comparing Fig. 4-10 (a) with (b), it can be seen that the use of wavelet features not only leads to stronger correlation with the gold standard, but also leads to lower variances. By comparing Fig. 4-10 (b) and (c), it is clear that discarding 20 features does not impair the system's performance. On the contrary, the selected feature set not only helps to increase the accuracy of the measurement, but also helps to reduce the variance in spite of a small increased variance at the viability of 0.85.
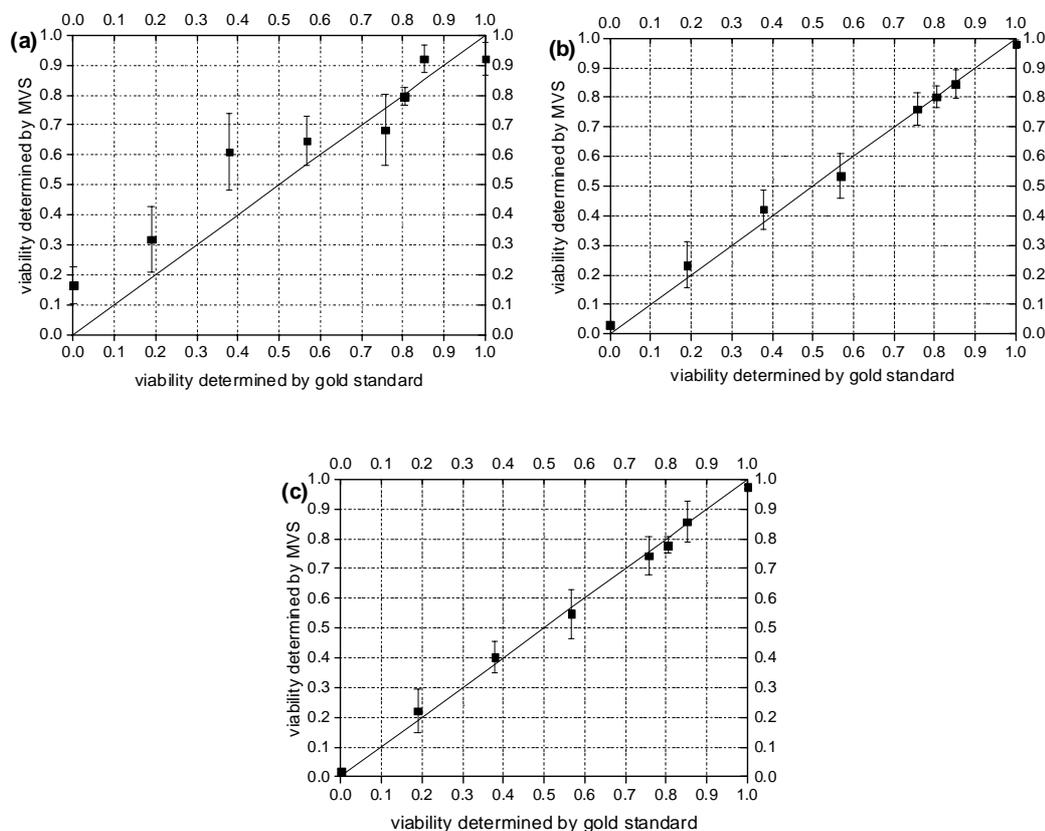
Fig. 4-10: Performance evaluation of the MVS: correlation analysis of the results given by the gold standard and by the MVS (a) using raw image patches; (b) using the complete set of 32 wavelet packet features; (c) using a selected subset of 12 features.

Discussion

In the course of feature selection, after each backward step a number of forward steps are performed as long as the resulting subsets are better than those previously evaluated at that level. With this so-called floating search mechanism, it is sure to achieve results at least close to optimal. In this sense, the features most frequently selected at all levels (at different levels, different numbers of best features are determined at the end of the algorithm) are considered as carrying the most important information. This can be well studied in Fig. 4-11. For instance, the best 5-featured subset contains feature 0, 1, 3, 10, and 19, which are labelled with a black filled circle. From this figure it can be recognized that feature 0, 3, 5, 10, 16, and 19 are the most frequently selected ones.
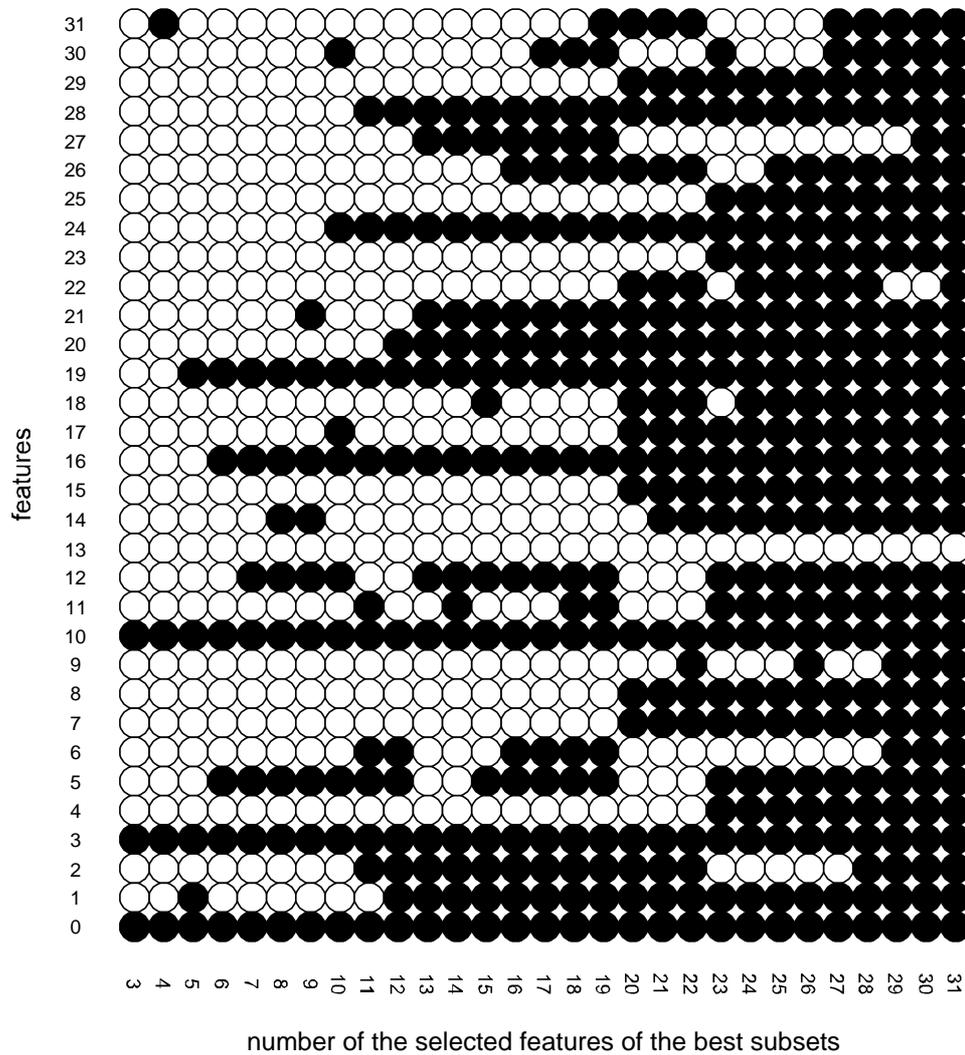
Fig. 4-11: The most frequently selected features. The black filled circles signify features that are selected for the best subsets, which are determined with SBFS and contain the given number of features.

Referring to Fig. 4-8, these features belong to subimage (0,0), (0,1), (3,0), and (3,3), the physical significances of which are given in Table 4-1 according to the principle of wavelet packet decomposition. These five subimages contain the most important discriminative information. It is evident that higher order details in horizontal and diagonal direction (subimage (3,0), (3,3)) and vertical details at a low level (subimage (0, 1)) are critical for classifying live and dead cells.

Table 4-1: The physical significance of the most frequently selected features of the two level complete wavelet packet decomposition (refer to Fig. 4-8).

Feature	Sub-image	Physical significance
0, 16	(0,0)	approximations of approximations (higher order approximations)
5	(3,0)	horizontal details of horizontal details (higher order horizontal details)
3, 19	(0,1)	vertical details of approximations (low order vertical details)
10	(3,3)	diagonal details of diagonal details (higher order diagonal details)

This opinion is supported by the reconstruction of cell images. First, any of the original cell images is decomposed using FWT. Second, all the pixels in the subimages (0,1), (3,0), and (3,3) are set to zero. Third, an inverse FWT (IFWT) is used to obtain a reconstructed image, which loses all the most important discriminative information. The comparison between original and reconstructed images is shown in Fig. 4-12. It can be seen that the reconstructed live cells (column 2) exhibit more "grid effect" than the reconstructed dead cells (column 4), which leads to a greater difference between the reconstructed and original live cells (column 1) than that between reconstructed and original dead cells (column 3). That may serve as evidence for supporting the assumption that live cells contain more detail information than dead ones. Based on Fig. 4-12, it is also clear that with the loss of the information in subimages (0,1), (3,0), and (3,3), which benefits viability classification, live cells are hardly to be distinguished from dead ones.

Following the similar idea, it has also been performed that all the pixels in the subimages except (0, 0), (0,1), (3,0), and (3,3) are set to zero. This attempt discards all information that is of less significance to distinguishing live cells from dead ones. The comparison between the original and reconstructed images is shown in Fig. 4-13. It is clear that the differences between the original and the reconstructed cell images is not so significant as those shown in Fig. 4-12. It is also implied that without the information in the subimages except (0,1), (3,0), and (3,3), live cells can still be distinguished from dead ones.

live		dead	
Original	reconstructed	original	reconstructed

Fig. 4-12: Comparison of original and reconstructed images of live and dead cells. Reconstruction is performed after assignment of zeros to the subimages (0,1), (3,0) and (3,3) (refer to Fig. 4-8) via inverse fast wavelet transform.

live		dead	
original	reconstructed	original	reconstructed

Fig. 4-13: Comparison of original and reconstructed images of live and dead cells. Reconstruction is performed after assignment of zeros to the subimages except (0, 0), (0,1), (3,0) and (3,3) (refer to Fig. 4-8) via inverse fast wavelet transform.

For an extended feature analysis, the distribution of the features for live and dead cells selected from the training set is displayed in a parallel coordinate plot in Fig. 4-14. Each thin, red, solid line represents a live cell, and each thin, black, dashed line represents a dead cell. The mean feature value over all live cells is displayed with a thick, white, solid line, while that over all dead cells is displayed with a thick, white, dashed line. Referring to the definition of features (Fig. 4-8), it is clear that with any energy feature (feature 0 ~ 15), live cells have a higher mean value. That means, in average live cells look brighter than dead cells (feature 0), and contain more details (feature 1 ~ 15). It is also clear that with any entropy feature (feature 16 ~ 31), live cells have a lower mean value (except for feature 16). It implies that live cells contain more inhomogeneous fine structures than dead cells.
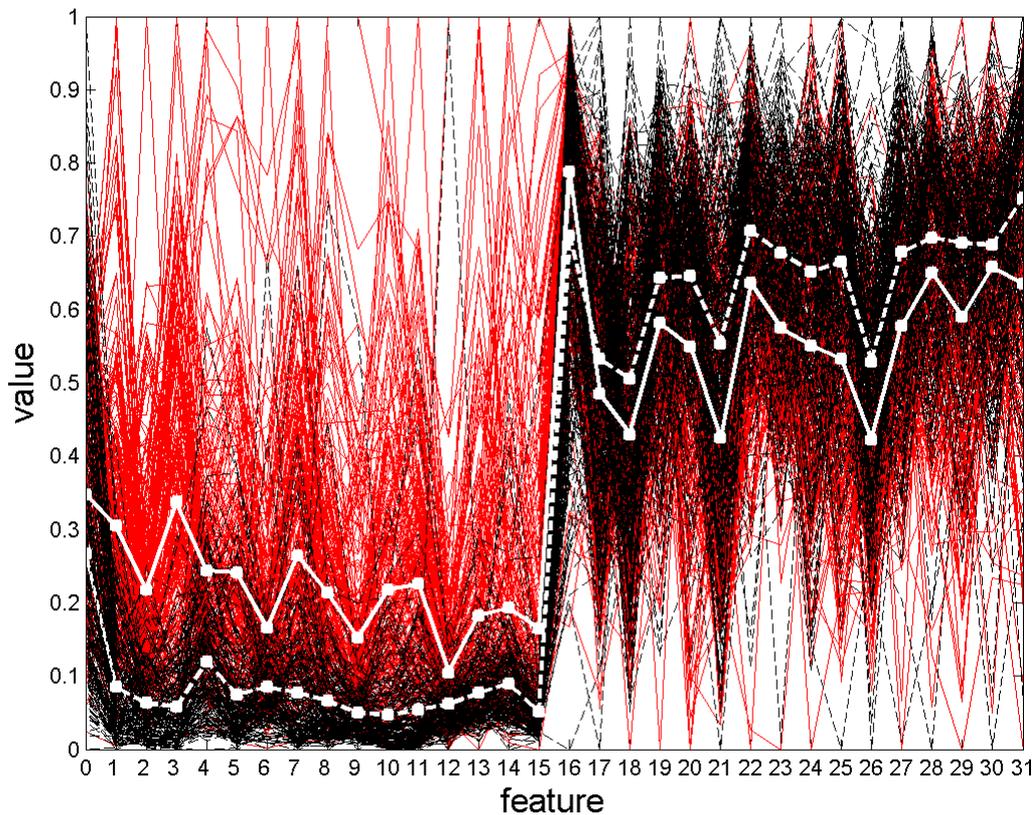
Fig. 4-14: Feature distribution of live and dead cells. Thin, read, solid lines: live cells; thin, black, dashed lines: dead cells; thick, white, solid line: mean feature value over all live cells; thick, white, dashed line: mean feature value over all dead cells.

Abbreviations

AI:	Approximation Information
DI:	Diagonal detail Information
DWT:	Discrete Wavelet Transform
FWT:	Fast Wavelet Transform
HI:	Horizontal detail Information
MVS:	Machine Vision System
VI:	Vertical detail Information
SBFS:	Sequential Backward Floating Selection
SFFS:	Sequential Forward Floating Selection
SVM:	Support Vector Machine

List of Symbols

$f(\cdot)$:	the signal
$\varphi_{j,k}(\cdot)$:	scaling functions
$\psi_{j,k}(\cdot)$:	wavelet functions
$W_{\varphi,j}(\cdot)$:	coefficients of the scaling functions in signal representation
$W_{\psi,j}(\cdot)$:	coefficients of the wavelet functions in signal representation
h_φ:	scaling vectors
h_ψ:	wavelet vectors
N:	width or height of a wavelet subimage
u_{ij}:	the ij-th pixel of a wavelet subimage
E:	energy of a wavelet subimage
E_p:	entropy of a wavelet subimage
K:	the number of samples taken from all-live or all-dead culture
S:	a feature subset
l_j^S:	viability of the all-live culture samples determined by the classifier
d_j^S:	viability of the all-dead culture samples determined by the classifier
l_0:	real viability of the all-live culture
d_0:	real viability of the all-dead culture
$C(\cdot)$:	criterion function in feature selection

Chapter 5

Viability Classification in Time Series Micrographs Using Principal Component Features

In chapter 3 and 4, efforts have been made to realize the measurement of cell viability by means of extracting information from static images of the cells. Since significant intracellular movements may only be observed in live cells, the temporal information of the interior structures of the cells may also be useful for distinguishing live and dead cells. In this chapter, we focus on extracting dynamic information of the cells using PCA (Principal Component Analysis) techniques, so that a way of classifying cell viability based on temporal patterns of live and dead cells can be found. Two kinds of classifiers are used for classification: one based on SVM (Support Vector Machine), the other based on GMM (Gaussian Mixture Model). The classification results of these two kinds of classifiers are compared.

