

# Effective automatic recognition of cultured cells in bright field images using fisher's linear discriminant preprocessing

Xi Long<sup>a</sup>, W. Louis Cleveland<sup>b,\*</sup>, Y. Lawrence Yao<sup>a</sup>

<sup>a</sup>Mechanical Engineering Department, Columbia University, 220 Mudd Bldg, MC4703, New York, NY 10027, USA

<sup>b</sup>The Department of Medicine at St Luke's Roosevelt Hospital Center and Columbia University, New York, NY 10019, USA

Received 19 October 2004; received in revised form 21 July 2005; accepted 26 July 2005

## Abstract

Automatic cell recognition in bright field microscopy is an inherently difficult task due to the immense variability of cell appearance. Yet, it is essential for a high-throughput robotic system. In this paper, we employed a pixel patch decomposition method to detect cultured cells in bright field images. To increase the classification accuracy, we proposed a novel Fisher's Linear Discriminant (FLD) preprocessing approach. This technique was applied to various experimental scenarios utilizing different imaging environments and the results were compared with those for the traditional Principal Component Analysis (PCA) preprocessing. Our FLD preprocessing was shown to be more effective than PCA primarily owing to its ability to maximize the ratio of between-class scatter to within-class scatter. The optimized algorithm has sufficient accuracy and speed for practical use in robotic systems capable of automatic micromanipulation of single cells.

© 2005 Elsevier Ltd All rights reserved.

*Keywords:* Cell recognition; Fisher's linear discriminant; Principal component analysis; Neural networks

## 1. Introduction

Recent progress in the development of methods for molecular genetic analysis (e.g. RT-PCR (reverse transcription-polymerase chain reaction), microarrays) has brought sensitivities to the level where single cells can be analyzed [23]. However, to carry out assays on significant numbers of cells, high throughput robotic systems are needed. These systems require identification of individual cells (often in cell cultures) for micromanipulation and subsequent molecular analysis. Identification of cells in cultures is a difficult task that is normally done by an experienced human observer using an optical microscope. However, the use of human observers represents a severe impediment to the development of high throughput robotic systems. Therefore, there is a major need for algorithms that permit automatic recognition of cells in images obtained with computer-controlled microscopes equipped with electronic cameras.

One approach to automatic cell recognition is to use fluorescent probes that have a chemical specificity for cell organelles. For example, DNA intercalators are routinely used to stain nuclear DNA for cell identification [9–11]. However, this approach can consume one or more fluorescence channels just for the purpose of cell identification. It is highly desirable to identify cells with a method that uses transmitted light illumination, thereby permitting all of the fluorescence channels to be used for other purposes. Of special importance is the use of fluorescent probes to provide additional cellular and subcellular information that can greatly enhance the value of the information obtained by molecular genetic analysis.

Given the variability of cell size and morphology, the presence of 'trash', as well as variations in microscope parameters, such as focus and illumination, it is a challenging task to develop a robust algorithm for automatic cell recognition. A further challenge is to develop a flexible algorithm that is easy to use by biologists, who may have minimal image processing and programming skills.

Early work on automatic cell recognition has been dominated by attempts to identify image information that is relatively invariant [3–6] and which, therefore, facilitates algorithms based on heuristic rules. These approaches tend

\* Corresponding author. Tel.: +1 212 523 7302; fax: +1 212 523 5377.  
E-mail address: [wlc1@columbia.edu](mailto:wlc1@columbia.edu) (W.L. Cleveland).

to be very sensitive to changes in cell type. For example, cell lines with different morphologies require independent optimizations that involve consideration of detailed features. In addition, these approaches appear to have poor tolerance to variations in focus and illumination conditions.

An alternative to the classical image analysis approach described above is the use of algorithms based on machine learning techniques, of which several are currently available [1]. Algorithms of this type have the compelling advantage that end-user programming is largely eliminated. Variations in cell type and other conditions are simply accommodated by training.

Among machine learning techniques, Artificial Neural Networks (ANN) have been extensively explored and are considered advantageous because they are able to capture complex, even nonlinear, relationships in high dimensional feature spaces that are not easily handled by algorithms based on heuristic rules [2].

Successful ANN-based cell recognition algorithms that use pixel values directly from primary gray-scale images (bright field) have been described [7,8]. However, these networks can be very complex due to the large numbers of inputs. This leads to slow performance and difficulty in training. In response to this problem, powerful statistical data processing techniques have been used in preprocessing to generate more abstract representations that are better suited for subsequent neural network analysis. For example, Nattkemper et al. applied Principle Component Analysis (PCA) in evaluation of fluorescence microscopy images [9–11]. Kämpfe et al. combined Independent Component Analysis (ICA) and Self-Organization Maps (SOM) for classification of cells with fluorescent nuclei [12]. These applications are successful in dealing with images in which nuclei, stained with a fluorescent probe, have a characteristic color and show considerable uniformity with respect to size and shape. In images obtained with transmitted light illumination, there is a greater variation of whole-cell size and shape. In addition, cellular debris and other forms of ‘trash’ can be similar in appearance to intact cells. Therefore, bright field images of unstained cells represent a more difficult challenge, which requires a novel approach for robust cell recognition.

In our study, we have developed a cell recognition approach based on feed-forward neural networks in combination with improved preprocessing. Initially, we explored PCA preprocessing as described in [9–11] and found it unsatisfactory. Subsequently, we explored Fisher’s Linear Discriminant (FLD) preprocessing. Although FLD has been used in computer vision strategies by others [17–19], it cannot be used for preprocessing in cell recognition tasks in the manner that PCA has been used and that changing from PCA to FLD is not straightforward. We have developed a novel strategy to make possible the use of FLD and found that FLD in combination with feed-forward neural networks gave performance sufficient for use

in a practical robotic system for single cell level molecular analysis. To our knowledge, FLD has never been used in combination with neural networks for cell recognition. In what follows, we present a quantitative and systematic evaluation of PCA and FLD as preprocessing steps for optimized feed-forward neural networks. This evaluation has used images of cultured cells as well as images of polymeric microspheres as models of cells.

## 2. Methods

We followed the cell recognition strategy described by Nattkemper et al. [9–11]. The process involves two stages: preprocessing and classification. The major task of preprocessing is to derive a representation of cells which makes subsequent classification computationally effective and insensitive to environmental changes by providing the classifier only with information essential for recognition. In the classification stage, a neural network is trained to recognize if an image patch contains a centered cell body. This is done with image patches represented by feature vectors derived in preprocessing.

