Rapid Communications

Automated Recognition of Patterns Characteristic of Subcellular Structures in Fluorescence Microscopy Images

Michael V. Boland,[2] Mia K. Markey,[1] and Robert F. Murphy[1]*

[1]Department of Biological Sciences and [2]Biomedical Engineering Program, Center for Light Microscope Imaging and Biotechnology, Carnegie Mellon University, Pittsburgh, Pennsylvania

Methods for numerical description and subsequent classification of cellular protein localization patterns are described. Images representing the localization patterns of 4 proteins and DNA were obtained using fluorescence microscopy and divided into distinct training and test sets. The images were processed to remove out-of-focus and background fluorescence and 2 sets of numeric features were generated: Zernike moments and Haralick texture features. These feature sets were used as inputs to either a classification tree or a neural network. Classifier performance (the average percent of each type of image correctly classified) on previously unseen images ranged from 63% for a classification tree using Zernike moments to 88% for a backpropagation neural network using a combination of features from the 2 feature sets. These results demonstrate the feasibility of applying pattern recognition methods to subcellular localization patterns, enabling sets of previously unseen images from a single class to be classified with an expected accuracy greater than 99%. This will provide not only a new automated way to describe proteins, based on localization rather than sequence, but also has potential application in the automation of microscope functions and in the field of gene discovery. Cytometry 33:366–375, 1998. © 1998 Wiley-Liss, Inc.

Key terms: protein localization; subcellular location; pattern recognition; microscopy, fluorescence; neural networks (computer); Zernike moments

The goal of the work we describe here is to develop methods that allow the numerical description and subsequent classification of the patterns characteristic of subcellular structures in fluorescence microscope images of eukaryotic cells. Such data are generated on a regular basis by labeling one or more cellular molecules with fluorescent dyes (most often by using antibodies against specific proteins). As currently practiced, investigators identify patterns based on experience or via comparison with patterns of known proteins. The question we address is whether these patterns can be described in a way that is amenable to further processing by a computer, thereby enabling automation of their analysis.

The extensive literature on pattern recognition describes its application to a wide variety of systems, but only sporadically to automated microscope image analysis. While the screening of Pap smears (4) has received significant attention from the pattern recognition community, the goal of recognizing potentially cancerous cells in a background of normal tissue stained with hematoxylin and eosin is inherently different from the problem of identifying a fluorescence pattern as being from one of a number of distinct classes.

Over the past 10 years, a number of automated systems for acquisition and analysis of fluorescence microscope images have been described. These efforts have been primarily directed towards image cytometry (15,20,22), in which the goal is obtaining accurate measurements of the total fluorescence of each cell, or towards automating fluorescence in situ hybridization (24,29), in which the goal is determining the number of fluorescence ''spots'' (chromosomes) in each cell. While some image cytometry systems provide the ability to calculate numerical features from the fluorescence distribution for each cell, these are usually used to identify cell types (e.g., distinguish lymphoid from myeloid cells) (15,29) rather than to describe the subcellular pattern per se. Thus, features appropriate for describing protein localization patterns have not been previously characterized in the context of fluorescence microscopy.

In considering various pattern recognition applications as a starting point for classifying protein localization patterns, we encountered a parallel in the field of handwritten character recognition. The problems are similar, in that while there are distinct classes of images (numbers and letters, organelle-specific localization patterns), there is also considerable variability within each class (individual versions of the number ''2'' can be quite different; the appearance of the Golgi apparatus varies from cell to cell). Approaches that can recognize individual handwritten characters have been described (2,17), and we therefore modeled our initial work on character recognition, and subsequently incorporated other approaches.

We set the following goal: to determine whether a pattern classification system could be developed (using feature sets not chosen with particular subcellular localization patterns in mind) that was able to correctly classify a set of up to 20 previously unseen images from a single class with an expected accuracy of greater than 99%. We describe here systems that are capable of recognizing fluorescence patterns representing the subcellular distribution of 5 different probes that meet the desired goal. The immediate biological implications of this work are that automated systems may be used to classify patterns for unknown proteins (and thereby can be used to select cell lines showing particular subcellular patterns). In the longer term, the work we describe demonstrates the feasibility of creating (in an automated manner) a systematics for protein localization patterns.