Material and Methods

Microorganism Strain

Like in Chapter 3, brewer's yeast, *Saccharomyces cerevisiae* (strain Tokay), is again chosen as the subject of investigation.

Preparation of Different Types of Cultures

In the experiments all-live and all-dead cultures are used, which are prepared in the same way as described in Chapter 3.

Gold Standard of Viability Assessment

The same fluorescence probe as used in Chapter 3 (FUN 1 cell stain, Invitrogen Ltd, Karlsruhe, Germany) for live/dead yeast viability evaluation is used as the gold standard for assessing the performance of the system.

Image Acquisition

Images of yeast cells have been taken with a normal laboratory microscope plus a dark-field illumination device (KL 1500 LCD, Schott, Germany). 12 samples of all-live culture and 12 samples of all-dead culture have been photographed with dark field microscopy. A 40x magnification objective lens has been used to take images at 5 successive time points for each sample of the cells. As the digital CCD camera (KP-F100, Hitachi, Japan) does not provide the option of automated capture of images at different time points, we take the time series images only manually. The temporal interval of every two successive time points is approximately 2 seconds.

Image Processing

Provided that training images are captured at T different time points. At each time point t, the captured image is denoted as I_j ($j = 1, 2, ... T$). Extraction of principal component features is performed in following steps.

Firstly, assume that there are totally m pixels on I_1, the pixel values of which are all greater than r. The i-th pixel is denoted as $p_{i,1}$ (see Fig. 5-1), in which the subscript "1" represents I_1, and $i = 1, 2, ... m$. The pixel value of $p_{i,1}$ is denoted as $v_{i,1}$, the coordinates of $p_{i,1}$ as $(x_{i,}, y_{i,})$. Taking into account the pixels at $(x_{i,}, y_{i,})$ on all the I_j's, a so called *time series vector* is denoted as $\mathbf{v}_i = (v_{i,1}, v_{i,2}, ... v_{i,T}) \in \mathcal{R}^T$.

Secondly, PCA is performed on the m time series vectors, and T eigenvectors ($\mathbf{e}_j$, $j = 1, 2, ... T$) and corresponding eigenvalues (c_j, $j = 1, 2, ... T$) can be determined.

Thirdly, all the time series vectors are projected onto the eigenvectors. Assume there are totally P pixels on each of the I_t's, then *principal component vectors* (PC vectors) are consequently obtained by

$$\mathbf{v}_i \xrightarrow{\text{PCA}} \mathbf{v}_i^{\text{e}} \ (i = 1, 2, ... P)$$

in which $\mathbf{v}_i^{\text{e}} = (v_{i,1}^{\text{e}}, v_{i,2}^{\text{e}}, ... v_{i,T}^{\text{e}})$ denotes the i-th PC vector, and $v_{i,j}^{\text{e}}$ denotes the j-th PC of $\mathbf{v}_i^{\text{e}}$. Thereafter, the j-th *PC images* (I_j^e) is obtained by taking all $v_{i,j}^{\text{e}}$:

$$I_j^e = \begin{bmatrix} v_{1,j}^e & v_{2,j}^e & \cdots \\ \cdots\cdots\cdots \\ \cdots\cdots v_{P,j}^e \end{bmatrix} (j = 1, 2, ...T)$$

Fourthly, use the cell detector described in Chapter 3 to find the cell centers on I_1. The j-th *PC patch* of a cell is collected on I_j^e by extracting all the pixel values within a N x N pixels sized window that centers at the cell.
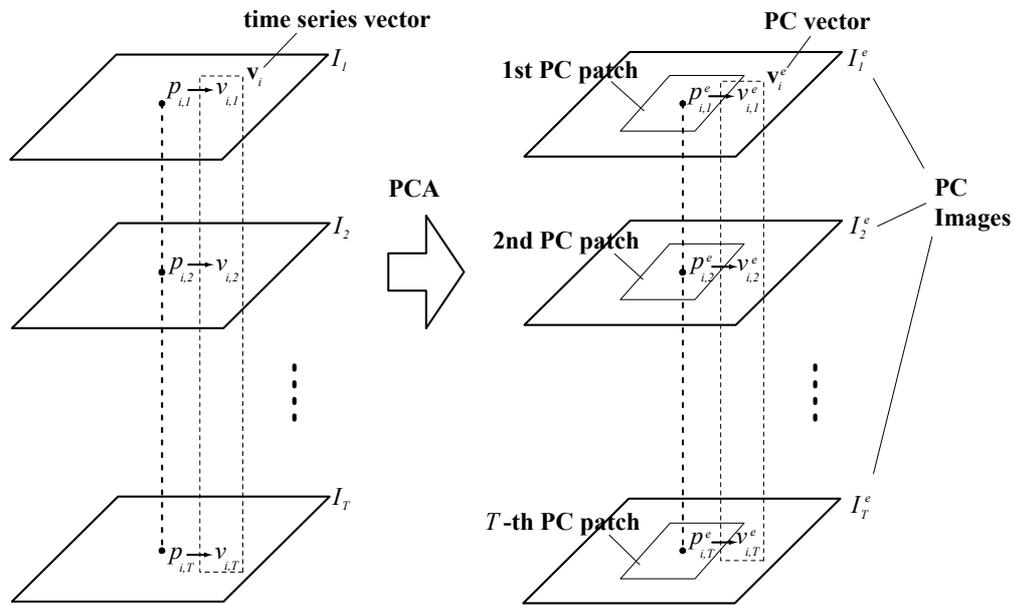
Fig. 5-1: PCA on the time series images to extract temporal features.

In Fig. 5-2, an example of a 2nd PC image of a time series of images with ten time points has been shown. In Fig. 5-3 the examples of some 2nd PC patches (51 x 51 pixels) of live and dead cells as shown. The three columns on the left hand side are 2nd PC patches of live cells; at the right hand side, four columns contain those of dead cells. It is clear that due to two different temporal intracellular patterns, the live cells can be discriminated from the dead cells without much effort of visual inspection. Despite of the apparent differences between live and dead cells, these patches still contain cross-talk information. It can be seen that in some patches there are not only one cell (row 1:column 3, row 2:colum 2, and so on), therefore, the information from the other cells would be mixed up and could cause errors. For this reason, a reduced patch size of only 31 x 31 pixels is used.
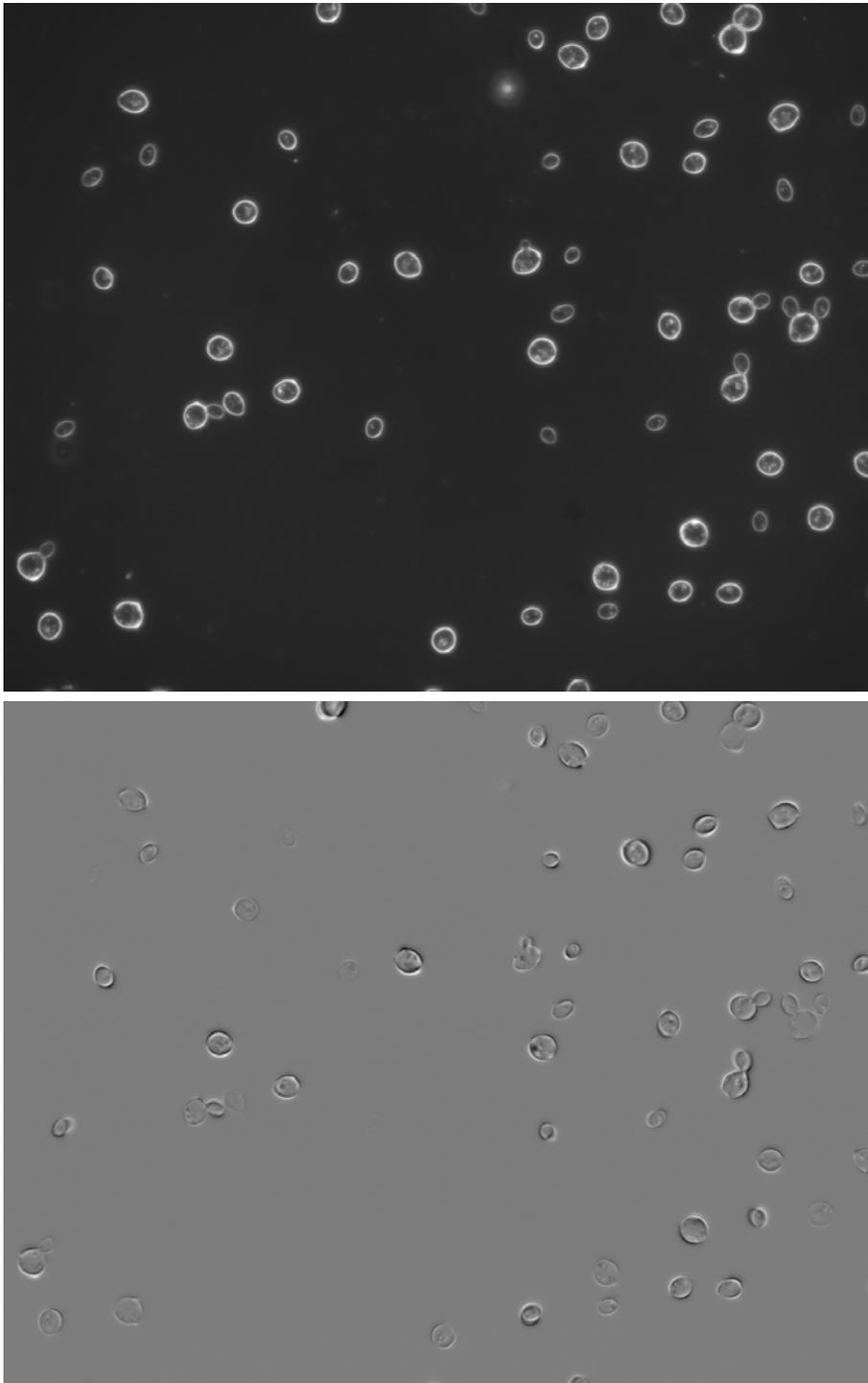
Fig. 5-2: An example of the PCA feature extraction. Upper: One of the original images of an image series with 10 time points. Bottom: the 2nd PC image.
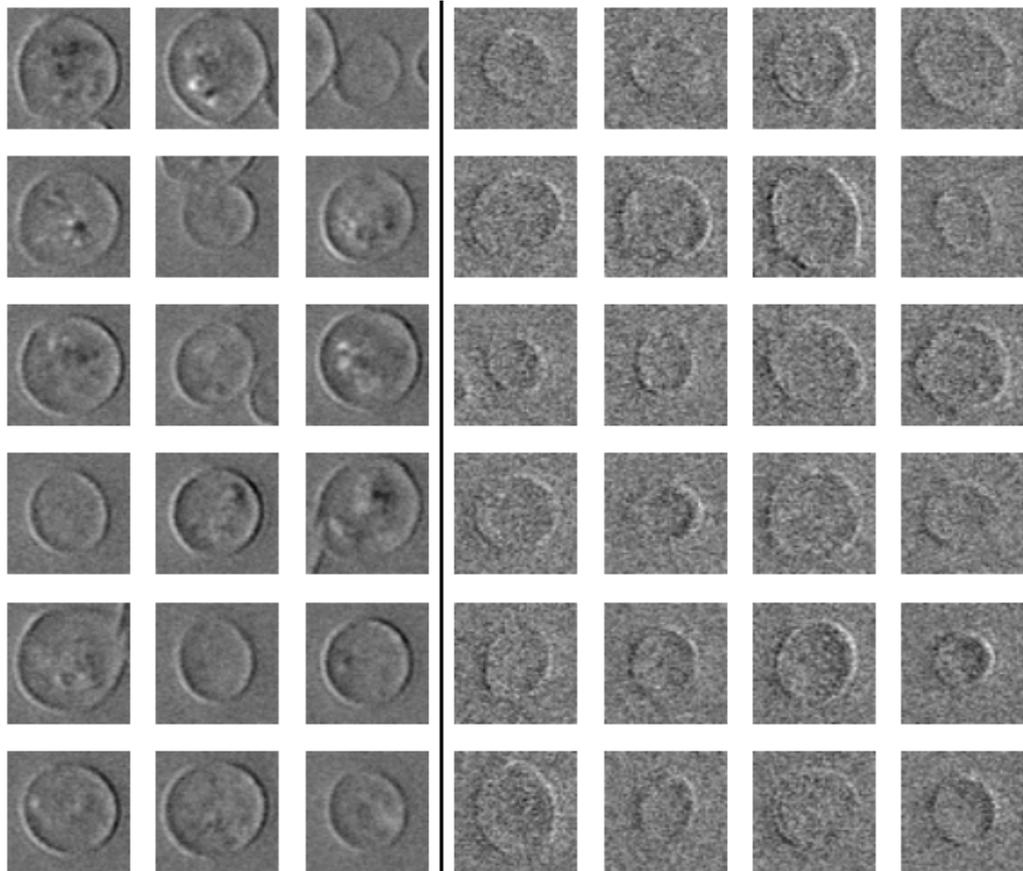
Fig. 5-3: Patch samples with a size of 51 x 51 pixels extracted from the 2nd PC images of live and dead cells. The three columns at the left hand side contain the live cell patches, and the four columns at the right hand side contain those from dead cells.

In order to train the SVM classifier, all-live cultures and all-dead cultures are prepared, and subsequently, T time series images are taken for the cells in these cultures. With regards to each cell in the images, as illustrated above, T PC patches are computed, and the j-th PC patch is used as the feature of the cell. By selecting different j, the performance with regards to different PC patch features can be compared. For each live cell, its certain PC patch is labeled with "1", and this labeled PC patch serves as one example of the positive training set. Similarly, the certain PC patch labeled with "-1" of each dead cell is put into the negative training set. In this way the SVM classifier can be trained for testing new images.