As illustrated in Fig. 1, cells in a digitized microscope image are detected in the following steps: First, for each pixel  $p$  in the microscopic image, a sub-image which consists of its  $m \times m$  neighborhood is extracted and mapped to a so-called *confidence value*  $C[p] \in [-1, 1]$  by the classifier, where the size  $m$  can be adjusted to accommodate

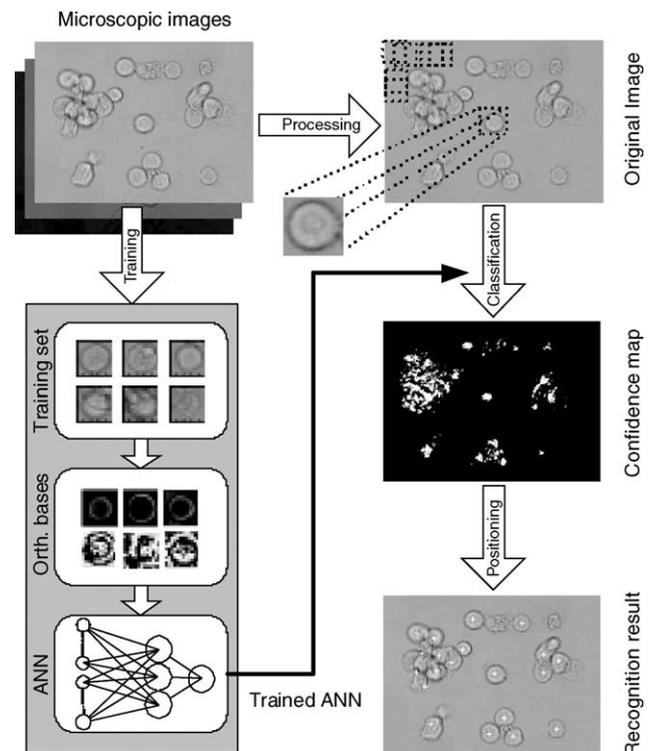


Fig. 1. Illustration of the recognition process.

cell size. After all the pixels are processed, a new image, called a *confidence map* is created. Pixels in the confidence map are the confidence values of their corresponding sub-images in the original microscope image and form ‘mountains’ with large peaks that represent cell positions. The cell positions can then be easily found by identifying local maxima in mountains. To increase speed (this procedure is optional), patches clearly indicating background, i.e. patches with average pixel intensities below a user-defined value (often determined by a simple histogram), are not analyzed.

2.1. Learning set

Fig. 2(a) shows a learning set used for deriving the representation of cells and training of the ANN. This set was created with the aid of an interactive program that displays the microscope image and allows a human expert to select the locations of patch centers with a mouse cursor after manual evaluation of the microscope image. A similar set was used for microspheres (not shown). The learning set  $\Omega$  is composed of two subsets ( $\Omega = \Omega^{\text{pos}} + \Omega^{\text{neg}}$ ).  $\Omega^{\text{pos}}$  contains patches of centered cells and is labeled ‘Cell’. All images in  $\Omega^{\text{pos}}$  belong to a single class.  $\Omega^{\text{neg}}$  is labeled ‘Non-cell’.

2.2. Preprocessing

Images in the learning set are represented by viewing each image as a point in a high dimensional image space. Considering the complexity and vastness of image space, both PCA and FLD are employed to reduce it to a subspace with much lower dimensional. Our method begins with the following presuppositions: under various transformations, images of a particular class of objects occupy a relatively small but distinct region of the image space; different classes of images occupy different regions of image space.

Both PCA and FLD preprocessing techniques exploit these assumptions, but in different ways and therefore exhibit significantly different properties.

2.2.1. PCA method

Principal component analysis (PCA) is a commonly used technique for dimensionality reduction in computer vision, particularly in object recognition [9–11,13–16].

Suppose the learning set  $x$  is composed of  $n$  sample images ( $x = \{x_1, x_2, \dots, x_n\}$ ), where each image is represented by a vector  $x_i$  in an  $N$ -dimensional image space. The goal is to find a linear transformation that maps the input set to an  $M$ -dimensional subspace, where  $M < N$ . After defining  $x' = \{x'_1, x'_2, \dots, x'_n\} = \{(x_1 - \mu), (x_2 - \mu), \dots, (x_n - \mu)\}$ , where  $\mu$  is the sample mean, the new feature vector set  $y = \{y_i\}$  can be defined by the linear transformation:

$$y = W^T x', \quad y_i = W^T x'_i \quad i = 1, 2, \dots, n, \tag{1}$$

where  $W$  is an  $N \times M$  matrix whose columns form a orthonormal basis of the subspace. PCA seeks to minimize the mean square reconstruction error, or equivalently, to maximize the projection variance. The optimal transformation matrix  $W_{\text{opt}}$  in PCA can be defined as:

$$W_{\text{opt}} = \underset{W}{\operatorname{argmax}} |W^T S_T W| \tag{2}$$

where  $S_T$  is an  $N \times N$  matrix called total scatter.

$$S_T = x' x'^T = \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T \tag{3}$$

A larger eigenvalue means more variance in the data captured by the corresponding eigenvector. Therefore, by eliminating all eigenvectors except those corresponding to the highest  $M$  eigenvalues, the feature space for recognition is reduced from the original  $N$ -dimensional image space to

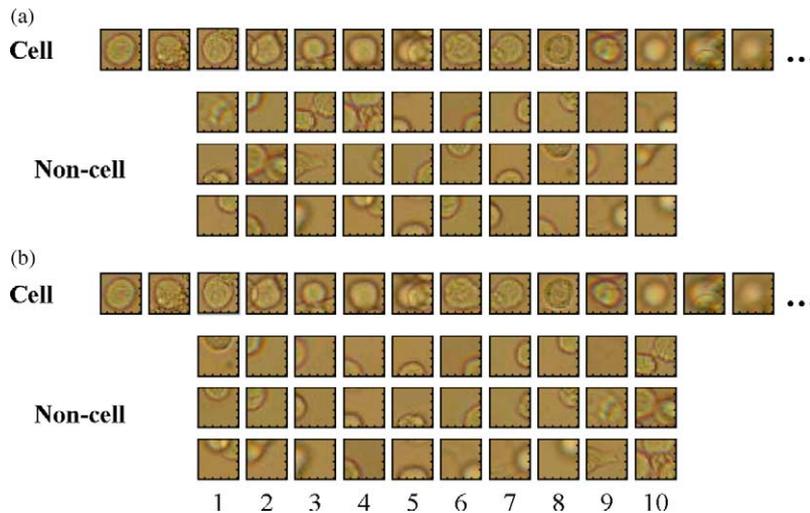


Fig. 2. Sample cell patches in the learning set. All patches are sized  $25 \times 25$  pixels. (a) Original learning set. (b) A novel strategy that makes possible the use of FLD: factoring the ‘non-cell’ class into subclasses.

the subspace spanned by the top  $M$  eigenvectors. The eigenvectors have the same dimension as the original images and are referred to as *eigenfaces* in a face recognition context [15,16] and *eigencells* in a cell recognition context [9–11].

It is worth noting that although the PCA projection is optimal for reconstruction from a low dimensional basis, it is not optimal from a classification point of view, for it only considers the total scatter of the whole sample set and makes no discrimination within the sample points (images). In PCA, the total scatter is maximized. Therefore, there is not only maximization of the between-class scatter, which is useful for classification, but there is also maximization of the within-class scatter, which should be minimized. Consequently, PCA retains or even exaggerates unwanted information such as illumination variation and out of focus blur during the projection process, which, unfortunately, may play a dominant role in many classification applications [19]. Points from individual classes in the low dimensional feature space may therefore not be well clustered, and even worse; the points from different classes could be mixed together.

### 2.2.2. FLD method

Unlike PCA, Fisher’s linear discriminant (FLD) considers not only between-class variation but also within-class variation, and optimizes the solution by maximizing the ratio of between-class scatter to within-class scatter. This can be expressed in mathematical terms as follows.