MATERIALS AND METHODS
Fluorescence Microscopy

All reagents were obtained from Sigma Chemical Co. (St. Louis, MO) unless otherwise indicated. Chinese Hamster Ovary (CHO) cells were grown for 2–3 days in α-MEM and 10% (v/v) calf serum (Intergen Co., Purchase, NY) on 19-mm cover slips coated with 0.1% (w/v) type I collagen in 0.1 M acetic acid. They were then fixed for 10 min with 2% paraformaldehyde in phosphate-buffered saline (PBS: 140 mM NaCl, 2.6 mM KCl, 8.1 mM Na_2HPO_4, 1.5 mM KH_2PO_4, 0.9 mM $CaCl_2$, 0.5 mM $MgCl_2$, pH 7.4), permeabilized for 10 min with 0.1% saponin in cytoskeletal stabiliza-

tion buffer (CSB: 137 mM NaCl, 5 mM KCl, 1.1 mM Na_2HPO_4, 0.4 mM KH_2PO_4, 4 mM $NaHCO_3$, 2 mM $MgCl_2$, 2 mM EGTA, 5 mM Pipes, 0.1% glucose, pH 6.1) and incubated for 60 min with a primary antibody. After 3 washes of 5 min each in CSB, the cells were incubated for 45 min with 12.5 μg/ml of a Cy5-conjugated secondary antibody (Jackson Immunoresearch, West Grove, PA) and 50 μg/ml Hoechst 33258 (Molecular Probes, Inc., Eugene, OR). The coverslips were washed 3 more times in CSB before mounting on microscope slides using gelvatol (60 ml of 10 mM Tris, 15 g Airvol 205 (Air Products, Allentown, PA), 30 ml glycerol, 1 g n-propyl gallate). Images of Cy5 and Hoechst 33258 fluorescence were acquired separately using a Zeiss Plan-Neofluar objective (100$\times$, 1.3 NA), and a Photometrics CH 250 cooled charge-coupled device (512 $\times$ 382 pixels, 23 μm/pixel) mounted on a customized Zeiss Axiovert microscope (9).

Monoclonal antibodies directed against the Golgi protein giantin (21), the lysosomal protein LAMP2 (11), the yeast nucleolar protein NOP4 (27), and tubulin (Sigma) were used as primary antibodies in separate labeling experiments. Working dilutions of antibody stock solutions were obtained by empirically optimizing for low background in the presence of adequate specific signal.

Each slide was scanned for single cells that were spread out on the coverslip (i.e., not rounded up in mitosis). Each such field of view was acquired as a stack of 3 images in which the focus was changed by 0.237 μm between each slice. The image collection is available at http: //www.ste. cmu.edu/murphylab/data.

Image Processing

The images were processed by first applying nearest neighbor deconvolution (1) to each 3-image stack in order to remove out of focus fluorescence from the central image plane. The next step involved manually defining rectangular regions of each deconvolved image that contained single cells. Only the pixels from the single, deconvolved image that were within this region were subject to further processing steps. The background fluorescence, defined as the most common pixel value in the region, was subtracted from all pixels. Finally, the images were thresholded using a constant multiple of the background fluorescence for that image. This multiple was 4 for all probes except Hoechst 33258, for which it was 1.5. These numbers were arrived at empirically by assessing the quality of images thresholded using various values. Pixels at or above this threshold were used in subsequent processing steps, those below the threshold were set to 0. In order to make feature calculations insensitive to changes in overall image brightness, each pixel value in the thresholded image was divided by the total fluorescence in that image.

Zernike Features

Two steps were required to convert a rectangular region to a unit circle for calculation of Zernike moments. First, the ''center of fluorescence'' (center of mass) for each image was calculated and used to define the center of the

pixel coordinate system. Second, since the Zernike polynomials are defined over a circle of radius 1, the x and y coordinates were divided by 150 (this corresponds to the size of an average cell at the magnification used in our experiments). Only pixels within the unit circle of the resulting normalized image, $f(x,y)$, were used for subsequent calculations. The Zernike moments, Z_{nl}, for an image were calculated using

$$Z_{nl} = \frac{n+1}{\pi} \sum_x \sum_y V_{nl}^*(x, y)\, f(x, y) \qquad (1)$$

where $x^2 + y^2 \leq 1$, $0 \leq l \leq n$, $n - l$ is even, and $V_{nl}^*(x, y)$ is the complex conjugate of a Zernike polynomial of degree n and angular dependence l, and