Gaussian Mixture Model (GMM) Classifier

A GMM classifier is used to compare the performance of the SVM classifier. In a GMM (Bishop, 1995), a probability density function is expressed as a linear combination of several basis functions in the Gaussian form. A model with M components is written as

$$p(\mathbf{x}) = \sum_{j=1}^{M} p(\mathbf{x}|j)P(j) \tag{1}$$

in which

$$\sum_{j=1}^{M} P(j) = 1, \text{ and } 0 \le P(j) \le 1$$

If the Gaussian components each have a covariance matrix which is some scalar multiple of the identity matrix so that $\boldsymbol{\Sigma}_j = \sigma_j^2 \mathbf{I}$, and hence

$$p(\mathbf{x}|j) = \frac{1}{\left(2\pi\sigma_j^2\right)^{d/2}} \exp\left(-\frac{\left\|\mathbf{x} - \boldsymbol{\mu}_j\right\|^2}{2\sigma_j^2}\right) \tag{2}$$

in which d denotes the dimensionality, then the mixture density contains the following adjustable parameters: $P(j)$, $\boldsymbol{\mu}_j$ and σ_j (where $j = 1, \ldots, M$). The method for determining the parameters of a GMM from a dataset containing K vectors is based on minimising the negative log-likelihood given by

$$E = -\sum_{n=1}^{K} \ln p(\mathbf{x}^n) = -\sum_{n=1}^{K} \ln\left\{\sum_{j=1}^{M} p(\mathbf{x}^n|j)P(j)\right\} \tag{3}$$

Because the likelihood is a differentiable function of the parameters, it is possible to use a so-called EM (Expectation-Maximisation) algorithm (Dempster *et al.*, 1977) to find the minima of E. The EM algorithm iteratively modifies the

GMM parameters to decrease E. When the old parameters are replaced by the new ones, from Eq. (3), the change in the likelihood can be written in the form

$$E^{\text{new}} - E^{\text{old}} = -\sum_n \ln\left\{\frac{p^{\text{new}}(\mathbf{x}^n)}{p^{\text{old}}(\mathbf{x}^n)}\right\}$$

Using Eq. (1), this can be rewritten as

$$E^{\text{new}} - E^{\text{old}} = -\sum_n \ln\left\{\frac{\sum_j P^{\text{new}}(j)p^{\text{new}}(\mathbf{x}^n \mid j)}{p^{\text{old}}(\mathbf{x}^n)} \frac{P^{\text{old}}(j \mid \mathbf{x}^n)}{P^{\text{old}}(j \mid \mathbf{x}^n)}\right\} \qquad (4)$$

According to Jensen's inequality, given a set of numbers λ_j such that $\sum_j \lambda_j = 1$,

$$\ln\left(\sum_j \lambda_j x_j\right) \geq \sum_j \lambda_j \ln(x_j)$$

Since $P^{\text{old}}(j|\mathbf{x})$ sum to unity, from Eq. (4) it yields

$$E^{\text{new}} - E^{\text{old}} \leq -\sum_n \sum_j P^{\text{old}}(j \mid \mathbf{x}^n) \ln\left\{\frac{P^{\text{new}}(j)p^{\text{new}}(\mathbf{x}^n \mid j)}{p^{\text{old}}(\mathbf{x}^n)P^{\text{old}}(j \mid \mathbf{x}^n)}\right\} \qquad (5)$$

Discarding terms with regard to the 'old' parameters, the right-hand side of (5) can be written in the form

$$Q = -\sum_n \sum_j P^{\text{old}}(j \mid \mathbf{x}^n) \ln\left\{P^{\text{new}}(j)p^{\text{new}}(\mathbf{x}^n \mid j)\right\}$$

and the smallest value for the upper bound is found by minimizing this quantity. In a GMM,

$$Q = -\sum_n \sum_j P^{\text{old}}(j \mid \mathbf{x}^n)\left\{\ln P^{\text{new}}(j) - d \ln \sigma_j^{\text{new}} - \frac{\|\mathbf{x}^n - \boldsymbol{\mu}_j^{\text{new}}\|^2}{2(\sigma_j^{\text{new}})^2}\right\} + \text{const.} \quad (6)$$

Consider $\sum_j P^{\text{new}}(j) = 1$, the minimization of this function with regard to $P^{\text{new}}(j)$ is done by introducing a Lagrange multiplier λ and minimizing the function

$$Q + \lambda\left(\sum_j P^{\text{new}}(j) - 1\right) \qquad (7)$$

Setting the derivatives of (7) with respect to $P^{\text{new}}(j)$ to zero, it yields

$$0 = \sum_n \frac{P^{\text{old}}(j \mid \mathbf{x}^n)}{P^{\text{new}}(j)} + \lambda \qquad (8)$$

Using $\sum_j P^{\text{new}}(j) = 1$ and $\sum_j P^{\text{old}}(j|\mathbf{x}^n) = 1$, it yields $\lambda = K$. Then the final update equations of the GMM are

$$P^{\text{new}}(j) = \frac{1}{K} \sum_n P^{\text{old}}(j \mid \mathbf{x}^n)$$

$$\boldsymbol{\mu}_j^{\text{new}} = \frac{\sum_n P^{\text{old}}(j \mid \mathbf{x}^n)\mathbf{x}^n}{\sum_n P^{\text{old}}(j \mid \mathbf{x}^n)} \tag{9}$$

$$(\sigma_j^{\text{new}})^2 = \frac{1}{d} \frac{\sum_n P^{\text{old}}(j \mid \mathbf{x}^n)\left\| \mathbf{x}^n - \boldsymbol{\mu}_j^{\text{new}} \right\|^2}{\sum_n P^{\text{old}}(j \mid \mathbf{x}^n)}$$

When the EM algorithm converges, a GMM classifier can be built in the following way: $\mathbf{x}$ is assigned to class C_k if

$$P(k \mid \mathbf{x}) \geq P(j \mid \mathbf{x}), \quad \text{for all } j \neq k$$

in which $P(j \mid \mathbf{x})$ is computed using

$$P(j \mid \mathbf{x})p(\mathbf{x}) = p(\mathbf{x} \mid j)P(j)$$

in addition with Eq. (1) and (2) as

$$P(j \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid j)P(j)}{p(\mathbf{x})} = \frac{P(j)}{\left(2\pi\sigma_j^2\right)^{d/2}} \exp\left(-\frac{\left\| \mathbf{x} - \boldsymbol{\mu}_j \right\|^2}{2\sigma_j^2}\right) \cdot \frac{1}{p(\mathbf{x})} \tag{10}$$

Consider for all j, $p(\mathbf{x})$ is the same, it can be computed simply as

$$\widetilde{P}(j \mid \mathbf{x}) = p(\mathbf{x} \mid j)P(j) = \frac{P(j)}{\left(2\pi\sigma_j^2\right)^{d/2}} \exp\left(-\frac{\left\| \mathbf{x} - \boldsymbol{\mu}_j \right\|^2}{2\sigma_j^2}\right) \tag{11}$$

Results

Results Using SVM classifier

The PC images and PC patches of the cells are computed subsequently to represent the time series information of the live and dead cells. For each one of the total 24 samples, PC patches of 20 cells are randomly selected to generate the datasets. The profile of these datasets are shown in Table 5-1. Each of this datasets includes either only live cells, denoted as L_i ($i =1,2, \ldots 12$), or only dead cells, denoted as D_i ($i =1,2, \ldots 12$).

Table 5-1: Profiles of the datasets (dark-field illumination, normal microscope, 40x objective lens, about 2 seconds per image, taken at 5 successive time points).

Dataset	Property	# total cells	# selected cells
L_1	Live	76	20
L_2	Live	51	20
L_3	Live	76	20
L_4	Live	60	20
L_5	Live	50	20
L_6	Live	67	20
L_7	Live	67	20
L_8	live	63	20
L_9	live	64	20
L_{10}	live	56	20
L_{11}	live	75	20
L_{12}	live	67	20
sum	-	772	240
D_1	dead	61	20
D_2	dead	52	20
D_3	dead	55	20
D_4	dead	47	20
D_5	dead	50	20
D_6	dead	45	20
D_7	dead	37	20
D_8	dead	59	20
D_9	dead	40	20
D_{10}	dead	46	20
D_{11}	dead	44	20
D_{12}	dead	46	20
sum	-	582	240
Γ	-	-	480

In order to test the classification performance on these dataset, one of the most popular methods, cross validation, is used to investigate the whole system. The *complete dataset* $\Gamma = \bigcup_k L_k D_k$, $k = 1,2\ldots12$, is divided into six disjoint subsets, $\Gamma = \bigcup_k \Gamma_k$, $k = 1,2\ldots6$, each of these subsets has equal number of positive and negative items (live and dead PC patches). The profile of these subsets is shown in Table 5-2. One of these subsets are picked out to be the test set, Γ_{te}; while the other five subsets are treated as the training set, Γ_{tr}. An SVM classifier with Gaussian kernel is trained with Γ_{tr}, and then it is used to classify the items in Γ_{te} into two different groups (live or dead), and the accuracy of the classification can be

evaluated in the presence of prior knowledge of the true viability of the cells. This process is repeated five more times with each Γ_k being treated in turn as the test set, Γ_{te}, while the other five are treated as Γ_{tr}. In this way, six cross validation results can be obtained.

Table 5-2: Profiles of the six disjoint subsets of the whole dataset shown in Table 5-1.

Dataset	# of PC patches of live cells	# of PC patches of dead cells
Γ_1	20	20
Γ_2	20	20
Γ_3	20	20
Γ_4	20	20
Γ_5	20	20
Γ_6	20	20

Four ROC parameters are used to describe the classification results, which are given as

TP: Positive cases that are classified as positive (true positive). It denotes the number of correctly recognized live cells.

FN: Positive cases that are classified as negative (false negative). It denotes the number of cells that are really live but classified as dead.

TN: Negative cases that are classified as negative (true negative). It denotes the number of correctly recognized dead cells.

FP: Negative cases that are classified as positive (false positive). It denotes the number of cells that are really dead but classified as live.

The test error of the whole cross validation can also be computed by averaging the six individual test errors (FN plus FP). We tested the system first using the 2nd PC patch of the cells in the dataset shown in Table 5-1. According to the results shown in Fig. 5-4, it is clear that the system has a best performance with a test error of 16% (namely, an accuracy of 84%) when the width of the Gaussian kernel is taken as $\sigma = 2$. In this case, the results of six individual cross validation cycles are shown in Fig. 5-5.
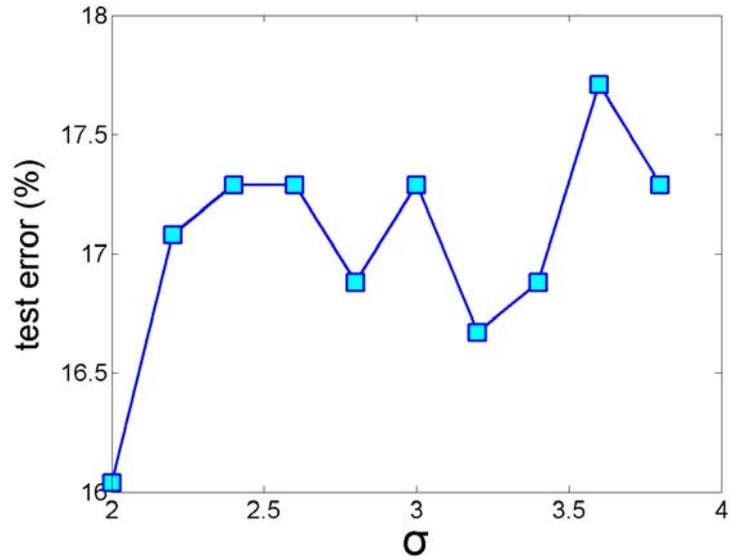
Fig. 5-4: Dependence of the test error with the width of the Gaussian kernel of the SVM classifier.
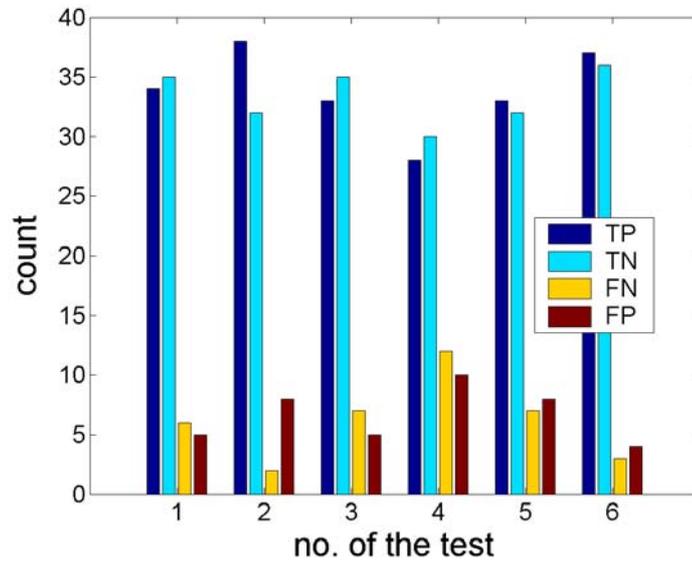
Fig. 5-5: Results of cross validation on the complete dataset $\Gamma = U_k L_k D_k$, $k = 1, 2 \ldots 12$. The complete dataset is divided into six disjoint subsets.

When a series of cell images is taken at different time, it is very critical to guarantee that there should be no significant cell movement, namely, each cell should be fixed at a determined position without rotation or displacement. The side effect of significant cell movements is demonstrated in Fig. 5-6, in which the 2nd PC image of the time series images is shown.
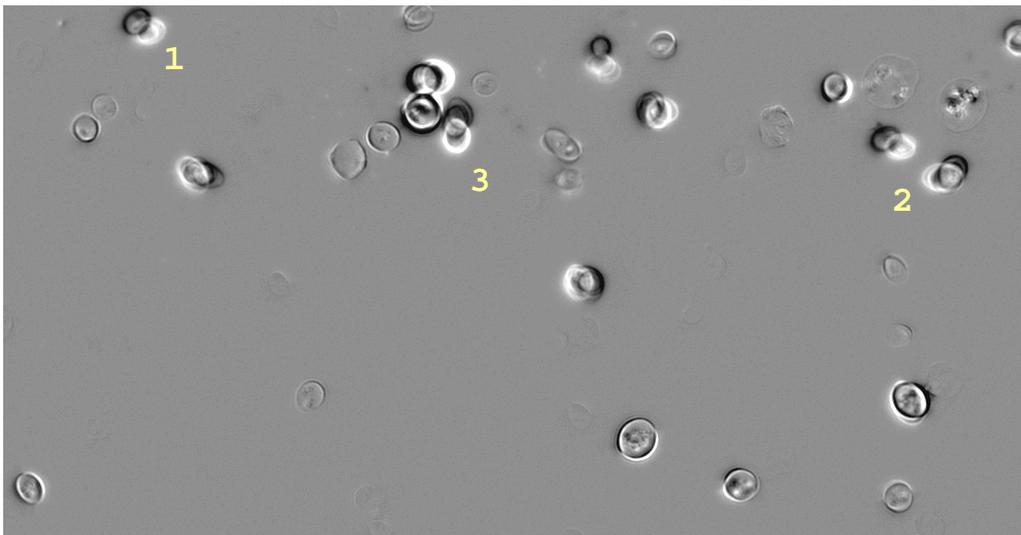
Fig. 5-6: The 2nd PC projection in the presence of significant cell movements.

In the process of taking the images, we didn't wait long enough to get the cells immobilized, thus, significant cell movement, including rotation and displacement, can still be seen in the images at different time points. In Fig. 5-6, the cells labeled with "1" and "2" and "3" are examples subjected to large movements. Thus, they exhibit a different appearance to all the other cells. One of the side effects with it might be, that it might make live and dead cells look very similar to each other in the PC images. In consequence, it might cause misclassification. That could also be one of the reasons why at best only an accuracy of 83% is obtained from the cross validation (refer to Fig. 5-4 and Fig. 5-5).

In order to overcome this obstacle, a higher frame rate is recommended when capturing cell images. Thus, we have switched to an advanced digital camera (CV-M10RS, JAI Inc., Japan) that can take time series images automatically with a much higher frame rate. With this camera, rates of 30 fps (frames per second) can be achieved.

Thereby, new experiments are possible by taking images at higher frame rates.

The temporal interval between any two successive images is 1/30 second, and images at 10 successive time points are taken. Time series images have been taken for six samples of the all-live and all-dead cultures, respectively. The profile of the dataset is listed in Table 5-3.

Table 5-3: Profiles of the datasets (dark-field illumination, normal microscope, 40x objective lens, 35 frames per second, images taken at 10 successive time points, patch size: 15 x 15 pixels).