Again, assume that each image in the learning set belongs to one of  $c$  classes  $\{C_1, C_2, \dots, C_c\}$ . The between-class scatter matrix  $S_B$  and within-class scatter matrix  $S_W$  can be defined as:

$$S_B = \sum_{i=1}^c m_i(\mu_i - \mu)(\mu_i - \mu)^T \quad (4)$$

and

$$S_W = \sum_{i=1}^c \sum_{x_k \in C_i} (x_k - \mu_i)(x_k - \mu_i)^T \quad (5)$$

where  $\mu$  is the grand mean,  $\mu_i$  is the mean of class  $C_i$  and  $m_i$  denotes the number of images in class  $C_i$ .

The objective of FLD is to find the  $W_{opt}$  maximizing the ratio of the determinants of the above scatter matrices:

$$W_{opt} = \underset{W}{\operatorname{argmax}} \frac{|W^T S_B W|}{|W^T S_W W|} \quad (6)$$

$W_{opt}$  is known to be the solution of the following generalized eigenvalue problem:

$$S_B W - S_W W \Lambda = 0 \quad (7)$$

where  $\Lambda$  is a diagonal matrix whose elements are the eigenvalues. The column vectors  $w_i$  ( $i = 1, \dots, m$ ) of matrix  $W$  are eigenvectors corresponding to the eigenvalues in  $\Lambda$

and are referred to as *fisherfaces* in a face recognition context [19]. In this work, we solved the matrix  $W$  with the QZ algorithm introduced by Moler and Stewart [20]. Note that the upper bound on  $m$  is  $c - 1$ , because there are at most  $c - 1$  nonzero generalized eigenvalues. Compared to the PCA method, the representation yielded by FLD tries to reshape the scatter instead of conserving its original details. It emphasizes the discriminatory content of the image. To illustrate the benefit of FLD projection, we projected the learning set to a three dimensional subspace using PCA and FLD respectively. Results are shown in Fig. 3. For viewing clarity, only 25% of the points (randomly selected) are displayed. One can clearly see that although the point distribution range of PCA projection is greater in all three directions, (i.e. the total scatter is larger), the points from different classes are somewhat mixed together. On the other hand, the points from different classes in FLD projection are better separated and, therefore, more suitable for classification.

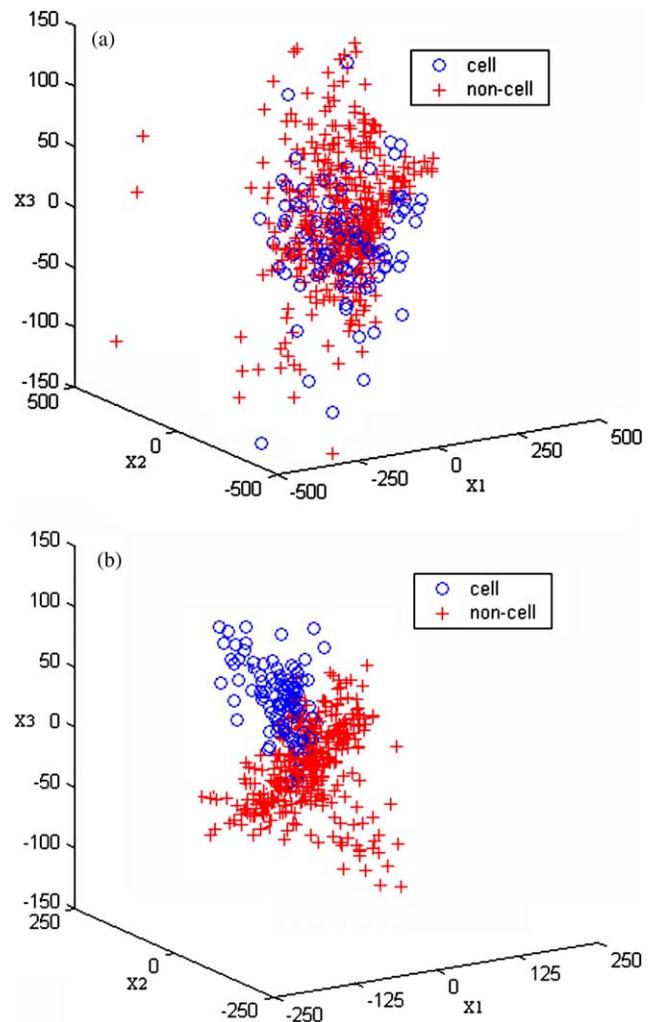


Fig. 3. Comparison of PCA and FLD by projecting the learning set to 3D subspace with PCA and FLD respectively. (a) With PCA, (b) With FLD. For viewing clarity, only 25% of the points (randomly selected) are shown.

### 2.2.3. FLD preprocessing strategy

It should be noted that when FLD is used in the preprocessing step to reduce the dimension of the initial feature space, one must reduce the dimension of the resulting subspace to no more than  $c - 1$ , where  $c$  is the number of recognition classes. In our case, where only one type of cell is present, there are only two recognition classes: ‘Cell’ and ‘Non-cell’. Therefore, if FLD preprocessing is used in the same manner as PCA, the resulting subspace can only be a one-dimensional space. A one-dimensional space is inadequate for a complex pattern recognition problem like cell recognition since features with only one dimension cannot provide enough information to the classifier.

We have developed a novel strategy to make possible the use of FLD. To our knowledge, this strategy has never been used previously. The key idea of this strategy is to treat the cell recognition as a multiclass problem in the preprocessing stage. It can be done by dividing the ‘Non-cell’ class into multiple subclasses. This is possible because the ‘Non-cell’ class contains a diverse set of cell fragments and non-cell objects. Factoring this class into  $n$  subclasses that are reasonably homogeneous makes the total number of classes equal to  $n + 1$  (‘Non-cell’ classes + ‘Cell’ class). The dimension of the subspace produced by FLD preprocessing can now be as high as  $(n + 1) - 1 = n$ . The above principle is illustrated by the example shown in Fig. 2 (b). Specifically, the ‘Non-cell’ class is manually divided into 10 subclasses. Classes 1–8 contain a specific fraction of a cell. Image patches in class 9 are almost blank (i.e. background) and class 10 includes image patches with multiple fragments of different cells. This ‘factoring’ of the ‘Non-cell’ class into subclasses makes the use of FLD a practical possibility, since the total class number is now 11 and the dimension of the feature space after FLD preprocessing can therefore be as high as 10. An additional benefit of our strategy is that it reduces variation within individual non-cell subclasses. This reduction of within-class variation is in addition to that which derives from the inherent ability of FLD to reduce within-class variation.

### 2.3. Classification

The classifier we used in this paper is a feed-forward Neural Network. It has been widely used and proven to be a powerful tool for classification tasks. In addition, there exist many hardware acceleration methods for ANN. This makes it possible to dramatically increase the processing speed, which is of great importance for the development of a high-throughput robotic system capable of automatic cell micromanipulation. The ANN uses the input–output mapping learned from a set of training samples to generalize to data not ‘seen’ before. The training of an ANN involves gradually modifying the synaptic weights according to the back-propagated output error for each sample until the desired average responses are obtained on the entire set.

The network structure was designed to obtain maximal flexibility by allowing the size of the network to be adjusted by easily changing only a few parameters. Besides the input layer, three layers were used: two hidden layers, and one output layer. The size of the neural network can be adjusted according to the size of training set and dimension of input vectors. In order to be able to establish all possible representations, a bias weight in addition to the inputs was included in each neuron. The hyperbolic tangent sigmoid function, TANSIG(x), was used as the transfer function throughout the network.