$$V_{nl}(x, y)$$
$$= \sum_{m=0}^{(n-l)/2} (-1)^m \frac{(n-m)!}{m! \left[\frac{(n-2m+l)}{2}\right]! \left[\frac{(n-2m-l)}{2}\right]!}$$
$$\cdot (x^2 + y^2)^{(n/2)-m} e^{il\theta} \qquad (2)$$

where $0 \leq l \leq n$, $n - l$ is even, $\theta = \tan^{-1}(y/x)$, and $i = \sqrt{-1}$.

We calculated the Zernike moments through degree 12 (Z_{nl} such that $n \leq 12$ in Eqn.1.). Since the moments themselves are complex numbers and are sensitive to rotation of the image, we used the magnitudes of the moments as features (i.e.) $|Z_{nl}|$ (17). This provided 49 descriptive features for each image.

Haralick Texture Features

Haralick's texture features (12) were calculated using the kharalick function of the cytometry toolbox (10) for Khoros (version 2.1 Pro, Khoral Research, Inc., Albuquerque, NM; http://www.khoral.com). We did not calculate the maximal correlation coefficient due to computational instability and were therefore left with 13 texture features for each image.

Feature Selection

In order to choose a subset of features from the combined Haralick and Zernike sets described above, we applied 2 feature selection methods to the training data. The first was accomplished with the STEPDISC procedure in SAS (SAS Institute, Cary, NC). We used the default parameters of the procedure, which is an implementation of stepwise discriminant analysis (19). One of the important defaults is the use of stepwise selection, which starts with an empty set of features and at each step adds to it the best feature not currently in the set, while also allowing features that are no longer among the best to be removed. Wilks' lambda statistic is used as the criterion to decide whether a feature should be added to or removed from the set of selected features.

The second method used a modified version of the multiple discriminant analysis criterion (8). We selected those features that had the largest ratio of the variance of that feature calculated using all samples in the training set to the sum of the variances of that feature calculated for each class (i.e., image type) in the training set

$$\frac{\text{var}(\mathbf{f})}{\sum_c \text{var}(\mathbf{f}_c)} \qquad (3)$$

where $\mathbf{f}_c$ contains only feature values from class c, and $\mathbf{f}$ contains features from all image classes.

Classification

Before classification, the image feature data were separated into distinct training and test sets in order to assess performance on images not seen by the classifier during training. Numbers of train/test images for each class were as follows: giantin, 47/30; Hoechst, 39/30; LAMP2, 37/60; NOP4, 25/8; tubulin, 25/26. These values were chosen so that we would have no less than 25 images from any class available for training purposes. After this separation, the training data were used to calculate the mean and variance of each feature. These values were then used to normalize the training data to have a mean of 0 and a variance of 1 for each feature. The same mean and variance were then used to normalize the test data (the resulting means and variances for the test set therefore differed somewhat from 0 and 1, respectively). The normalized training and test sets were used with the neural network classifier, and the non-normalized sets were used with the classification tree.

Classification trees were implemented using the tree function of S-Plus (version 3.4 for the HP 9000, MathSoft, Seattle, WA). The tree-generating algorithm was allowed to run to completion using the 173-image training set, and the performance of that tree on the 154 images in the test set was recorded. Test images that were assigned equally to more than 1 class were considered to be "unknown."