Dataset	Property	# total cells	# selected cells
L_1'	live	22	22
L_2'	live	37	37
L_3'	live	28	28
L_4'	live	45	45
L_5'	live	40	40
L_6'	live	36	36
sum	-	208	208
D_1'	dead	31	31
D_2'	dead	36	36
D_3'	dead	28	28
D_4'	dead	34	34
D_5'	dead	29	29
D_6'	dead	27	27
sum	-	185	185

The cross validation results derived from these datasets are summarized in Fig. 5-7, in which the classification results derived from not only the 2nd, but also the other order of PC patches are shown. In contrast to the best case in Fig. 5-4, when 2nd PC patches are taken into account as the feature of distinguishing live and dead cells, an increase of accuracy from 84.0% to 89.3% can be seen. Moreover, it can also be seen that the 1st PC patches are superior to the 2nd ones in viability classification, the former leading to an accuracy of around 96%. In addition, it can be seen that the PC patches of the other orders (3rd…10th) have led to lower classification accuracy.
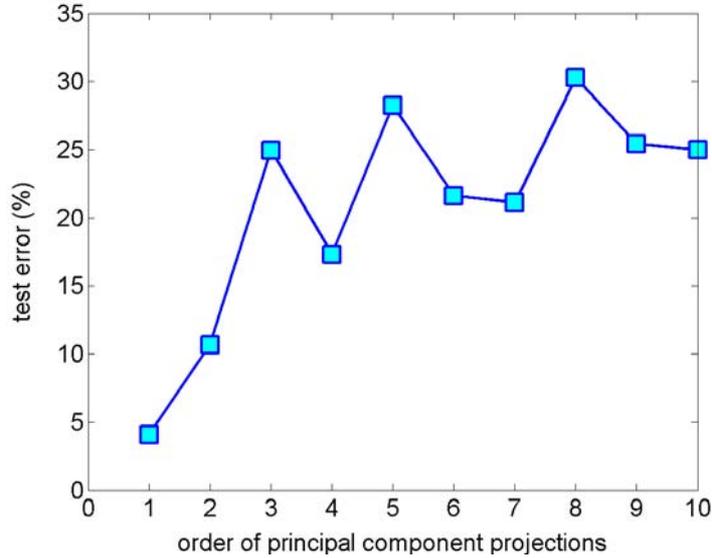
Fig. 5-7: Dependence of the test error on the order of the principal component projections that is taken as the feature to distinguish live and dead cells.

In previous attempts, the PC patches were computed for each of the sub-datasets (L_k, D_k, $k = 1,2…12$) separately (denoted as *total PCA* in Fig. 5-8). Despite the inspiring results obtained, one more reasonable way of performing PCA is that the eigenvectors are computed with the data points in the training set, while the PC patches of the test set are computed by projecting to those eigenvectors. We simulate this situation by an experiment that carries out PCA on part of the complete dataset (denoted as *partial PCA* in Fig. 5-8), and projects the rest to the determined PC space. In this case, $L_1' \sim L_5'$ and $D_1' \sim D_5'$, described in Table 5-3, have been used to compute the eigenvectors; while L_6' and D_6' are only projected to the space constructed by those computed eigenvectors. The results in this case (Fig. 5-8) can be compared with those recorded in Fig. 5-7. As can be seen, there is little difference between these two strategies. This could be an argument for supporting the strategy that new captured cell images need simply to be adapted to the already set up space constructed by the eigenvectors derived from the training sets, and the eigenvectors and eigenvalues need not be computed for the testing set any more, thus consuming less computation time. This tactic will not deteriorate the system's performance, at least in our attempt. For the case of 1st or 2nd PC patch features, the classification accuracy does not decline; on the contrary, it is even slightly better.
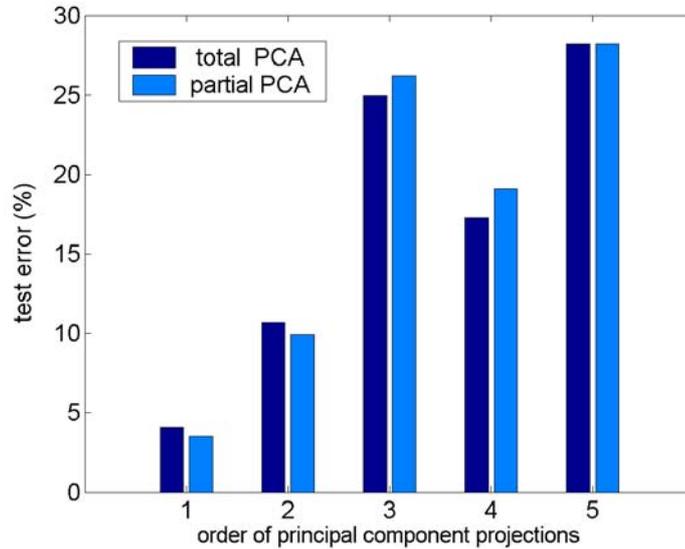
Fig. 5-8: Classification results of total PCA and partial PCA.

Another important issue that should be taken into account is, while it is easy to obtain the images of live cells for training the SVM classifier, it is not that easy to collect the negative training examples (dead cells), to let the SVM classifier learn the distinct features of dead cells. This is due to the large variety of causes for cell death in the practice. Those reasons include over-age, physical damage, gene-regulated death, shortage of nutrition, and so on. Thus, it is difficult to simulate all the possible situations of cell death. In our experiments the cells are killed in a water bath at 70 °C for two hours. Therefore, if only the images of so-killed dead cells are used as the negative samples of training sets, the SVM classifier could not have the optimal performance, since the visual appearance of the dead cells that die from other reasons may be different from those that die from heating. Allowing for this, we have switched our attention from the two-class SVM classifier to a one-class SVM classifier, which takes only positive samples (live cells) as training set, and makes its decision on cell viability only based on that. The main advantage of this technique is no need for collecting a negative training set.

A one-class SVM classifier has been applied in cross-validation of those datasets listed in Table 5-1. We compare its performance with that of the two-class classifier (refer to Fig. 5-7) in Fig. 5-9. It is clear that the two-class SVM classifier outperforms the one-class classifier when the 1st and 2nd PC patches are taken as

the features. When higher order of PC patches are used, the one-class SVM classifier performs steadily around the accuracy of about 85%. In comparison, in these cases, the two-class SVM classifier doesn't work well enough. The best performance of the one-class SVM classifier is observed with the 3rd PC patch features, with an accuracy of 88.9%, which is very close to the best case of the two-class SVM classifier with the 2nd PC patches. Therefore, it suggests that the one-class SVM classifier can also be used to classify the cells on the basis of taking the 3rd PC patches as the discriminative features.
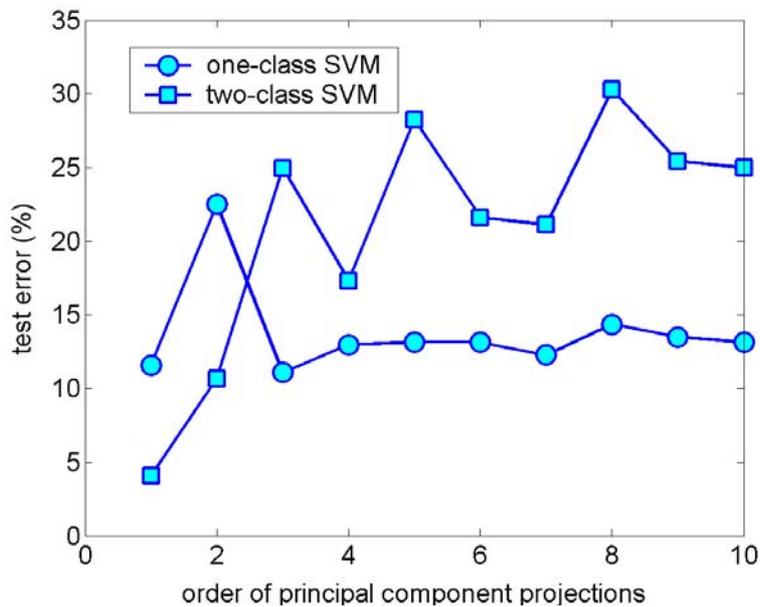
Fig. 5-9: Comparison of the performance of the one-class SVM classifier and the two-class SVM classifier.

Results Using GMM classifier

Full Length Vectors
The same datasets, as described in Table 5-3, have also been used to evaluate a GMM classifier. In these tests, $L_1' \sim L_5'$ and $D_1' \sim D_5'$ are used to compute the eigenvectors; while L_6' and D_6' are only projected to the space constructed by those computed eigenvectors, in other words, it is performed in the partial PCA way. For convenience, various orders of PC patches as denoted as follows:

The 1st PC patches: $PT_1 = \{ pt_{1i} \}$, $(i = 1, 2 \ldots 393, pt_{1i} \in \mathfrak{R}^{961})$
The 2nd PC patches: $PT_2 = \{ pt_{2i} \}$, $(i = 1, 2 \ldots 393, pt_{2i} \in \mathfrak{R}^{961})$
The 3rd PC patches: $PT_3 = \{ pt_{3i} \}$, $(i = 1, 2 \ldots 393, pt_{3i} \in \mathfrak{R}^{961})$
The 4th PC patches: $PT_4 = \{ pt_{4i} \}$, $(i = 1, 2 \ldots 393, pt_{4i} \in \mathfrak{R}^{961})$
The 5th PC patches: $PT_5 = \{ pt_{5i} \}$, $(i = 1, 2 \ldots 393, pt_{5i} \in \mathfrak{R}^{961})$

Table 5-4: Performance of the GMM classifier with the datasets listed in Table 5-3.

Dataset	#Compon.	TP	FN	FP	TN	CR(%)
PT_1	1	166	42	14	171	85.8
	2	179	29	32	153	84.5
	3	187	21	44	141	83.5
	4	193	15	35	150	87.3
	5	-	-	-	-	-
PT_2	1	169	39	61	124	74.6
	2	168	40	82	103	69.0
	3	-	-	-	-	-
PT_3	1	105	103	125	60	42.0
	2	87	121	84	101	47.8
	3	-	-	-	-	-
PT_4	1	111	97	114	71	46.3
	2	106	102	99	86	48.9
	3	-	-	-	-	-
PT_5	1	-	-	-	-	-
	2	162	46	98	87	63.4
	3	-	-	-	-	-

Here PT_k ($k = 1, \ldots 5$) is the set of the k-th PC patches ($pt_{k,i}$) of the 393 cells, in which 208 are live, and the other 185 are dead.

GMM method is applied to PT_k, and the classifiers' performances are listed in Table 5-4. In simulating the distribution of the data points, GMM models with different number of Gaussian distributions (components) are used. The results reveal that only PT_1 leads to good classification performance. In particular, when the number of the Gaussian components is 4, it yields best performance, with the highest accuracy (classification rate, CR) of 87.3%; while the GMM classifiers based on other PC patches, i.e. $PT_2 \sim PT_5$, do not show inspiring classification performance.

One may notice, in the computational attempts mentioned above, the full-sized vector $pt_{ni} \in \mathfrak{R}^{961}$ of each cell patch has been used. Due to the great dimensionality, the GMM modeling and EM optimization process could not be accomplished in some cases. For instance, when a 5-component GMM model is used to simulate

the density of the data from PT_1, the EM process cannot converge at all, which leads to the blank of the results at corresponding positions. In order to get the program well converging, we have performed another PCA to the PC patches, aiming at reducing the dimensionality of the dataset.

To this end, the vectors from PT_k, whose dimensionality is 31 x 31 = 961, are processed with PCA to reduce the dimensionality.
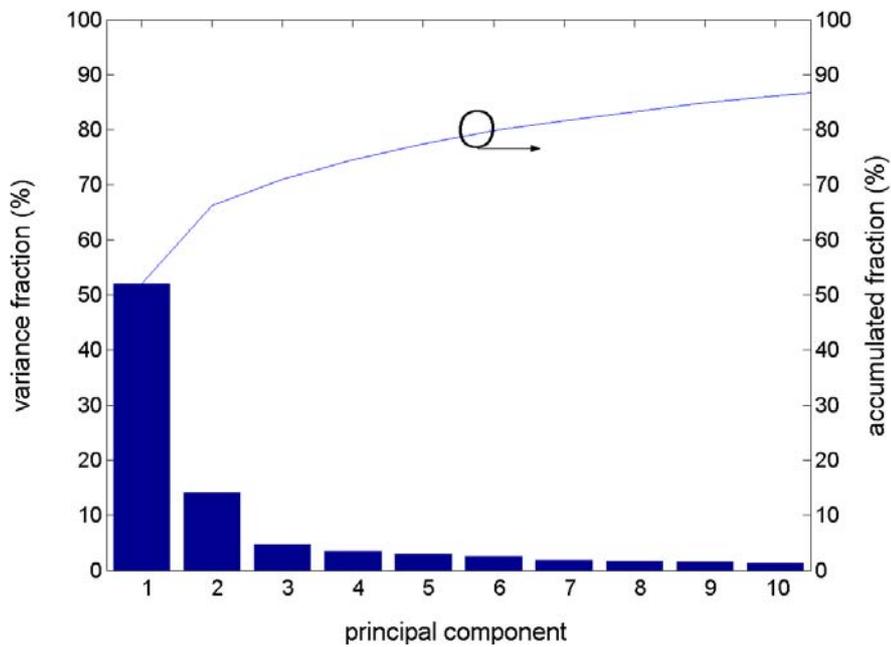
Fig. 5-10: Variance of each principal component of the vectors in PT_1.

Dimensionality Reduced Vectors
PCA is carried out on the vectors in PT_1, and the eigenvalues are shown in Fig. 5-10. When dimensionality of those vectors are reduced from 961 to d_r, and various number of Gaussian components are used to simulate the data density, the final classification results are shown in Table 5-5.

Table 5-5: Performance of the GMM classifier with the datasets listed in Table 5-3, using PT_1 as distinguishing feature, with reduced dimension of the vectors.

d_r	#Compon.	TP	FN	FP	TN	CR(%)
1	1	172	36	113	72	62.1
	2	153	55	108	77	58.5
	3	156	52	114	71	57.8
	5	149	59	107	78	57.8
	9	138	70	107	78	55.0
2	1	167	41	30	155	82.0
	2	166	42	40	145	79.1
	3	171	37	45	140	79.1
	5	180	28	42	143	82.2
	9	173	35	43	142	80.2
4	1	170	38	17	168	86.0
	2	170	38	46	139	78.6
	3	186	22	35	150	85.5
	5	195	13	25	160	90.3
	9	186	22	34	151	85.8
6	1	169	39	12	173	87.0
	2	175	33	52	133	78.4
	3	192	16	28	157	88.8
	5	192	16	19	166	91.1
	9	192	16	32	153	87.8

The best performance occurs when target dimensionality is about 6 and a 5-component GMM model is used to simulate the density distribution of the datasets, with classification rate above 91%, which is even better than without dimensionality reduction, as shown in Table 5-4.

The results based on PT_2 are shown in analogy in Fig. 5-11 and Table 5-6. However, in all of these cases, the classification rate (accuracy) is lower than 80%.