When a network is trained by examples, the most important issue is how well it generalizes to new data. It is known that the capacity for generalization greatly depends on the structure of the neural network [1]. Generally, more neurons in the hidden layers give the system more capacity to partition the data. However, if the network has too many neurons, it will learn insignificant aspects of the training set and lose its generalization ability, a phenomenon termed overfitting. Unfortunately, there is no simple rule to determine how many hidden units are required for a given task. A rule of thumb is to obtain a network with the fewest possible neurons in the hidden layer. Using the smallest possible size not only helps improve generalization, it also increases the computational speed of the system for there is roughly a linear relationship between network size and speed [21].

One commonly used method to train an ANN with good generalization ability is to start with a small initial network and gradually add new hidden units in each layer until efficient learning is achieved. This approach has many drawbacks, among which are: slow learning and difficult-to-avoid local minima. To avoid these drawbacks, we employed an alternative strategy referred to as *pruning*, which starts with a large network and excises unnecessary weights and units. The training results of our ANNs are shown in Section 4.1.

## 3. Materials and experimental conditions

Both microspheres and living cells were used as testing materials in our study. The microspheres were 2.5  $\mu\text{m}$ -diameter, dry-red Fluorescent Polymer Microspheres from Duke Scientific (Cat. No. 36-5). The cells used were K562 chronic myelogenous leukemic cells (ATCC; Cat. No. CCL-243) grown at 37.0 °C in BM + 1/2 TE1 + TE2 + 10% fetal calf serum (FCS) [22]. For microscope observation, cells and microspheres in culture medium were dispensed into polystyrene 96-well microplates, which have well bottoms that are 1 mm thick.

An Olympus Model-CK inverted microscope equipped with a 20 $\times$  planachromat objective and a SONY DSC-F717 digital camera was used to obtain digitized images.

The image processing, ANN training and classification programs were written in MATLAB code and implemented in MATLAB Version 6.5.0.180913a (R13) supplemented

with Image Processing Toolbox Version 3.2 and Neural Network Toolbox Version 4.0.2. A standard PC equipped with an Intel Pentium 4/1.6G processor with 256-MB RAM was used.

In order to train and optimize the neural classifier, a set  $\Phi$  of 1700 input-output pairs ( $\Phi = \{(I_i, O_i)\}, i = 1, 2, \dots, 1700$ ) was created by projecting the learning set  $\mathcal{Q}$  (containing patches of  $25 \times 25$  pixels) to linear subspaces using both PCA and FLD methods. Accordingly, the set was also composed of two subsets  $\Phi = \Phi^{\text{pos}} + \Phi^{\text{neg}}$ . The positive subset  $\Phi^{\text{pos}} = \{(I_i^{\text{pos}}, 1)\}$  consisted of feature vectors  $I_i^{\text{pos}}$  computed from the image patches in  $\mathcal{Q}^{\text{pos}}$ , together with the target output classification value  $O_i^{\text{pos}} = 1$ . The other subset  $\Phi^{\text{neg}} = \{(I_i^{\text{neg}}, -1)\}$  consisted of feature vectors  $I_i^{\text{neg}}$  computed from image patches in  $\mathcal{Q}^{\text{neg}}$  and the target output value  $O_i^{\text{neg}} = -1$  of the classifier. This set was further split into a training set of 1400 samples and a test set of 300 samples. The training set was used exclusively to modify the weights. The test set was used as validation set to estimate the generalization ability.

## 4. Results and discussion

### 4.1. ANN optimization and training

We first used an empirical method [21] to determine an upper bound for each layer. Then the optimal number of neurons in the two hidden layers was estimated by independently decreasing the number of hidden neurons in each layer from the upper bound to 1, and evaluating the generalization properties of the ANN on the test set at each step. To avoid entrapment in a local error minimum, every training session was repeated five times and the best weights were used for each number of hidden neurons.

Fig. 4 illustrates the generalization properties of the ANN for different numbers of neurons in the first layer, while keeping the size of the second hidden layer constant at five neurons. The mean squared error (the difference between the actual output and the desired output for the samples in the test set) was plotted vs. the number of neurons. The error rate improved as the number of hidden neurons was increased, but leveled out at around 40 neurons when preprocessed by PCA and 37 neurons by FLD. The experiment was repeated with the number of neurons in the second layer changed from 1 to 10 and similar but worse results were obtained (not shown). Based on above results, we chose 40 neurons for PCA preprocessing and 37 for FLD preprocessing in the first hidden layer and five neurons in the second hidden layer for our subsequent studies.

### 4.2. Microsphere experiments

In order to study systematically the factors that affect recognition accuracy and to compare the relative efficiencies of PCA and FLD preprocessing, we used microspheres

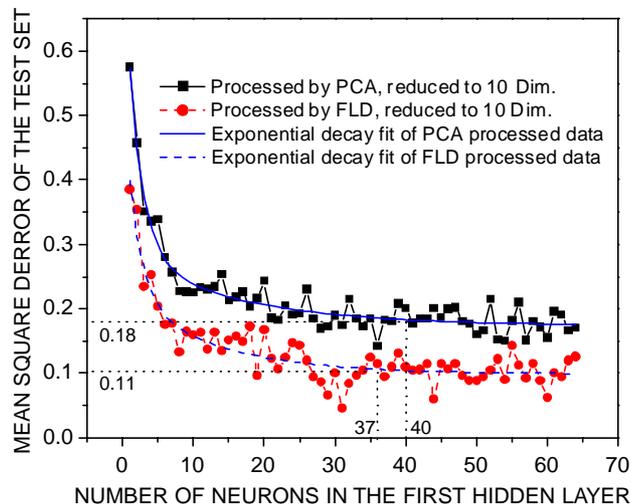


Fig. 4. Effect of neuron number in the first hidden layer on the generalization properties of the ANN. The size of the second hidden layer was kept constant at five neurons.

as model cells. The microspheres are very uniform in size, shape and color and are stable over time. This facilitates experimental reproducibility and makes it possible to create ideal scenes in which critical factors can be individually isolated and well controlled. Furthermore, the ability to create scenes with very small within-class variation by using microspheres permits a test of the hypothesis that FLD gives better performance because it controls within-class variation.

Many experimental factors can affect bright field images of living cells. Among these are variations in focus, illumination, and image noise. These factors could in turn affect cell recognition accuracy. For example, variation in focus is especially important, since it is often the case that there is no single focal plane that is optimal for all the cells in a microscope field. Another factor that could affect the recognition efficiency is the variation in size. In view of

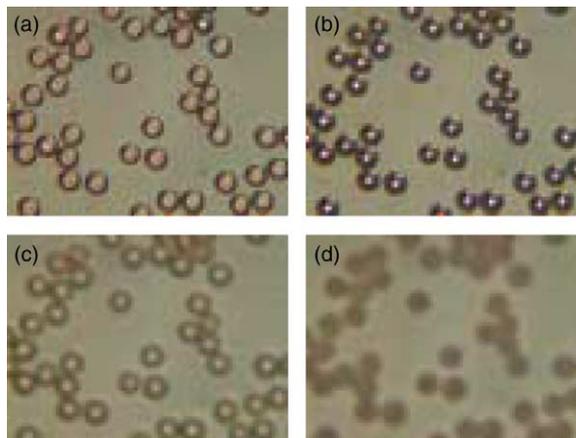


Fig. 5. Typical sample images for focus variation experiment. (a) Focused: the focal plane is at the equator of the microsphere; (b) 12.5  $\mu\text{m}$ : the focal plane is at the supporting surface; (c) 25  $\mu\text{m}$ : the focal plane is 25  $\mu\text{m}$  below the equator and (d) 37.5  $\mu\text{m}$ : the focal plane is 37.5  $\mu\text{m}$  below the equator.