Backpropagation neural networks were implemented using PDP++ (http://www.cnbc.cmu.edu/PDP++). Networks were configured with the number of inputs equal to the number of features being used at any particular time, 20 hidden nodes (unless specified otherwise), and 5 output nodes (1 for each class of input). The learning rate was empirically chosen to be 0.1, and the momentum was 0.9. The desired outputs of the network for each training sample were defined as 0.9 for the node corresponding to the input class and 0.1 for the other nodes. To minimize any bias in the training and testing process, the aforementioned test set was divided into 8 pairs of "stop" and "evaluation" sets (each pair contained one-eighth of the test images in the evaluation set and the remainder of the test images in the stop set). Training of the network was stopped when the sum of squared error for a particular stop set reached a minimum, where the error of a particular output node is defined as the difference be-
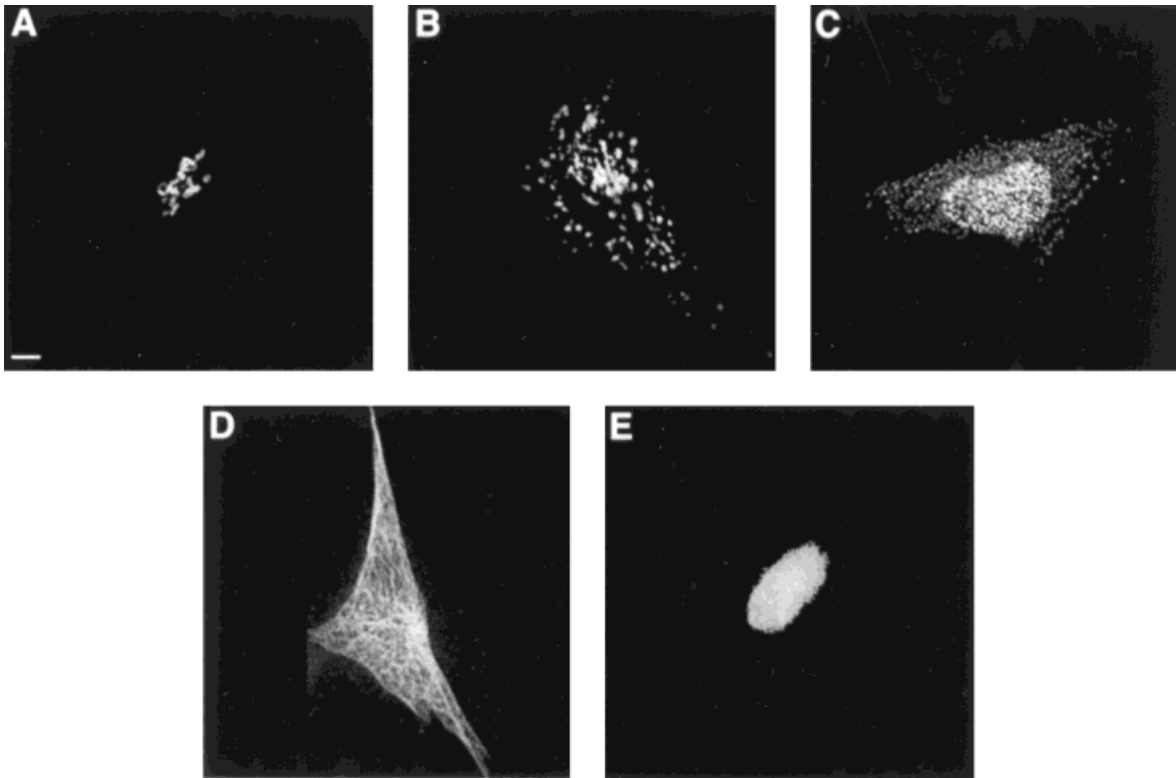
FIG. 1. Examples of the images used as input to the classification systems described in the text. These images have had background fluorescence subtracted and have had all pixels below threshold set to 0. Representative images are shown for cells labeled with antibodies against giantin (**A**), LAMP2 (**B**), NOP4 (**C**), tubulin (**D**), and with the DNA stain Hoechst 33258 (**E**). Scale bar = 5 µm.

tween its desired and actual output values. The performance of the network at the stopping point was measured using the corresponding evaluation set. This process was repeated for the 8 pairs of stop and evaluation sets, and the classification results combined to generate confusion matrices. When measuring the performance of the network using the evaluation data, each sample was classified as belonging to the class corresponding to the largest of the 5 output values.

Reconstruction From Zernike Moments

As described previously (17,28), reconstructed images, $\hat{f}(x, y)$, were generated from Zernike moments using

$$\hat{f}(x, y) = \sum_{n=0}^{n_{max}} \sum_{l} Z_{nl} V_{nl}(x, y) \tag{4}$$

where $0 \leq l \leq n$, $n - l$, is even, and n_{max} is the highest degree of moments used (12 in our case).