Table 5-6: Performance of the GMM classifier with the datasets listed in Table 5-3, using PT_2 as distinguishing feature, with reduced dimension of the vectors.

d_r	#Compon.	TP	FN	FP	TN	CR(%)
1	1	196	12	67	118	79.9
	2	151	57	76	109	66.2
	3	145	63	72	113	65.6
	5	144	64	67	118	66.7
	9	147	61	70	115	66.7
2	1	179	29	60	125	77.4
	2	146	62	62	123	68.4
	3	148	60	70	115	66.9
	5	157	51	69	116	69.5
	9	154	54	65	120	69.7
4	1	176	32	61	124	76.3
	2	157	51	74	111	68.2
	3	166	42	67	118	72.3
	5	172	36	77	108	71.2
	9	162	46	77	108	68.7
6	1	170	38	61	124	74.8
	2	154	54	80	105	65.9
	3	177	31	70	115	74.3
	5	162	46	69	116	70.7
	9	164	44	65	120	72.3
	9	162	46	73	112	69.7
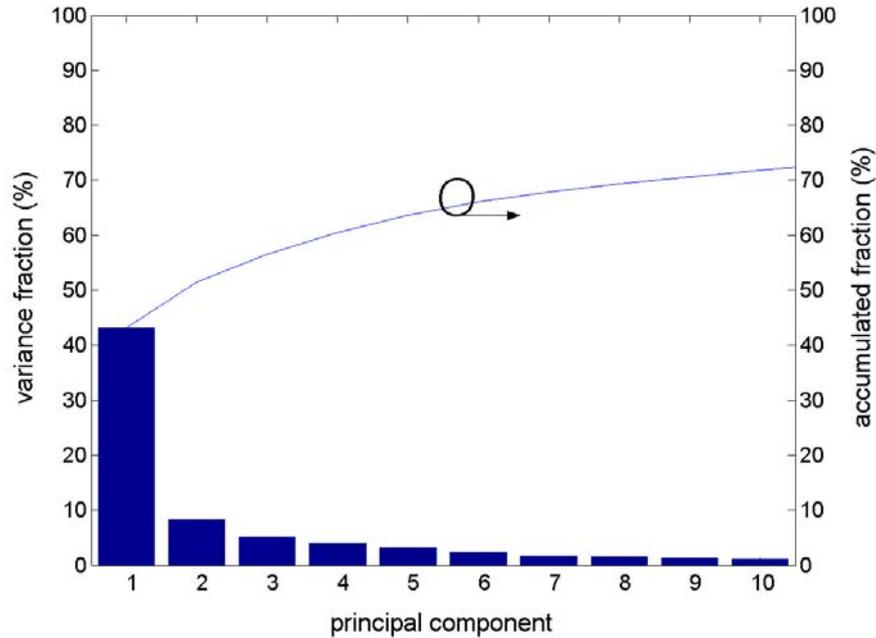
Fig. 5-11: Variance of each principal component of the vectors in PT_2.

So it can be seen that if GMM classifiers are to be applied to distinguish live and dead cells, the 1st PC patches should be used. PC patches of other orders are of no use, as they lead to lower performance. These observations are quite different from the conclusions derived from the SVM classifier, as shown in Fig. 5-9.

The results derived from the aforementioned different strategies are summarized in Fig. 5-12 and Fig. 5-13 for comparison. From Fig. 5-12, it can be seen that

1. It is better to take the 1st PC patches from the time-series images as the features for distinguishing live and dead cells than using the 2nd PC patches, as the six methods all perform well when the 1st PC patches are taken as the main feature.
2. If 2nd PC patches are taken as the features, only H2F, H2P and H1P have led to good classification results, provided that an accuracy of at least 85% is to be reached. It implies that higher frame rates in capturing images should be applied, as the classification accuracy can therewith be increased from 84% to around 90%.
3. SVM classifiers outperform GMM classifiers in our attempts of distinguishing live and dead cells of brewery's yeast *Saccharomyces cerevisiae*.
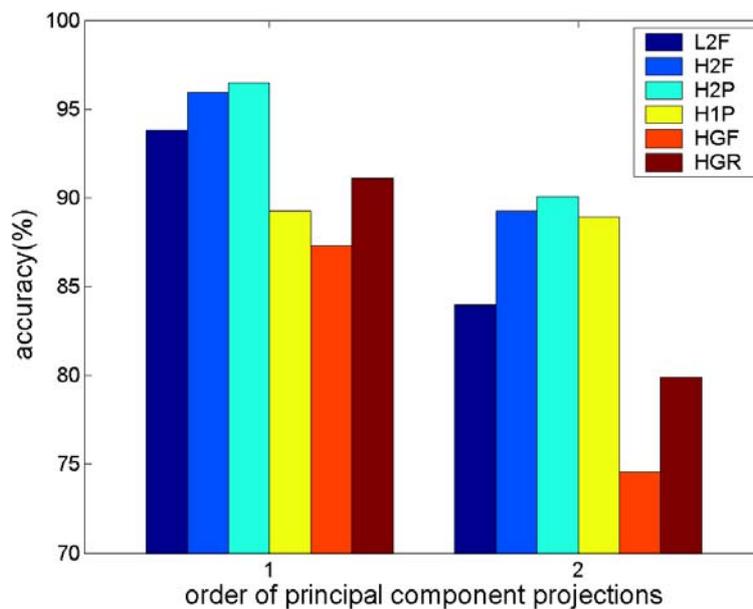
Fig. 5-12: Comparisons of results obtained based on six different experimental procedures. L2F: low frame rate photographing, 2-class SVM; H2F: high frame rate photographing, 2-class SVM; H2P: high frame rate photographing, 2-class SVM, partial PCA; H1P: high frame rate photographing, 1-class SVM; HGF: high frame rate photographing, GMM, full dimension; HGR: high frame rate photographing, GMM, reduced dimension.
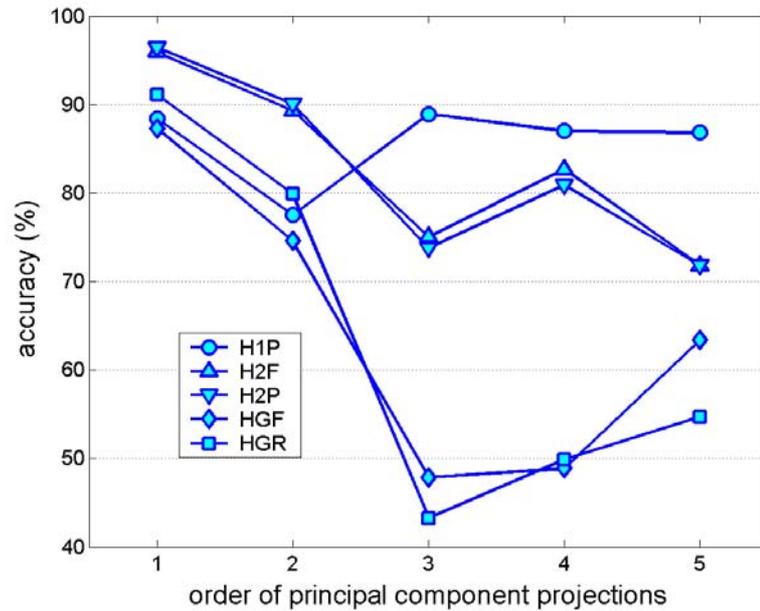
Fig. 5-13: Comparison of results obtained from five experiments with high-rate photographing. Features were extracted from different orders of principal components. H2F: high frame rate photographing, 2-class SVM; H2P: high frame rate photographing, 2-class SVM, partial PCA; H1P: high frame rate photographing, 1-class SVM; HGF: high frame rate photographing, GMM, full dimension; HGR: high frame rate photographing, GMM, reduced dimension.

From Fig. 5-13, it can be seen that

1. 2-Class SVM classifiers outperform 1-Class SVM classifiers when low order (1st and 2nd) PC patches are used as features of distinguishing cells; and it outperforms GMM classifiers in all cases.

2. The 1-Class SVM classifiers outperform the other two types of classifiers with higher orders of PC patches, despite a bad performance with the 1st and 2nd PC patches. In particular, when 3rd, 4th and 5th PC patches are taken as features, only the 1-Class SVM classifiers can achieve an accuracy higher than 85%, while the other two types of classifiers fail in these cases.

3. The GMM classifiers have worse performance in almost all cases, except for the case with 1st PC patches. It is clear that this type of classifier does not even achieve accuracies better than 70% when higher-order PC patches are considered.

Abbreviations

EM:	Expectation-Maximisation
FN:	False Negative
FP:	False Positive
GMM:	Gaussian Mixture Model
PCA:	Principal Component Analysis
SVM:	Support Vector Machine
TN	True Negative
TP:	True Positive

List of Symbols

T:	number of the images in the time series
I_j:	image at time point j
m:	number of pixels in an image
$p_{i,j}$:	the i-th pixel in the image at time point j
$v_{i,j}$:	the grayscale value of the i-th pixel in the image at time point j
$\mathbf{v}_i$:	time series vector of the i-th pixel
e_j:	eigenvector
c_j:	eigenvalue
$\mathbf{v}^e_i$:	the principal component vector of the i-th pixel
$v^e_{i,j}$:	the j-th principal component of $\mathbf{v}^e_i$
$\mathbf{I}^e_j$:	the j-th principal component image
$\mathbf{x}$:	a sample of a dataset
$p(\mathbf{x})$:	probability of $\mathbf{x}$
$p(\mathbf{x}\mid j)$:	probability of $\mathbf{x}$ under the condition that $\mathbf{x}$ belongs to class j
$P(j)$:	probability of class j
$\boldsymbol{\mu}_j$:	mean of class j
σ_j:	variance of class j
d:	dimensionality
E:	log-likelihood
λ:	lagrange multiplier
C_k:	the k-th class
L_i:	live cell dataset
D_i:	dead cell dataset
Γ:	complete dataset
Γ_k:	subset of Γ in cross validation
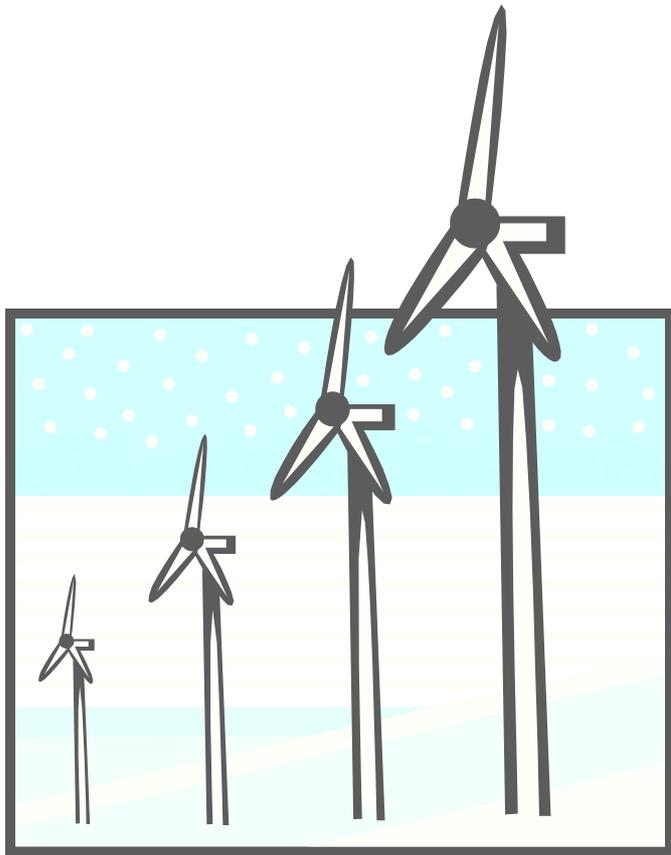
Chapter 6

Viability Classification in Time Series Micrographs Using Object Tracking and Dynamics Operators

In this chapter the similar feature selection technique, which has been applied in Chapter 4 for cell viability classification on one single micrograph, will be employed to investigate the possibility of distinguishing live and dead cells on TSI (Time Series Images) captured by the MVS (Machine Vision System). The basic idea is: if some features of live cells change with time in a different manner from those of dead cells, this different pattern of feature change might be used to classify the viability of a cell. Our solution to achieving this goal is composed of two parts: cell tracking and feature selection based on dynamics operators.

Material and Methods

Cell Movement Problem

In capturing TSI, it may not be possible to avoid movements of the cells. In this sense, the position of any cell will be shifted (for instance, by the impact of a stirrer) on the TSI. Three examples of TSI of a single cell are shown in Fig. 6-1. The large shift of the cell's position can be clearly observed. Allowing for this, cell tracking must be performed to find the corresponding position of each cell on each single image of the TSI before computing the features of it, or, otherwise it would not make sense to compute the "features" of the image patches like those at time t_5 in Fig. 6-1.
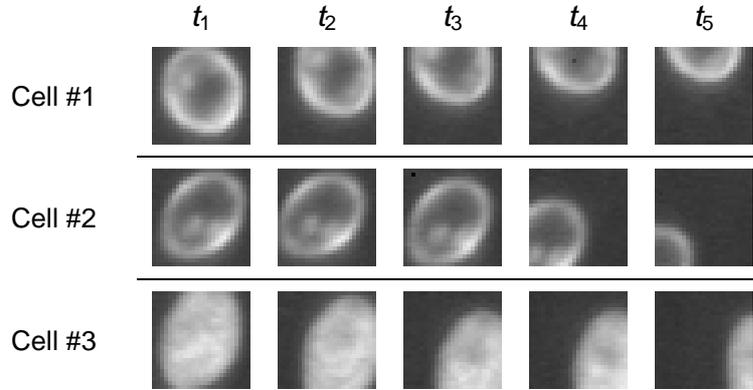
Fig. 6-1: Cell movements in time series images.

Gabor Features

Gabor filters are commonly used for feature extraction in various pattern recognition applications, as their filtering characteristics can be fine tuned to achieve good localization properties in the frequency domain. The general functional form of a 2D Gabor filter in space and frequency domain (Daugman, 1988) is given by:

$$g_a(x,y) = \exp\left(-\frac{x^2 + y^2}{2\sigma_s^2}\right)\cos(\omega_1 x + \omega_2 y)$$

$$G_a(u,v) = A\left\{\exp\left[-\frac{(u-\omega_1)^2 + (v-\omega_2)^2}{2\sigma_f^2}\right] + \exp\left[-\frac{(u+\omega_1)^2 + (v+\omega_2)^2}{2\sigma_f^2}\right]\right\}$$

$$A = 2\pi\sigma_s^2, \quad \sigma_f = \frac{1}{2\pi\sigma_s}$$

where σ_s is the standard deviation of the Gaussian envelope. The parameters (ω_1, ω_2) define the spatial frequency of a sinusoidal wave, which can also be expressed in polar coordinates as radial frequency ρ and orientation θ.

$$\rho^2 = \omega_1^2 + \omega_2^2, \quad \tan\theta = \omega_2 / \omega_1$$

Suitable parameters of five Gabor filters ($Gf_0 \sim Gf_4$) for the images have been found, which are shown in Table 6-1. The spatial functional forms of these filters are shown in Fig. 6-2.

Table 6-1: Suitable parameters of the implemented Gabor filters.

	ρ	σ_s	θ
Gf_0	0	2.30	-
Gf_1	1.30	2.30	0
Gf_2	1.30	2.30	$\pi/4$
Gf_3	1.30	2.30	$\pi/2$
Gf_4	1.30	2.30	$3\pi/4$

Gf_1 $\qquad$ Gf_2 $\qquad$ Gf_3 $\qquad$ Gf_4 $\qquad$ Gf_5
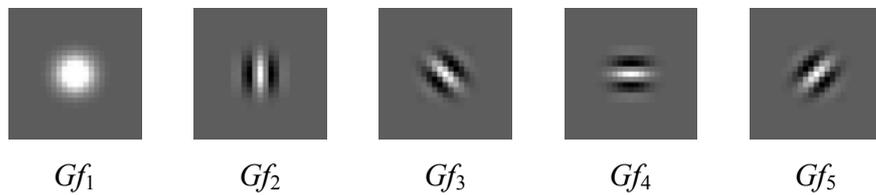
Fig. 6-2: Plot of the five implemented Gabor filters in spatial domain.