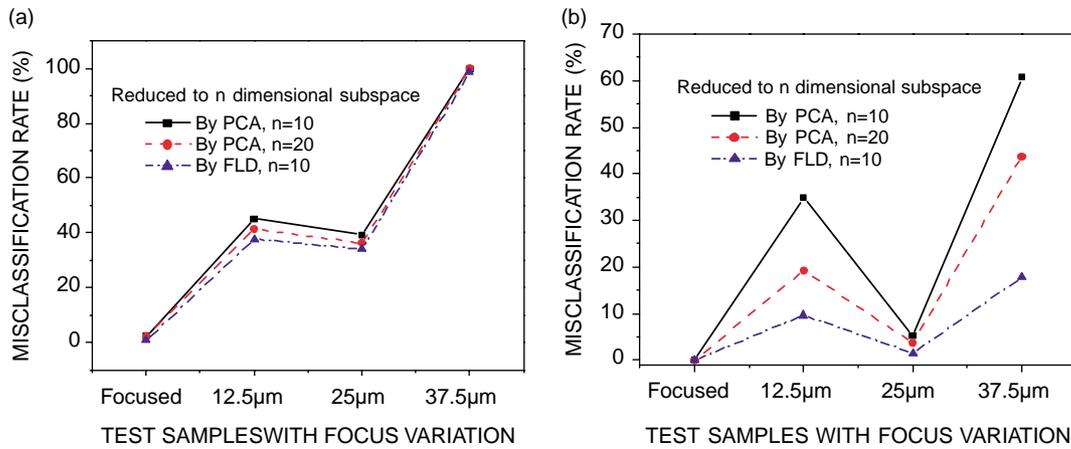


Fig. 6. Misclassification rates with different focus conditions and preprocessing methods. (a) Scheme 1: trained with only focused samples and applied to all samples; and (b) Scheme 2: trained with focused and 25 μm focus variation samples and applied to all samples.

these considerations, we have systematically studied the effects of the above factors on recognition accuracy.

For all microsphere experiments, recognition was performed as described in the METHODS section. For FLD preprocessing, the dimensionality was reduced to 10. For PCA preprocessing, results are shown when both 10 and 20 principal components were used since it has been suggested that more principle components can improve performance [19].

#### 4.2.1. Focus variation

We created four image groups at different focal planes relative to the microsphere equatorial plane to quantify the effects of focus variation, with all other conditions unchanged: (a) *Focused*: the focal plane is at the equator of the microsphere (i.e. 12.5 μm above the supporting surface); (b) 12.5 μm: the focal plane is at the supporting surface; (c) 25 μm: the focal plane is 25 μm below the equator and is within the plastic bottom of the microplate well and (d) 37.5 μm: the focal plane is 37.5 μm below the equator. Typical sample images are shown in Fig. 5. Two experimental schemes were performed on these images. In Scheme 1, each method was trained on the first group and then tested on all groups. In Scheme 2, each method was trained on the first and third group and then tested again on all groups, in which the test on the second group was an interpolation test and on the fourth group was an extrapolation test. Fig. 6 shows the experimental results.

#### 4.2.2. Illumination variation

Images were taken under five light intensity levels of the microscope: (a) *Intensity level 3*: representing extremely weak illumination; (b) *Intensity level 4*: representing weak illumination; (c) *Intensity level 5*: representing normal illumination; (d) *Intensity level 6*: representing strong illumination and (e) *Intensity level 7*: representing extremely strong illumination. Typical sample images from each

level are shown in Fig. 7. Two experimental schemes were performed using these images. To create the situation of small within-class variation, ANNs based on both PCA and FLD were trained with images only in Intensity level 3 and then tested with all levels in Scheme 1. In Scheme 2, within-class variation was purposely introduced by training the neural network with Intensity levels 4, 5 and 6 together and then tested again with all levels. Fig. 8 (a) shows the result of Scheme 1 and Fig. 8 (b) shows that of Scheme 2.

#### 4.2.3. Size (scale) variation

In the size variation experiment, computer generated images of microspheres with 0, 5, 10, 15, and 20% variations in size were used as shown in Fig. 9. Again, two schemes were used to exam the effect of size variation on both PCA and FLD methods. In Scheme 1, ANNs were trained with only microspheres having 0% size variation and tested to all sizes. In Scheme 2, they were trained using images with both 0 and 15% variation. The patch size used

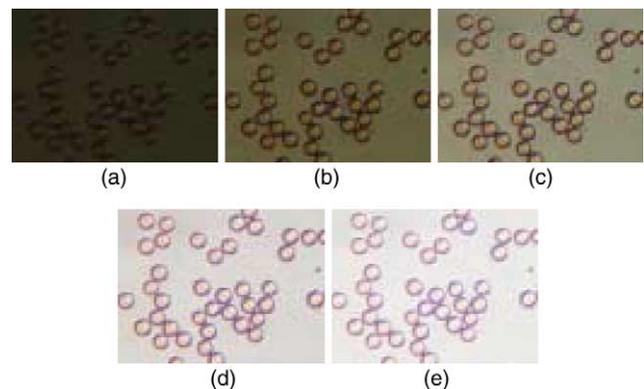


Fig. 7. Typical sample images for illumination variation experiment. Images were taken under five light intensity levels of the microscope. (a) Intensity level 3: representing extremely weak illumination; (b) Intensity level 4: representing weak illumination; (c) Intensity level 5: representing normal illumination; (d) Intensity level 6: representing strong illumination and (e) Intensity level 7: representing extremely strong illumination.

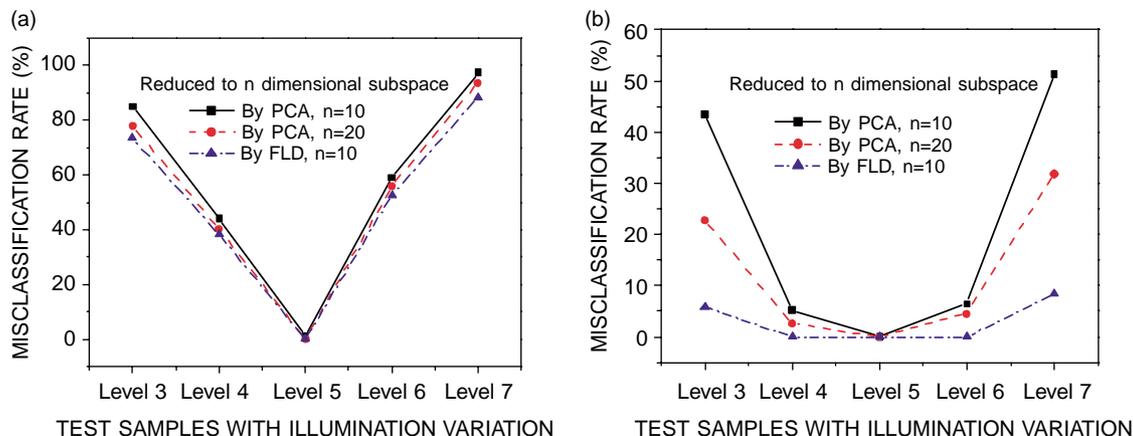


Fig. 8. Misclassification rates with different illumination conditions and preprocessing methods. (a) Scheme 1: trained with only level 5 and applied to all levels; and (b) Scheme 2: trained with level 4, 5 and 6 and applied to all levels.

in both schemes was fixed to a value that was big enough to contain the biggest microspheres. The experimental results of the two schemes are shown in Fig. 10 (a) and (b), respectively.