RESULTS
Image Collection and Processing

We started by collecting images of CHO cells showing 5 distinct subcellular patterns. Cells were grown to subconfluent levels on collagen-coated microscope coverslips, fixed in paraformaldehyde, and permeabilized with sapo-nin. The cells were incubated with 1 of 4 primary antibodies (chosen to yield qualitatively different patterns), and stained with Hoechst 33258 (to label the nucleus) in parallel with a fluorescently conjugated secondary antibody. The antibodies used were against giantin (a Golgi protein), LAMP2 (a lysosomal protein), tubulin (a cytoskeletal protein), and NOP4 (an *S. cerevisiae* nucleolar protein). (The antibody against NOP4 crossreacted with a CHO protein mainly located in the nucleus, but also found in the cytoplasm.) Coverslips were searched for fields containing cells that were well spread and separated from their neighbors. A stack of 3 images was then taken in which the focus was adjusted by a small amount between each image in the stack. Slides were processed in this way until there were enough digital images available to train and test the classification schemes described below. There were 33 to 97 images available for each class of fluorescence distribution.

Image stacks were deconvolved to remove out of focus fluorescence, cropped to a rectangular region containing a single cell, corrected for background fluorescence, and thresholded as described in Materials and Methods.

Sample images for each class of pattern are shown in Figure 1. These images were chosen to represent their respective classes using a feature-based method for picking a representative image (M.K. Markey, M.V. Boland, and R.F. Murphy, submitted for publication). Prior to subse-
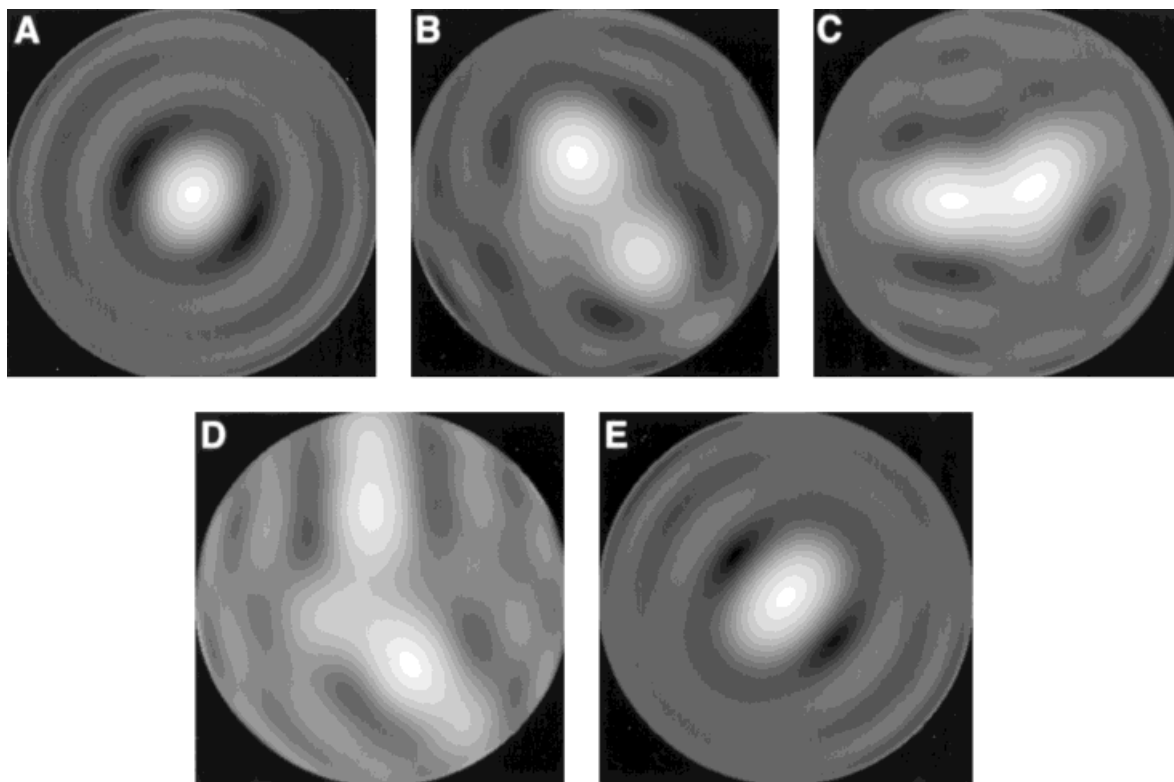
FIG. 2. Reconstructions of fluorescence images from Figure 1 using the first 49 Zernike moments. **A**: giantin. **B**: LAMP2. **C**: NOP4. **D**: tubulin. **E**: Hoechst 33258.

quent feature extraction steps, the images were segregated into distinct training and test sets.