Convolution of an N x N sized cell patch with the k-th Gabor filter (Gf_k) leads to the k-th Gabor representation (G_k), which is also of N x N size. An example of the five Gabor representations of a cell is displayed in Fig. 6-3. The energy of the k-th Gabor representation (G_k) is computed as the k-th feature (f_k) of the cell patch.

$$f_k = \frac{\sum_i \sum_j (G_k(i, j))^2}{N^2} \quad , k = 1,2,...5$$

original cell image $\qquad$ G_1 $\qquad$ G_2 $\qquad$ G_3 $\qquad$ G_4 $\qquad$ G_5
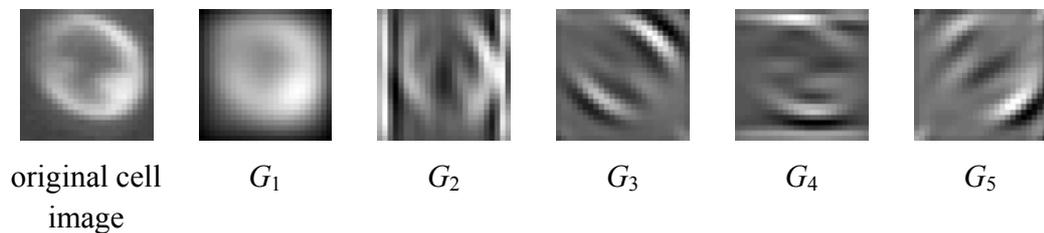
Fig. 6-3: A cell image and corresponding Gabor representations.

Cell Tracking

The idea of cell tracking is demonstrated in Fig. 6-4. In a TSI containing images at T different time points ($I_0, \ldots I_{T-1}$), a cell detection program is run to find all cell positions on I_0. The i-th cell centre on I_0 is denoted as $p_i^{(0)}$. Around $p_i^{(0)}$, an $N \times N$ sized cell patch is picked out and its Gabor feature vector $\boldsymbol{F}_i^{(0)} = (f_{1,i}^{(0)}, \ldots f_{5,i}^{(0)})$ is computed.
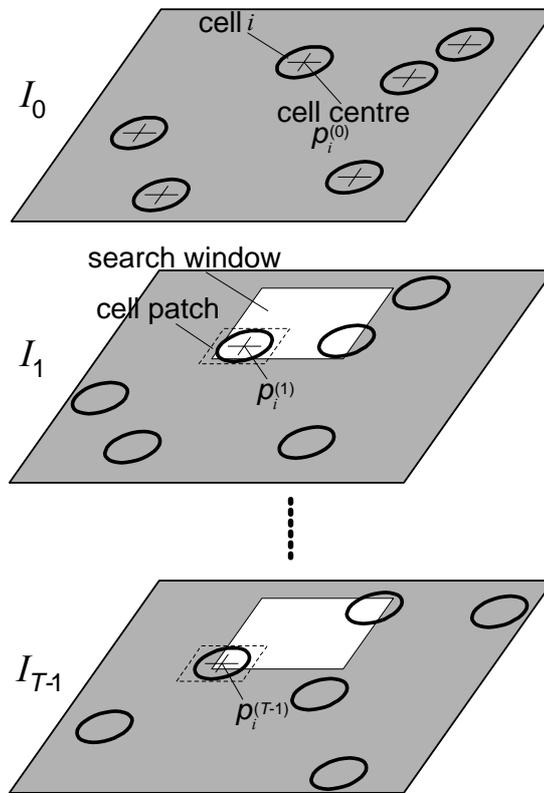
Fig. 6-4: Illustration of the cell tracking procedure.

In order to find the corresponding position of the cell on I_1, an $M \times M$ sized search window centred at $p_i^{(0)}$ is opened. In this search window, the pixel, at which the cell patch is centred, whose Gabor feature vector is most closely matching $\boldsymbol{F}_i^{(0)}$, will be regarded as the new position of the cells ($p_i^{(1)}$). In evaluating how closely a given Gabor feature vector $\boldsymbol{F} = (f_1, \ldots f_5)$ is matching $\boldsymbol{F}_i^{(0)}$, the Euclid distance d shown in Eq. (1) is used. The smaller d, the more closely $\boldsymbol{F}$ is matching $\boldsymbol{F}_i^{(0)}$. This process will also be carried out analogously for images $I_0, \ldots I_{T-1}$.

$$d = \left\| \boldsymbol{F} - \boldsymbol{F}_i^{(0)} \right\|_2 = \sqrt{\sum_{j=1}^{5} (f_j - f_{j,i}^{(0)})^2} \qquad (1)$$

Dynamics Operators

In cell tracking processes, the cell positions in the TSI are found, and denoted as $p_i^{(0)}, \ldots p_i^{(T-1)}$. In accordance, the wavelet feature vectors ($\boldsymbol{W}_i^{(0)}, \ldots \boldsymbol{W}_i^{(T-1)}$), each of which contains 32 wavelet features (refer to Chapter 4), are computed for the corresponding cell patches around $p_i^{(0)}, \ldots p_i^{(T-1)}$.

We use a series of dynamics operators $\mathscr{D}_i$ ($i = 1, \ldots 8$) in order to make use of the dynamic information of the wavelet features in the TSI. We define $\mathscr{D}_i$ as:

$$\mathscr{D}_1(\boldsymbol{W}_i^{(0)}, \ldots \boldsymbol{W}_i^{(T-1)}) = \frac{1}{T-1} \sum_{t=1}^{T-1} \Delta \boldsymbol{W}_i^{(t)} = \frac{1}{T-1} \sum_{t=1}^{T-1} \left(\boldsymbol{W}_i^{(t)} - \boldsymbol{W}_i^{(t-1)} \right)$$

$$\mathscr{D}_2(\boldsymbol{W}_i^{(0)}, \ldots \boldsymbol{W}_i^{(T-1)}) = \frac{1}{T-1} \sum_{t=1}^{T-1} \left| \Delta \boldsymbol{W}_i^{(t)} \right| = \frac{1}{T-1} \sum_{t=1}^{T-1} \left| \boldsymbol{W}_i^{(t)} - \boldsymbol{W}_i^{(t-1)} \right|$$

$$\mathscr{D}_3(\boldsymbol{W}_i^{(0)}, \ldots \boldsymbol{W}_i^{(T-1)}) = \frac{2}{T-1} \sum_{t=1}^{(T-1)/2} \Delta(\Delta \boldsymbol{W}_i^{(2t)})$$

$$= \frac{2}{T-1} \sum_{t=1}^{(T-1)/2} \left[\left(\boldsymbol{W}_i^{(2t)} - \boldsymbol{W}_i^{(2t-1)} \right) - \left(\boldsymbol{W}_i^{(2t-1)} - \boldsymbol{W}_i^{(2t-2)} \right) \right]$$

$$\mathscr{D}_4(\boldsymbol{W}_i^{(0)}, \ldots \boldsymbol{W}_i^{(T-1)}) = \frac{2}{T} \sum_{t=1}^{T/2} \Delta \boldsymbol{W}_i^{(2t)} + \Delta \boldsymbol{W}_i^{(T-1)}$$

$$= \frac{2}{T} \sum_{t=1}^{T/2} \left(\boldsymbol{W}_i^{(2t)} - \boldsymbol{W}_i^{(2t-2)} \right) + \left(\boldsymbol{W}_i^{(T-1)} - \boldsymbol{W}_i^{(T-2)} \right)$$

$$\mathscr{D}_5(\boldsymbol{W}_i^{(0)}, \ldots \boldsymbol{W}_i^{(T-1)}) = \frac{2}{T} \sum_{t=1}^{T/2} \Delta \boldsymbol{W}_i^{(2t)} = \frac{2}{T} \sum_{t=1}^{T/2} \left(\boldsymbol{W}_i^{(2t)} - \boldsymbol{W}_i^{(2t-2)} \right)$$

$$\mathscr{D}_6(\boldsymbol{W}_i^{(0)}, \ldots \boldsymbol{W}_i^{(T-1)}) = \frac{1}{2} \left(\boldsymbol{W}_i^{(2)} - \boldsymbol{W}_i^{(0)} \right)$$

$$\mathscr{D}_7(\boldsymbol{W}_i^{(0)}, \ldots \boldsymbol{W}_i^{(T-1)}) = \left(\boldsymbol{W}_i^{(2)} - \boldsymbol{W}_i^{(1)} \right) - \left(\boldsymbol{W}_i^{(1)} - \boldsymbol{W}_i^{(0)} \right)$$

$$= \boldsymbol{W}_i^{(2)} - 2\boldsymbol{W}_i^{(1)} + \boldsymbol{W}_i^{(0)}$$

$$\mathscr{D}_8(W_i^{(0)}, ... W_i^{(T\text{-}1)}) = \frac{1}{2}\left(W_i^{(4)} - W_i^{(2)}\right) - \frac{1}{2}\left(W_i^{(2)} - W_i^{(0)}\right)$$

$$= \frac{1}{2}\left(W_i^{(4)} - 2\,W_i^{(2)} + W_i^{(0)}\right)$$

By means of the k-th dynamics operators ($\mathscr{D}_k$), the time series wavelet feature vectors ($W_i^{(0)}, ... W_i^{(T\text{-}1)}$) of the i-th cell patch can be transformed into a dynamics wavelet feature vector $W_i(k)$:

$$W_i(k) = \mathscr{D}_k(W_i^{(0)}, ... W_i^{(T\text{-}1)}) \tag{2}$$

From the definitions, it can be seen that the meaning of the dynamics operators are given by those gathered in Table 6-2.

Table 6-2: Meaning of the applied dynamics operators.

$\mathscr{D}_{1, 4, 5, 6}$	1st derivative of the wavelet feature vector
$\mathscr{D}_2$	absolute value of the 1st derivative of the wavelet feature vector
$\mathscr{D}_{3, 7, 8}$	2nd derivative of the wavelet feature vector

$\mathscr{D}_1 \sim \mathscr{D}_5$ aim at using as much information in the TSI as possible, but it can be seen that $\mathscr{D}_1$ uses actually only the feature vectors $W_i^{(T\text{-}1)}$ and $W_i^{(0)}$. In comparison, $\mathscr{D}_4$ and $\mathscr{D}_5$ use much more information in the TSI. On the contrary, $\mathscr{D}_6 \sim \mathscr{D}_8$ try to extract dynamic information only from part of the images in the TSI.

In this work, $T = 5$, thus, the definitions of the dynamics operators are simplified into:

$$\mathscr{D}_1(W_i^{(0)}, ... W_i^{(4)}) = \frac{1}{4}\left(W_i^{(4)} - W_i^{(0)}\right)$$

$$\mathscr{D}_2(W_i^{(0)}, ... W_i^{(4)}) = \frac{1}{4}\sum_{t=1}^{4}\left|W_i^{(t)} - W_i^{(t\text{-}1)}\right|$$

$$\mathscr{D}_3(W_i^{(0)}, ... W_i^{(4)}) = \frac{1}{2}\left(W_i^{(4)} - 2W_i^{(3)} + 2W_i^{(2)} - 2W_i^{(1)} + W_i^{(0)}\right)$$

$$\mathscr{D}_4(W_i^{(0)}, ... W_i^{(4)}) = \frac{1}{2}\left(2W_i^{(4)} - W_i^{(3)} + W_i^{(2)} - W_i^{(1)} - W_i^{(0)}\right)$$

$$\mathscr{D}_5(W_i^{(0)}, ... W_i^{(4)}) = \frac{1}{2}\left(W_i^{(3)} - W_i^{(2)} + W_i^{(1)} - W_i^{(0)}\right)$$

$$\mathscr{D}_6(W_i^{(0)}, ... W_i^{(4)}) = \frac{1}{2}\left(W_i^{(2)} - W_i^{(0)}\right)$$

$$\mathcal{D}_7(\boldsymbol{W}_i^{(0)}, ... \boldsymbol{W}_i^{(4)}) = \boldsymbol{W}_i^{(2)} - 2\,\boldsymbol{W}_i^{(1)} + \boldsymbol{W}_i^{(0)}$$

$$\mathcal{D}_8(\boldsymbol{W}_i^{(0)}, ... \boldsymbol{W}_i^{(4)}) = \frac{1}{2}\left(\boldsymbol{W}_i^{(4)} - 2\,\boldsymbol{W}_i^{(2)} + \boldsymbol{W}_i^{(0)}\right)$$

Feature Selection

For a given dynamics feature vector $\boldsymbol{W}_i(k) = (w_{i,1}(k), \dots w_{i,32}(k)) \in \mathfrak{R}^{32}$, the SBFS algorithm (refer to Chapter 4) is used to select the optimal feature subset that leads to best classification results.

Feature selection is carried out for the TSI based on the performance evaluation of an SVM classifier with a linear kernel on determining viability of all-live and all-dead cultures. Assume that from the all-live culture K samples are taken and imaged as TSI. Dynamics operator $\mathcal{D}_k$ is used to extract the dynamics information using $\boldsymbol{W}_i(\mathrm{k}) = \mathcal{D}_k(\boldsymbol{W}_i^{(0)}, ... \boldsymbol{W}_i^{(T-1)})$ for all the inputs in the training and test sets, and the viability values determined by the SVM classifier are denoted as $l_j^{S,\mathcal{D}_k}$ ($j = 1, 2, \dots K$), S being a feature subset. Analogously, assume that from the all-dead culture also K samples are taken, and the viability values determined by the SVM classifier are denoted as $d_j^{S,\mathcal{D}_k}$ ($j = 1, 2, \dots K$). Assume the real viability of the all-live culture is l_0, and that of the all-dead culture is d_0, the criterion function C with regards to S and $\mathcal{D}_k$ can be constructed in the following form:

$$C(S, \mathcal{D}_k) = -\frac{1}{N}\sum_{j=1}^{N}(l_0 - l_j^{S,\mathcal{D}_k})^2 - \frac{1}{N}\sum_{j=1}^{N}(d_0 - d_j^{S,\mathcal{D}_k})^2$$

It is clear that the higher the criterion function value, the better is the classifier's performance, and consequently, the better is the feature subset S.

Results

The dataset, which we use for evaluating the system based on TSI, dynamics operators and wavelet feature selection, is the same as that used in Chapter 4. The difference is that in Chapter 4 only the micrographs at the FTP (First Time Point) of the TSI are investigated.

TSI of 5 time points have been captured for 10 samples from the all-live culture. The temporal interval between any two of the images in one TSI is one second. In the cell detection process, 466 live cells are recognized in the 10 FTP images. Analogously, 491 dead cells are detected from 10 FTP images of the

all-dead culture. Subsequently, cell tracking is performed in all of the TSI. The parameters used in cell tracking are shown in Table 6-3.

Table 6-3: Suitable parameters of the cell tracking procedure.

Size of each image	512x512
Size of cell patch	25 x 25
Size of search window	25 x 25
1st Gabor filter (Gf_1)	$\rho = 0$, $\sigma_s = 2.30$
Gf_2	$\rho = 1.30$, $\sigma_s = 2.30$, $\theta = 0$
Gf_3	$\rho = 1.30$, $\sigma_s = 2.30$, $\theta = \pi/4$
Gf_4	$\rho = 1.30$, $\sigma_s = 2.30$, $\theta = \pi/2$
Gf_5	$\rho = 1.30$, $\sigma_s = 2.30$, $\theta = 3\pi/4$

Five Gabor features are computed for each single cell on the FTP image of the TSI, and the corresponding positions of the same cell are found on the other images of the TSI. Cell tracking results of three cells are shown as examples in Fig. 6-5. The effect of cell position correction is significant, as it can be seen that in the cases with cell tracking, the cell centers are arranged to coincide with those of the cell patches.