#### 4.2.4. Noise variation

The noise used in noise variation experiments was zero-mean Gaussian noise with different standard deviations. An image set with five groups of images, each have different noise levels was created by adding computer generated noise to original images. The original images (standard deviation equals zero) belonged to the first group. Groups 2, 3, 4 and 5 contained images in which the standard deviations equaled 15, 30, 45 and 60 respectively. Sample images from each group are shown in Fig. 11. The two experimental schemes were: first, both PCA and FLD were applied to only Group 1 and then tested on all groups. Second, the training set was expanded to include both Groups 1 and 4. Fig. 12 (a) shows the result of Scheme 1 and Fig. 12 (b) shows that of Scheme 2.

Some interesting points were revealed in these experiments. First, as one can see from the results, both PCA and FLD preprocessing performed well if presented with images in the test set, which were selected from the group(s) used for training. This is reasonable because the classifiers have learned very similar data during the training. Second, increasing the number of principal components in PCA preprocessing did improve the performance, but it was still no better than that of FLD. Furthermore, both preprocessing methods performed similarly in Scheme 1 for each of the factors studied, but very differently in Scheme 2, with the error rate of FLD being much less than that of PCA in both interpolation and extrapolation tests. The reason lies in that, for Scheme 1, all images in the training set came exclusively from a single group, in which all microspheres had very homogeneous appearance. Therefore, when we extracted patches from these images and classified them into classes similar to those in Fig. 2, the within-class variations were very

small. As expected, FLD was not superior to PCA in this case, since the variation was almost entirely between-class variation. Scheme 2, on the contrary, purposely introduced within-class variation into the training set by using images from different groups. In this case, the FLD method could learn the variation trend from the training set and choose projection directions that were nearly orthogonal to the within-class scatter, projecting away variations in focus, illumination, size and noise; the PCA method could not. Consequently, the generalization ability of the neural network with FLD preprocessing was greatly improved and substantially better than a similar neural network with PCA preprocessing in Scheme 2-type experiments.

#### 4.3. Living cell experiments

The method introduced in this paper was also systematically studied using digitized microscope images of living cells. We first trained an ANN classifier with the pixel patches in the training set created above and then tested it on whole images of cell cultures. Please note that none of the pixel patches in the training set was extracted from the testing images. The testing images were divided into three groups denoting three different scenarios. Scenario 1 is the simplest case where cells are almost completely separate (i.e. not aggregated) and the background is clean. Scenario 2 is more complex where most cells are attached to each other and there are trash and debris in the background. Scenario 3

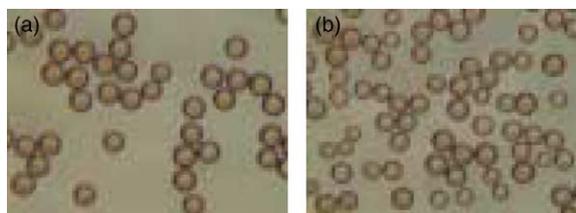


Fig. 9. Typical sample images for size variation experiment. (a) Image of microspheres with no size variation; (b) Image of microspheres with 0, 5, 10, 15 and 20% size variation.

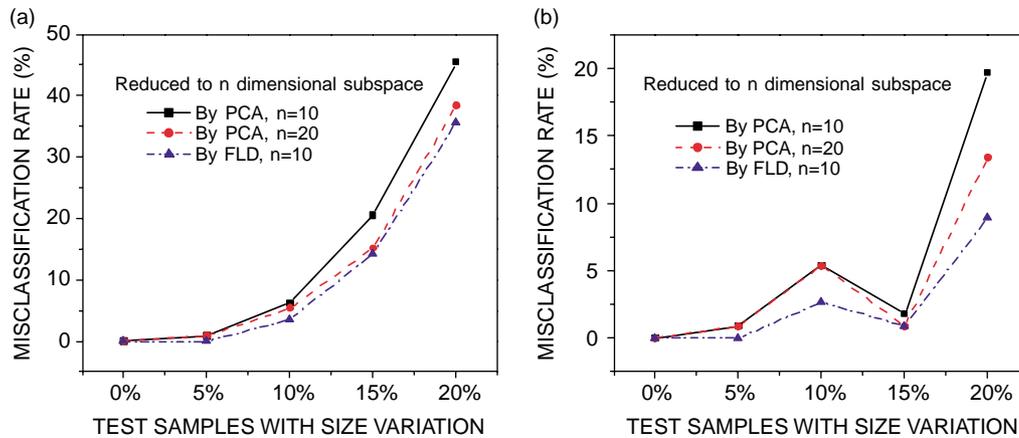


Fig. 10. Misclassification rates with different size variations and preprocessing methods. (a) Scheme 1: trained with only 0% variation samples and applied to all samples; and (b) Scheme 2: trained with 0 and 15% variation samples and applied to all samples.

represents the most complex case where most cells are aggregated together and there is more trash and debris in the background. The three microscope images used in the test are shown in Fig. 13. These images show considerable out of focus blur, cells in clumps occupying multiple focal planes, as well as size variations.

To obtain a standard for evaluation of our classifiers, pre-selected microscope images were evaluated independently by three human experts. Experts were asked to identify objects with the normal appearance of a viable cell and to exclude ghosts of cells, i.e. objects having shape and size similar to viable cells but lower contrast. One list called ‘Human Standard’ was created by merging the three lists of all human experts. To be included in the list, an object had to be identified as a cell by at least two experts.

In experiments with living cells, images were reduced to 10-dimensional subspaces for both PCA and FLD methods. Results obtained with our classifiers were compared to the Human Standard by evaluating *sensitivity* (SE) and *positive predictive value* (PPV) as described previously [9]. The SE of a classifier is defined as the percentage of cells in the reference standard, which are identified by the classifier and the PPV is the percentage of classifier detected cells which are also listed in the reference standard.

The cell positions detected by our classifier are denoted by white crosses in the images (see Fig. 14 for Scenario 3 result—Scenarios 1 and 2 are not shown). The detected cells were carefully compared with human standard. Basically, if a detected cell position is within a maximum distance of 5 pixels from one (and only one) of the cell positions in the human standard, we considered it as a valid detection, otherwise, an error. SE and PPV results of our classifiers are shown in Table 1 and Fig. 15.

The results show that for Scenario 1, which presents a relatively simple case, both PCA and FLD produced very good results. For example, they both achieved SE values of 97.73% and PPV values of 100%. For Scenario 2, where the image is more complex, the SEs of PCA and FLD dropped

to 87.76, 95.92%, respectively, and PPVs dropped to 89.58, 95.92%, respectively, which indicates that the FLD is superior to PCA when the image becomes more complex. This can be seen even more clearly in the very complex case represented by Scenario 3. Here, the SE percentage for FLD is 11.88 greater than that for PCA and the PPV percentage is 8.5 greater.