Zernike Feature Extraction and Image Reconstruction

Arguably the most important step in pattern recognition is the appropriate choice of numbers to represent an image (such numerical descriptors of an image are called features). Since a long-term goal is a system that is able to distinguish the localization of many proteins, not just the 5 patterns we used in this study, we decided to utilize 2 sets of "general purpose" features rather than choosing individual features to discriminate particular patterns.

Since cells in fluorescence images have arbitrary location and orientation, we sought features that were invariant to the translation and rotation of cells within a field of view. This search led us first to moment invariants (14), and then to the more appealing Zernike moments (28,30) (Eq. 1). Although originally used in the description of optical aberration (5,30), the Zernike polynomials, on which the Zernike moments are based, have recently found application in pattern recognition (2,3,16–18,25). Based on previous work, we chose to calculate the Zernike moments up to degree 12 ($n \le 12$ in Eq. 1), giving us 49 numbers describing each image. These complex valued moments are not invariant to rotation, so our final 49 features were obtained by calculating their magnitudes (which are). The magnitudes of the 49 moments are very

different from one another, and this difference hindered subsequent classification when using a neural network classifier (see below). We therefore normalized the features as described in Materials and Methods before using them with the BPNN classifier.

Since the Zernike polynomials are an orthogonal basis set (a set of functions for which the integral of the product of any pair of functions is 0), it is possible to use the Zernike moments calculated for a particular image to reconstruct that image. In theory, reconstruction of a continuous (i.e., not pixelated) image without error requires an infinite number of Zernike moments. Since we used only 49 moments to describe our images, it was of interest to examine representative images reconstructed from those moments. The reconstructions (Fig. 2) provide some insight into the amount of information that is included in the 49 Zernike moments used for classification (it is clear that much of the detailed information in each image is not preserved in the low-degree moments). Note, in particular, that the 5 reconstructed images are visibly different, despite representing more than 1,400 to 1 compression of the circular region defined around each cell (70,650 pixels per 300 pixel diameter circle vs. 49 Zernike moments).

Classification Using Zernike Features

We initially sought a means of visualizing the degree of separation of the 5 image classes in the high dimension
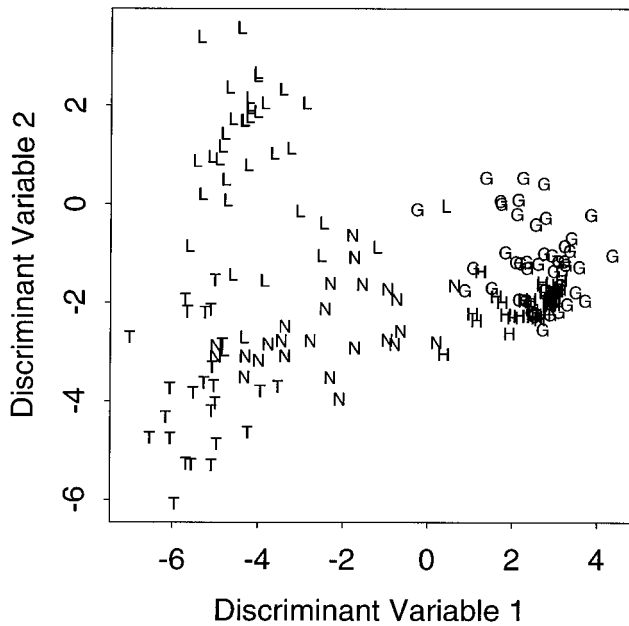
FIG. 3. Visualization of the 5 image classes in a two-dimensional projection of the 49-dimensional Zernike feature space. The training data were transformed using linear discriminant analysis, and then the first 2 linear discriminant variables were plotted for each image (G, giantin; H, Hoechst; L, LAMP2; N, NOP4; T, Tubulin).

space provided by the Zernike moments. To this end, we applied linear discriminant analysis to the features. Using the training data and the discr function in S-Plus, we obtained a new set of variables that are linear combinations of the original features and which are generated to maximize the resulting ratio of interclass spread to the sum of intraclass spreads. (It is worth noting that this optimization criteria leads to a different choice of linear combinations than would be obtained by principal components analysis, which does not consider the class of an observation and which maximizes the variation contained in successive linear combinations.) A scatter plot of the first 2 linear discriminant variables for all observations in the training set is shown in Figure 3. While there is some separation of the classes using the first 2 discriminant variables, there is also significant overlap between classes (some of these overlaps were resolved in plots of the third and fourth discriminant variables, data not shown). However, the limited discriminating ability of these variables did not prove useful for classifying previously unseen images, in that the test data set did not show similar clustering of the classes when plotted using the same linear transformations (data not shown).