After the positions of each cell in the TSI are found, the feature selection process is carried out. 232 live cell TSI and 247 dead cell TSI are used to generate the training set; while other 234 live cell TSI and 244 dead cell TSI are used to generate the test set. With these TSI's, dynamics feature vectors are computed according to Eq. (2) for the SBFS feature selection process. The results of the feature selection based on different orders of dynamics operators ($\mathscr{D}_1 \sim \mathscr{D}_8$) are shown in Fig. 6-6.

The common tendency of the curves can be observed in this figure. When the number of features is large, namely, the discarded features are few, performance of classifying live and dead cells is not good because of redundant and cross-talk information among these features. When more and more features are discarded, the performance is improved, and the best performance is achieved commonly with a feature number in the range of 10 and 24. However, if too many features are removed from the feature set, the performance decreases dramatically, especially when the number of features is below 6 due to the loss of essential discriminative information.

It is clear that the best two results that achieve the highest criterion values are derived from $\mathscr{D}_1$ and $\mathscr{D}_6$. Referring to their definitions, they both denote the 1st derivative of the wavelet features of the cells. The difference is that $\mathscr{D}_1$ computes the difference between the wavelet features at the 4th ($W_i^{(4)}$) and 0th ($W_i^{(0)}$) time

points, which span a larger time interval than does $\mathscr{D}_6$, which computes the difference between $W_i^{(2)}$ and $W_i^{(0)}$. In comparison, the dynamics operators that use information at more time points, like $\mathscr{D}_2$, $\mathscr{D}_4$ (using $W_i^{(0)} \sim W_i^{(4)}$) and $\mathscr{D}_5$ (using $W_i^{(0)} \sim W_i^{(3)}$), do not lead to better results.
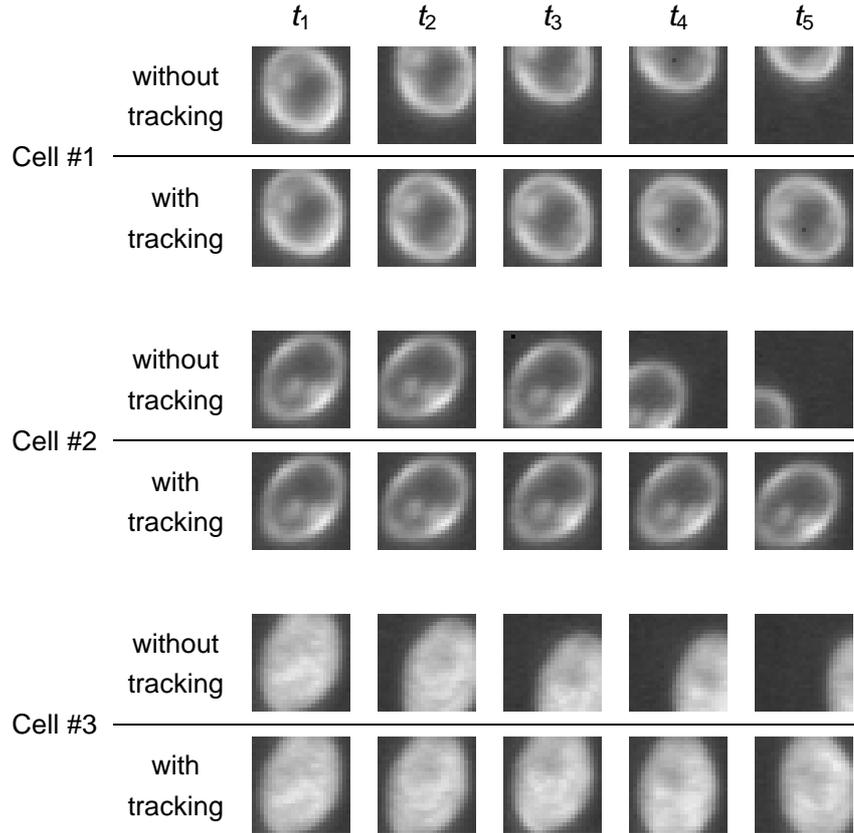
Fig. 6-5: Comparisons of time series of images with and without cell tracing.

It makes also little sense to use dynamics operators that are based on the 2nd derivative of the wavelet features, no matter whether they use features at all time points ($\mathscr{D}_3$ using $W_i^{(0)} \sim W_i^{(4)}$) or only at some ($\mathscr{D}_7$ using $W_i^{(0)}$, $W_i^{(1)}$ and $W_i^{(2)}$; $\mathscr{D}_8$ using $W_i^{(0)}$, $W_i^{(2)}$ and $W_i^{(4)}$). All these dynamics operators do not lead to better results than those achieved by $\mathscr{D}_1$ and $\mathscr{D}_6$. This may mean that the 2nd derivative of the wavelet feature does not significantly differ between live and dead cells, at least not so significantly as the 1st derivative of the wavelet feature changes for live and dead cells.
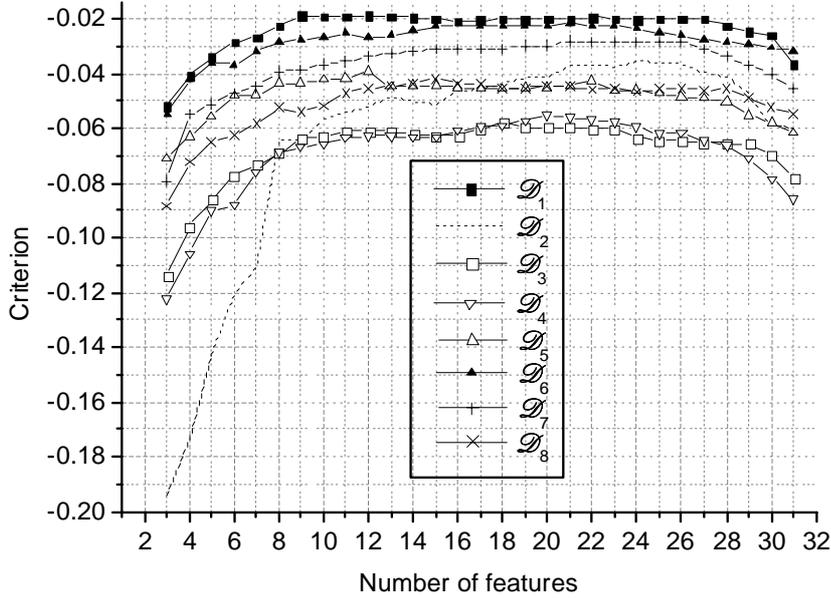
Fig. 6-6: Results of feature selection using different dynamics operators.

If only the two curves of $\mathscr{D}_1$ and $\mathscr{D}_6$ are investigated (see Fig. 6-7), it can be discovered that $\mathscr{D}_1$ is superior to $\mathscr{D}_6$ for all number of features. The global optimum with $\mathscr{D}_1$ is found when the number of features is 12, the criterion value being -0.01854, while in this case the criterion value is -0.02671 with $\mathscr{D}_6$. The global optimum with $\mathscr{D}_6$ is found when the number of features is 21, the criterion value being -0.02224. It is proposed to use the optimal subsets that contain 16 features, because only when the number of features is between 16 and 21, the classification results with $\mathscr{D}_1$ and $\mathscr{D}_6$ are similar to each other, which means that more stable results can be achieved without having to choose which of $W_i^{(4)}$ and $W_i^{(2)}$ should be used to compute the 1st derivative of the wavelet feature changes.
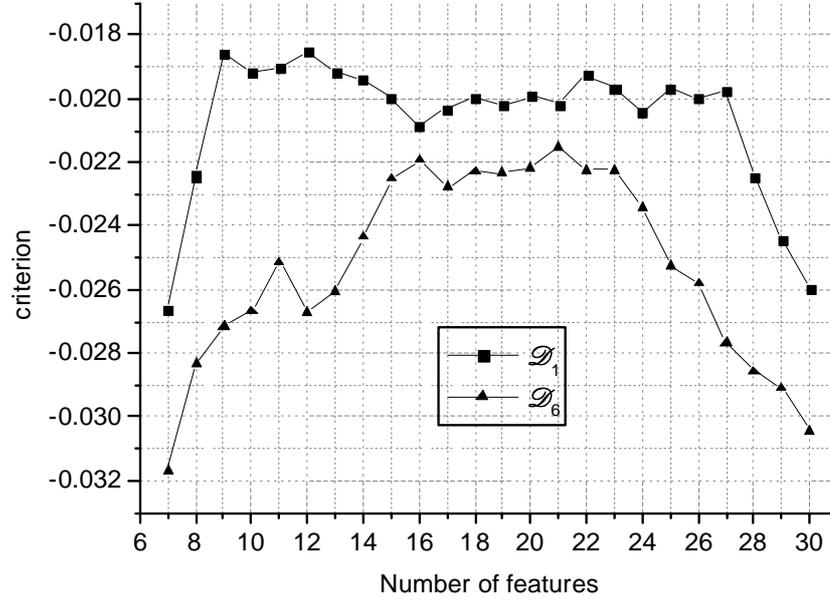
Fig. 6-7: Results of feature selection using dynamics operators $\mathscr{D}_1$ and $\mathscr{D}_6$.

If $\mathscr{D}_1$ is used to extract the dynamics information of the wavelet feature changes, the 16 indices of the selected features are (0, 2, 4, 5, 6, 8, 10, 16, 17, 18, 19, 20, 22, 24, 28, 29). Referring to the layout of the wavelet features and subimages illustrated in Chapter 4, these selected features belong to wavelet subimages (0,0), (0,1), (0,2), (1,0), (1,1), (1,2), (2,0), (2,2), (3,0), (3,1) and (3,3). The meaning of these subimages can be found in the illustration of wavelet packet decomposition in Chapter 4.

In order to evaluate the classification performance of the system for cell populations with determined viability between 0% and 100%, mixed cultures are prepared as test sets. Mixed cultures (refer to Chapter 3) are obtained by mixing all-live and all-dead cultures at a series of ratios (1:4, 2:3, 3:2, 4:1, 17:3, and 9:1). For each mixture, the viability is measured by taking the average of 5 manual counts based on FUN 1 stain (refer to Chapter 3). As the cell density of the all-live cultures is a little bit different from that of the all-dead cultures, the outcome viability is slightly deviated from the nominal value as to mixing ratio. For instance, the nominal viability of the 2:3 (all-live to all-dead) mixture is 0.4, while the actual value is about 0.37. The viability determined by FUN 1 stain is regarded as the gold standard and compared with that by the pattern classification system, in

which 5 samples of each mixture are imaged at 5 different time points, and the TSI are processed in the aforementioned approach, including cell tracking and dynamics feature computing. The selected subset of 16 features is used to represent the dynamic properties of the cells and fed to the classifier. Thereafter, the 5 viability values given by the classifier are averaged as the result of the pattern classification system. The correlation between the reference method and the pattern classification system is shown in Fig. 6-8. A correlation factor of $R^2 = 0.992$ is observed, which represents an excellent classification performance.
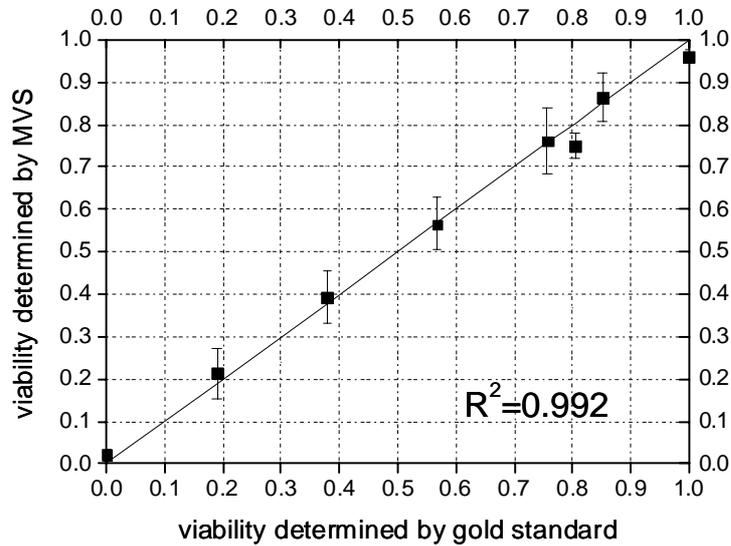
Fig. 6-8: Correlation between results obtained by gold standard and those by the MVS.

Abbreviations

FTP: First Time Point
MVS: Machine Vision System
SBFS: Sequential Backwards Floating Selection
TSI: Time Series Images

List of Symbols

$g_a(\cdot)$:   Gabor filter function in space domain
$G_a(\cdot)$:   Gabor filter function in frequency domain

σ_s: standard deviation of the Gaussian envelope

ω_1, ω_2: spatial frequency of the sinusoidal wave

ρ, θ: radial frequency and orientation of the sinusoidal wave in polar coordinates

Gf_k: Gabor filters

G_k: Gabor representations

N: width or height of an image patch

f_k: Gabor feature

T: number of the images in the time series

I_t: image at time point t

p: a pixel in an image

$\boldsymbol{F}_i$: Gabor feature vector

M: width or height of a search window

d: Euclid distance of two vectors

$\boldsymbol{W}_i^{(t)}$: the wavelet feature vector of cell i at time point t

$\mathscr{D}_k$: the k-th dynamics operator

$\boldsymbol{W}_i(k)$: the dynamics wavelet feature vector of cell i derived from $\mathscr{D}_k$

K: the number of samples taken from all-live or all-dead culture

S: a feature subset

$l_j^{S,\mathscr{D}_k}$: viability of the all-live culture samples determined by the classifier based on the k-th dynamics operator

$d_j^{S,\mathscr{D}_k}$: viability of the all-dead culture samples determined by the classifier based on the k-th dynamics operator

l_0: real viability of the all-live culture

d_0: real viability of the all-dead culture

$C\,(\cdot)$: criterion function in feature selection
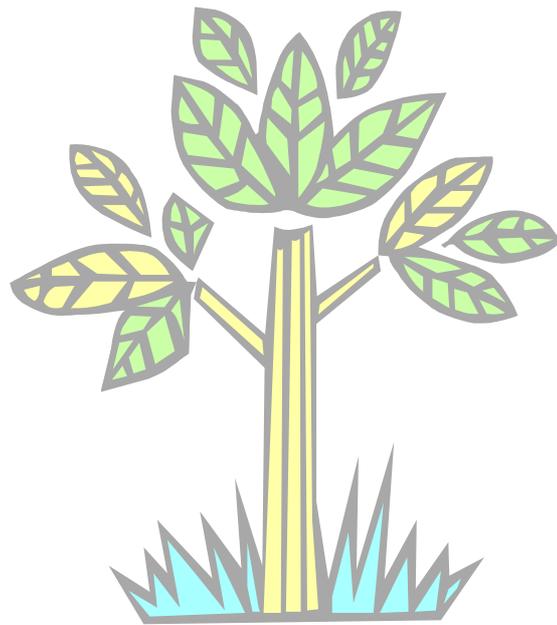
Chapter 7

Conclusions and Outlook

A new-type of *in situ* probe has been developed to acquire dark field images of cells in bioreactors. It has been derived from an *in situ* bright field microscope that is able to measure cell density in bioreactors during fermentation processes. The illumination part of the probe has been replaced with a dark field device, for which an aspheric condenser has been designed and built, so that high contrast dark field images can be obtained by means of an integrated CCD camera. A relay lens is implemented to improve the sharpness of the images.