As noted previously, our results with microspheres suggest that FLD can better generalize from training sets with a single type of confounding factor. The experiments with living cells described in this section clearly show that FLD gives superior generalization even when multiple types of confounding factors are present simultaneously.

It should also be noted that a close inspection of results yielded by our algorithm suggests that it can distinguish between cell ghosts and viable cells, similar to a human observer. In future studies, we will investigate this possibility systematically using dye exclusion/retention techniques to distinguish rigorously between cell ghosts and viable cells.

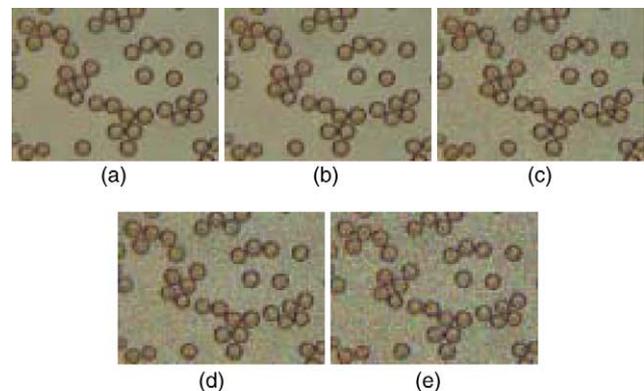


Fig. 11. Typical sample images for noise variation experiment. The noise used was zero-mean Gaussian noise described by different standard deviations. (a) STD=0; (b) STD=15; (c) STD=30; (d) STD=45; (e) STD=60.

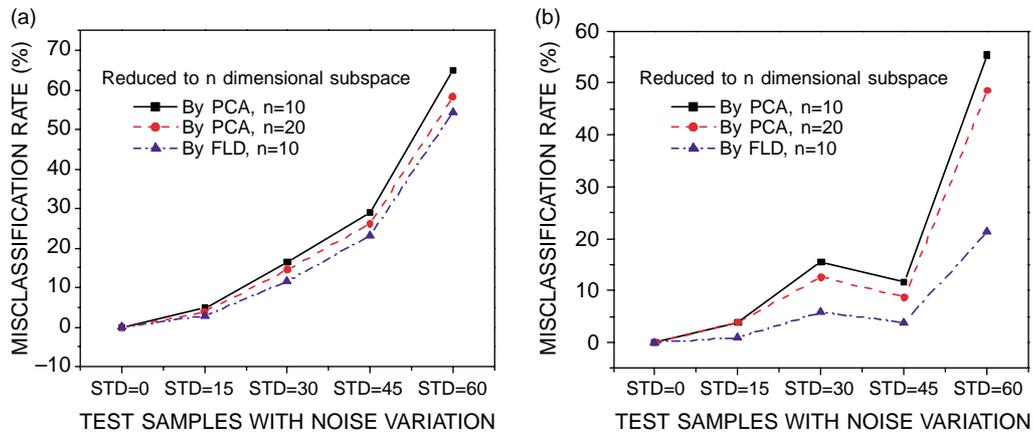


Fig. 12. Misclassification rates with different noise conditions and preprocessing methods. (a) Scheme 1: trained with STD=0 samples and applied to all samples; and (b) Scheme 2: trained with STD=0 and STD=45 samples and applied to all samples.

When our current system is used with a  $25 \times 25$  pixel patch, a  $640 \times 480$  sized image requires a processing time of 1–8 min, depending on the number of cells present. This is judged to be acceptable for some applications. Moreover, we expect substantial speed improvements if the Matlab environment is replaced by dedicated neural network software. Further improvement of speed should be readily available with specialized hardware (e.g. clusters).

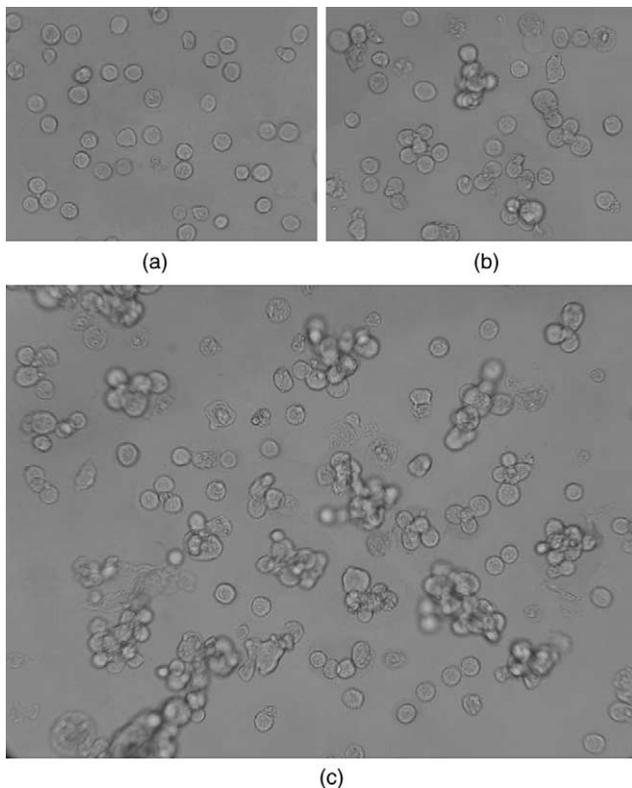


Fig. 13. Testing images used in the living cell experiment. (a) Scenario 1: cells are almost completely separate and the background is clean; (b) Scenario 2: most cells are attached to each other and there are trash and debris in the background; (c) Scenario 3: most cells are clumped together and the background is full of trash and debris.

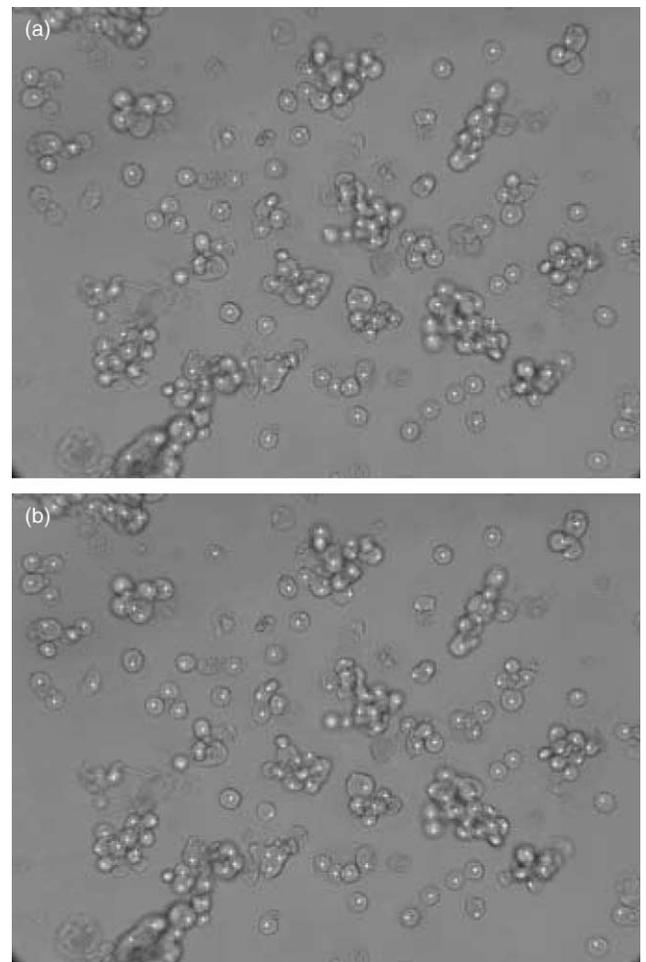


Fig. 14. Detection results of our classifier (Scenario 3). The cell positions detected were denoted by white crosses in the images. (a) By ANN with PCA preprocessing (Sensitivity: 82.5%, Positive predictive value: 83.02%); (b) by ANN with FLD preprocessing (Sensitivity: 94.38%, Positive predictive value: 91.52%).

Table 1  
SE and PPV results for images shown in Fig. 13

		Scenario 1 (%)	Scenario 2 (%)	Scenario 3 (%)
PCA	SE	97.73	87.76	82.5
	PPV	100	89.58	83.02
FLD	SE	97.73	95.92	94.38
	PPV	100	95.92	91.52

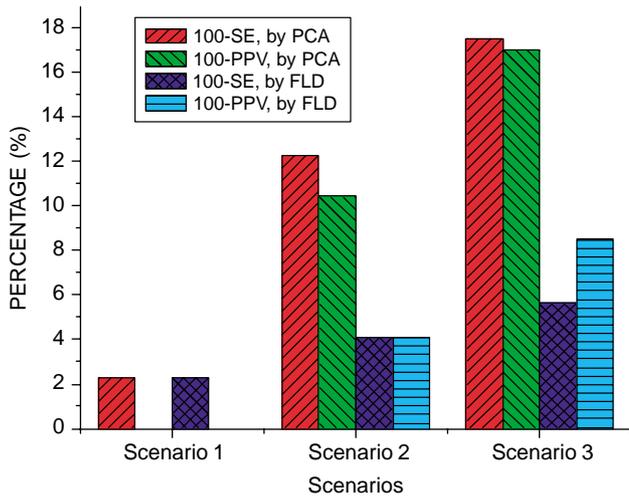


Fig. 15. Sensitivity (SE) and Positive predictive value (PPV) results for the sample images shown in Fig. 13 using different preprocessing methods (shown in the form of 100-SE and 100-PPV).

## 5. Conclusions

An effective algorithm for cell recognition in bright field microscopy has been introduced in this paper. Bright field microscopy frees the limited number of available fluorescence channels for other purposes. In addressing the variability of cell size and morphology, as well as variations in microscope parameters, such as focus and illumination, a class specific method, Fisher's linear discriminant (FLD), was clearly superior to Principal Component Analysis (PCA) in all four types of variations studied. This holds true for both microsphere and actual cell recognition. The primary reason is that FLD optimizes the solution by maximizing the ratio of between-class scatter to within-class scatter, while PCA maximizes the effect of total scatter only.

## References

- [1] T.M. Mitchell, Machine learning, McGraw-Hill, New York, 1997.
- [2] P. Sajda, C. Spence, L. Parra, A multi-scale probabilistic network model for detection, synthesis and compression in mammographic image analysis, *Medical Image Analysis* 7 (2003) 187–204.
- [3] Y.-L. Fok, J.C.K. Chan, R.T. Chin, Automated Analysis of Nerve-Cell Images Using Active Contour models, *IEEE Transactions Medical Imaging* 15 (3) (1996) 353–368.
- [4] S., Shigetoshi, et al., Cell recognition by image processing. (Recognition of dead or living plant cells by neural network), *JSME International Journal, Series C: Dynamics, Control, Robotics, Design and Manufacturing* 37 (1) (1994) 202–208.
- [5] V.A. Kovalev, A.Y. Grigoriev, Hyo-Sok Ahn, N.K. Myshkin, Segmentation technique of complex image scene for an automatic blood cell counting system, *SPIE* 2710 (1996) 805–810.
- [6] G. Constantinos, Constantinos G. Loukas, George D. Wilson, Borivoj Vojnovic, Alf Linney, An image analysis-based approach for automated counting of cancer cell nuclei in tissue sections, *Cytometry Part A* 55A (2003) 30–42.
- [7] P.J. Sjöström, B.R. Frydel, L.U. Wahlberg, Artificial neural network-aided image analysis system for cell counting, *Cytometry* 36 (1999) 18–26.
- [8] M.V. Boland, R.F. Murphy, A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of hela cells, *Bioinformatics* 17 (2001) 1213–1223.
- [9] T.W. Nattkemper, T. Twellmann, H. Ritter, W. Schubert, Human vs. machine: evaluation of fluorescence micrographs, *Computers in Biology and Medicine* 33 (2003) 31–43.
- [10] T.W. Nattkemper, H. Ritter, W. Schubert, A neural classifier enabling high-throughput topological analysis of lymphocytes in tissue sections, *IEEE Transactions Information Techniques Biomedicine* 5 (2) (2001) 138–149.
- [11] T.W. Nattkemper, H. Wersing, W. Schubert, H. Ritter, A neural network architecture for automatic segmentation of fluorescence micrographs, *Neurocomputing* 48 (2002) 357–367.
- [12] T. Kämpfe, T.W. Nattkemper, H. Ritter, Combining independent component analysis and self-organizing maps for cell image classification, *Proceedings DAGM Lecture Notes in Computer Science* 2191 (2001) 262–268.
- [13] L. Sirovich, M. Kirby, Low dimensional procedure for the characterization of human faces, *Journal of the Optical Society of America* 4 (3) (1987) 519–524.
- [14] H. Murase, S. Nayar, Visual learning and recognition of 3D objects from appearance, *International Journal of Computer Vision* 14 (1995) 5–24.
- [15] M. Turk, P. Pentland, Face recognition without features, *Proceedings IAPR Workshop on Machine Vision Applications, Tokyo* (1990) 267–270.
- [16] M. Turk, P. Pentland, Eigenfaces for recognition, *Journal of Cognitive Neuroscience* 3 (1) (1991) 71–96.
- [17] S. Baker, S. Nayar, Pattern rejection, *Proceedings IEEE Conference Computer Vision and Pattern Recognition* (1996) 544–549.
- [18] Y. Cui, D. Swets, J. Weng, Learning-based hand sign recognition using SHOSLIF-M, *International Conference on Computer Vision* (1995) 631–636.
- [19] P.N. Belhumeur, J.P. Hespanha, J. Kriegman, Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection, *IEEE Transactions on PAMI* 19 (7) (1997) 711–720.
- [20] C.B. Moler, W. Stewart, An algorithm for generalized matrix eigenvalue problems, *SIAM Journal of Numerical Analysis* 10 (1973) 99–130.
- [21] A.S. Pandya, R.B. Macy, Pattern recognition with neural networks in C++, CRC Press, Boca Raton, Fla, 1996.
- [22] W.L. Cleveland, I. Wood, B.F. Erlanger, Routine large-scale production of monoclonal antibodies in a protein-free culture medium, *Journal of Immunological Methods* 56 (1983) 221–234.
- [23] M.B. Kelz, G.W. Dent, S. Therianos, P.G. Marciano, et al. Single-cell antisense RNA amplification and microarray analysis as a tool for studying neurological degeneration and restoration, *Science Aging Knowledge Environ* 2002(1):rel.