In contrast to DIC (Differential Interference Contrast) and phase contrast imaging technique, dark field microscopy provides not only high contrast images, on which sub-cellular structures can be well recognized, but also a relatively low technical complexity of the optical system. According to the results obtained, the dark field micrographs captured by the *in situ* probe are suitable for both cell detection and viability classification. Therefore, it has an advantage over the original bright field microscope, with which only cell density can be determined.

To test the probe, brewer's yeast *Saccharomyces cerevisiae* is chosen as the target microorganism. Images of the yeast cells in a bioreactor are captured, processed, and analysed automatically by means of mechatronics, image processing, and machine learning. Two support vector machine based classifiers are used for discriminating cells from background, and live cells from dead cells, respectively. The results of the *in situ* experiments showed strong correlation between results obtained by the probe and those by widely accepted standard methods. Some other strategies of distinguishing live and dead cells that are based on feature generation and selection and time series images have also been developed for the system. This *in situ* probe has been proved to be an adequate device for on-line monitoring of both cell density and viability with high accuracy and stability.

Since the software framework of this system is based on two separate stages of supervised classification, the classifiers need to be trained before they can be used for evaluating micrographs. Compared to other applications based on pattern recognition technique, the task of collecting training sets in this work is less complicated. In total, only two types of cell cultures are required to be prepared: all-live and all-dead cultures. The classifiers are trained to learn from the images of these cultures, and then they can be implemented to distinguish cells from background, or, live from dead cells, respectively. Though the all-dead cultures used to train the classifiers are obtained by heating the cells, the classifiers are suitable for evaluating mixtures of live cells and starved cells as well.

It has been revealed that feature selection using SBFS (Sequential Backwards Floating Selection) can improve the performance of the system. The reason lies in the fact that the raw data may contain redundant or misleading information derived from non-uniform system configurations. For instance, unstable illumination intensity, non-uniform focal position, etc. In these cases, feature selection plays a role of excluding features containing such disturbing information, and leads to better results.

Wavelet features are found to be suitable to describe the distinguishing properties of the live and dead cells, as wavelet feature selection is successful in viability classification. According to the analysis, live cells exhibit morphologically more details and are intracellularly more organized than dead ones, which display more homogeneous and diffuse grey values throughout the cells.

Feature selection also reduces the dimensionality of the datasets. That enables the implementation of SVM (Support Vector Machine) classifiers with a linear kernel, which are supposed not to be used in high-dimensionality cases. As a result, it is not required to select the width parameter of the SVM classifiers with Gaussian kernels.

Comparing the results from TSI (Time Series Images) and those from STI (Single Time point Images), no matter using PCA (Principal Component Analysis) features or wavelet features plus dynamics operators, it has been shown that there is no advantage in using TSI over STI. However, the implementation of TSI is far more complicated than that of STI, as the former requires not only capturing of a series of images at high frame rate, but also tracking of the cells.

Although the IDMP (*In situ* Dark field Microscopy Probe) has been successfully developed and implemented in a prototype bioreactor, there are still problems remaining to be solved in future work.

First of all, due to the fact that the dark field illumination part of the IDMP is overlong, it cannot be applied in typical small or middle size bioreactors, like 7 L

or 20 L ones, as it will interfere with the stirrer inside. Therefore, it is necessary to reduce the probe's length, such that it can fit into lab-scale bioreactors.

There is still a problem with the watertightness of the sapphire windows, as they are fixed with some glue that is not sufficiently heat-resistant. Thus, during the operation of the LED (Laser Emitting Diode), which produces much heat, the glue will be impaired. As a result, the sapphire windows have to be changed and glued again after a couple of operations, or otherwise water could come into the chamber, damaging the LED. One possible solution to this problem might be a more heat-resistant glue instead of the current one.

Another problem is with the intensity of the illumination. Currently available LED's do not provide sufficient luminance, thus, capturing images cannot be performed with very short shutter time. The consequence is that the probe cannot image mobile cells, i.e., before each snapshot, it has to close the sampling region and wait for sufficient time until the cells are immobilized. This greatly reduces the rate at which image could be taken.

Up to now, the experiments have been carried out only with the yeast *Saccharomyces cerevisiae*. However, it would be interesting to implement the probe to monitor the growth of other yeasts such as *Saccharomyces pombe*, of animal cells, and of cells of other types of organisms. The challenges of dealing with various cells that have different morphological forms and properties are still to be met.

The framework of the software for this probe is composed of two separate steps: cell detection and viability classification. In future work, the images can also be processed in only one step by applying a three-class classifier to distinguish live, dead cells and background. In this case, multi-class SVM, boosted tree classifiers, or other multi-class classifiers can be applied.

References

Anton F, Burzlaff A, Kasper C, Brückerhoff T and Scheper T. 2007. Preliminary Studz towards the Use of In-situ Microscopy for the Online Analysis of Microcarrier Cultivations. Eng. Life. Sci. 1: 91-96.

Bishop CM. 1995. Neural Networks for Pattern Recognition. Oxford University Press, Oxford, UK. pp.310

Bittner C, Wehnert G, Scheper T. 1998. *In Situ* Microscopy for On-line Determination of Biomass. Biotechn. Bioeng. 60: 24-35

Castro-Concha LA, Escobedo RM, Miranda-Ham M. 2006. Measurement of Cell Viability in In Vitro Cultures. Plant Cell Culture Protocols (2nd ed.). Humana Press. Totowa, New Jersey, the U. S.

Charnbolle A, De Vore RA, Lee NY, Lucier BJ. 1998. Nonlinear wavelet image processing: variational problems,compression, and noise removal through wavelet shrinkage. IEEE T. Image Process. 7: 319-335.

Chen CH, Pau LF and Wang PSP (eds.). 1998. The Handbook of Pattern Recognition and Computer Vision (2nd ed.). Chapter 2.1: Texture Analysis. pp. 207-248. World Scientific Publishing Co., Singapore.

Chen W, Giger ML, Lan L and Bick U. 2004. Med. Phys. 31: 1076-1082.

Claes, JE and Van Impe JF. 1999. On-line estimation of the specific growth rate based on viable biomass measurements: experimental validation. Bioproc. Biosyst. Eng. 21: 389-395.

Cook JA and Mitchell JB. 1989. Viability measurements in Mammalian Cell Systems. Anal. Biochem. 179: 1-7.

Cortes C and Vapnik V. 1995. Support-vector networks. Mach. Learn. 20: 273-297.

Dash M and Liu H. 1997. Feature Selection for Classification. Intell. Data Anal. 1: 131-156.

Daugman JG. 1988. Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression. IEEE Trans. Acoust. Speech Signal Process. 36: 1169-1179.

Dempster AP, Laird NM and Rubin DB. 1977. Maximum likelihood from incomplete data via the EM algorithm. J. Royal Stat. Soc., B. 39: 1-38

Garcia C, Zikos G, Tziritas G. 2000. Wavelet packet analysis for face recognition. Image Vision Comput. 18: 289-297.

Gerig G, Welti D, Guttmann CRG, Colchester ACF and Székely G. 2000. Exploring the discrimination power of the time domain for segmentation and characterization of active lesions in serial MR data. Med. Image Analysis. 4: 31-42.

Green DR and Kroemer G. 2004. The Pathophysiology of Mitochondrial Cell Death. Science. 305: 626-629.

Guo G, Li SZ and Chan KL. 2001. Support vector machines for face recognition. Image Vision Comput. 19: 631-638.

Guyon I and Elisseeff A. 2003. An Introduction to Variable and Feature Selection. J. Mach. Learn. Reasearch. 3: 1157-1182.

Hamahashi S, Onami S and Kitano H. 2005. Detection of nuclei in 4D Nomarski DIC microscope images of early Caenorhabditis elegans embryos using local image entropy and object tracking. BMC Bioinformatics. 6: 125

Harris CM and Kell DB. 1985. The estimation of microbial biomass. Biosensors. 1: 17-84.

Haussecker H and Geiler P (eds.) 1999. Handbook of Computer Vision and Applications. Chapter 12: Texture Analysis. pp. 275-308. Academic Press Inc., USA.

Haykin, SS. 1999. Neural networks: a comprehensive foundation (2nd ed.). pp. 318-347. Prentice Hall. Upper Saddle River, New Jersey, the U. S.

Heath JP. 2005. Dictionary of Microscopy. Chichester, England: John Wiley & Sons. 357p.

Heggart H, Margaritis A, Stewart RJ, Pilkington M, Sobezak J, Russell I. 2000. Measurement of brewing yeast viability and vitality: a review of methods. Tech. q. - Master Brew. Assoc. Am. 37: 409-430.

Huang K and Murphy RF. 2004. Boosting accuracy of automated classification of fluorescence microscope images for location proteomics. BMC Bioinformatics. 5:78

Huang L and Shimizu A. 2005. A multi-expert approach for robust face detection. Pattern. Recognition. 39: 1695-1703

Huls PG, Nanninga N, Van Spronsen E. 1992. A computer-aided measuring system for the characterization of yeast populations combining 2D-image analysis, electronic particle counter, and flow cytometry. Biotechnol. Bioeng. 39: 343-350.

Jain A and Zongker D. 1997. Feature Selection: Evaluation, Application, and Small Sample Performance. IEEE T. Pattern Anal. Mach. Intell. 19: 153-158

Joeris K, Frerichs JG, Konstantinov K, Scheper T. 2002. In-situ microscopy: Online process monitoring of mammalian cell cultures. Cytotechnology. 38: 129-134

Kronlof J. 1991. Evaluation of a capacitance probe for determination of viable yeast biomass. Proc. 23rd European Brewing Congress, Lisbon, European Brewing Convention, 233-240.

Kudo M, Sklansky J. 2000. Comparison of algorithms that select features for pattern classifiers. 33: 25-41.

Long X, Cleveland WL, Yao YL. 2006. Automatic detection of unstained viable cells in bright field images using a support vector machine with an improved training procedure. Comput. Biol. Med. 36: 339-362.

Madeo F, Fröhlich E, Ligr M, Grey M, Sigrist SJ, Wolf DH, and Fröhlich KU. 1999. Oxygen Stress: A Regulator of Apoptosis in Yeast. J. Cell Biol. 145: 757-767

Majno G and Joris I. Apoptosis, oncosis, and necrosis: an overview of cell death. 1995. Amer. J. Pathol. 146: 3-15

Meier DS and Guttmann CRG. 2003. Time-series analysis of MRI intensity patterns in multiple sclerosis. NeuroImage. 20: 1193-1209.

Nattkemper TW. 2001. A Neural Network-Based System for High Throughput Fluorescence Micrograph Evaluation. Doctoral Thesis. Technical Faculty, Bielefeld University, Germany.

Nattkemper TW, Ritter HJ, Schubert W. 2001. A Neural Classifier Enabling High-Throughput Topological Analysis of Lymphocytes in Tissue Sections. IEEE T. Inf. Technol. B. 5: 138-149.

Nattkemper TW, Twellmann T, Ritter H, Schubert W. 2003. Human vs. Machine: evaluation of fluorescence micrographs. Comput. Biol. Med. 33: 31-43.

Papageorgiou CP, Oren M and Poggio T. 1998. A general framework for object detection. Proceedings of the Sixth International Conference on Computer Vision. pp. 555-562.

Pichler O, Teuner A, Hosticka BJ. 1996. A Comparison of texture feature extraction using adaptive gabor filtering, pyramidal and tree structured wavelet transforms. Pattern Recogn. 29: 733-742.

Plas DR and Thompson CB. 2002. Trends Endocrinol. Metab. Cell metabolism in the regulation of programmed cell death. 13: 75-78

Pons MN, Vivier H, Remy JF, Dodds JA. 1993. Morphological characterization of yeast by image analysis. Biotechnol. Bioeng. 42: 1352-1359.

Pudil P, Novovičová J, Kittler J. 1994. Floating search methods in feature selection. Pattern Recogn. Lett. 15: 1119-1125.

Ramsay JO, Silverman BW. 2005. Functional Data Analysis. 2nd ed.. Springer, Berlin, Germany.

Rätsch M, Romdhani S, Vetter T. 2004. Efficient Face Detection by a Cascaded Support Vector Machine Using Haar-Like Features. Lect. Notes Comput.

Sci. 3175: 62-70.

Ruldolph G, Brückerhoff T, Bluma A, Korb G and Scheper T. 2007. Optische Inline-Messverfahren zur Zellzahl- und Zellgrößenbestimmung in der Bioprozesstechnik. Chemie Ingenieur Technik. 79: 42-51.

Scheper T. 1991. Bioanalytik. Vieweg-Verlag, Braunschweig, Germany.

Soltanian-Zadeh H, Rafiee-Rad F, Pourabdollah-Nejad DS. 2004. Comparison of multiwavelet, wavelet, Haralick and shape features for microcalcification classification in mammograms. Pattern Recogn. 37: 1973-1986.

Somol P, Pudil P, Novovičová J, and Paclík P. 1999. Adaptive floating search methods in feature selection. Pattern Recog. Lett. 20: 1157-1163.

Sonnleitner B, Locher G, Fiechter A. 1992. Biomass Determination. J. Biotechnol. 25: 5-22.

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. Biotechn. Bioeng. 47: 106-116.

Taylor DL, Woo ES, Giuliano KA. 2001. Real-time molecular and cellular analysis: the new frontier of drug discovery. 12: 75-81.

Thomas CR and Paul GC. 1996. Applications of image analysis in cell biology. Curr. Opin. Biotechnol. 7: 35-45.

Twellmann T, Lichte O and Nattkemper TW. 2005. An Adaptive Tissue Characterisation Network for Model-Free Visualisation of Dynamic Contrast-Enhanced Magnetic Resonance Image Data. IEEE Tran. Med. Image. 24: 1256-1266.

Van Imple J, Iserentant D, Vanrolleghem P. 1998. Advanced Instrumentation, Data Interpretation and Control of Biotechnological Processes, pp. 41-66. Kluwer Academic Publishers, Dordrecht, The Netherlands.

Vecht-Lifshitz SE and Ison AP. 1992. Biotechnological applications of image analysis: present and future prospects. J. Biotechnol. 23: 1-18.

Viola P and Jones M. 2001. Robust Real-time Object Detection. Intern. J. Comput. Vision. 57: 137-154.

Viola P and Jones M. 2001. Rapid Object Detection using a Boosted Cascade of Simple Features. Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01), Vol. 1, pp. 511-519.

Watson RWJ, Austin GD, D'Amore T. 1994. Studies of on-line viable yeast biomass with a capacitance biomass monitor. Biotechn. Bioeng. 43: 337-341.

References

Wei N, Flaschel E, Saalbach A, Twellmann T, Nattkemper TW. 2005. Reagent-free automatic cell viability determination using neural network based machine vision and dark-field microscopy in Saccharomyces cerevisiae. Proceeding of the 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 1-4 Sept 2005, Shanghai, China.

Wei N, You J, Friehs K, Flaschel E, Nattkemper TW. 2007. In situ dark field microscopy for on-line monitoring of yeast cultures. Biotechnol. Lett. 29: 373-378.

Wei N, You J, Friehs K, Flaschel E, Nattkemper TW. 2007. An in situ probe for on-line monitoring of cell density and viability on the basis of dark field microscopy in conjunction with image processing and supervised machine learning. Biotechnol. Bioeng. accepted.

Xiang H, Liu J, Li M, Huo R. 2005. Online rapid monitoring system for cell viability and function in bioartificial liver support system. Biomedical engineering und clinical medicine (in Chinese). 2005, vol. 1.