We proceeded to use 2 methods of classification that are able to generate more complex decision boundaries than linear discriminant analysis. The first of these was the classification tree (6). The goal is to divide the multidimensional feature space with decision boundaries (in this case linear and parallel to the feature axes) such that images of each class are largely separated from each other. The appealing characteristic of this classifier is that it generates

an interpretable tree structure as output that includes rules for correctly recognizing each class of input. By following a particular sample down the tree, it is also possible to determine which features are able to discriminate that sample from those of other classes. A classification tree was generated using all of the training data with the default options of the S-Plus tree function.

Once the tree was generated, the test data were applied to it, and performance was assessed by generating a confusion matrix from the resulting classifications (Table 1). A confusion matrix is generated by determining where a classifier is "confused" about the classification of particular images. The row of a particular entry indicates the true classification of those images, while the column represents the class to which those images were assigned by the classifier. Non-zero values in the off-diagonal elements of the matrix therefore indicate mistakes made by the classifier. The average performance of the classification tree using Zernike features was 65%, where average performance is calculated as the mean of the values along the diagonal of the confusion matrix. The performance is acceptable for all classes except tubulin, which was frequently confused with LAMP2. The results suggest that the classification tree was trained to recognize the training data too closely (and thus, like the linear discriminant variables above, did not perform well on previously unseen data). Using the prune.tree function of S-Plus and an empirically-chosen cost-complexity measure, we were able to increase the performance of the classification tree such that all classes were recognized at a rate greater than 50% and the average classification rate was 63% (data not shown). Because the performance was not entirely satisfactory and because the pruning required was not systematic, we proceeded to implement a more sophisticated classifier.

A classifier that is widely used and is implemented in various commercial and freely distributed software is the backpropagation neural network (BPNN) (23). We chose the BPNN as our second classifier because it is able to generate decision boundaries that are significantly more complex than the rectilinear boundaries of the classification tree (13). A disadvantage to the BPNN is that the ready interpretability of the classification tree is lost. The BPNN was implemented in PDP++ as a 3-layer network, with 49 inputs (1 for each Zernike feature), 20 hidden nodes, and 5 output nodes (1 for each class of image.) The network was fully connected between layers.

The network was trained such that only results from the training data set were used to modify the network weights. To visualize the training process, the sum of squared error between the desired value of the output nodes and the actual value of those nodes was calculated at regular intervals. The error for the training data was calculated after each training epoch and the error for the stop data was determined after every third training epoch. One epoch of training is defined as a single pass through all of the samples in the training data. In order to prevent over training and therefore "memorization" of the training data, training was stopped when the sum of squared error

Table 1
Confusion Matrix Generated From the Output of a Classification Tree Trained and Tested
With the Zernike Features*

| True classification | Output of classification tree | | | | | | Number of images |
	Giantin	Hoechst	LAMP2	NOP4	Tubulin	Unknown	
Giantin	**80%**	3%	7%	7%	0%	3%	30
Hoechst	20%	**80%**	0%	0%	0%	0%	30
LAMP2	10%	3%	**62%**	10%	15%	0%	60
NOP4	0%	0%	25%	**75%**	0%	0%	8
Tubulin	0%	0%	69%	0%	**27%**	4%	26

*Images assigned to more than one class by the tree are included in the "Unknown" category. Due to rounding, rows do not always sum to 100%.

Table 2
Confusion Matrix Generated From the Output of a
Backpropagation Neural Network Trained and Evaluated
With the Zernike Features*

| True classification | Output of neural network | | | | | Number of images |
	Giantin	Hoechst	LAMP2	NOP4	Tubulin	
Giantin	**97%**	0%	3%	0%	0%	30
Hoechst	3%	**93%**	0%	3%	0%	30
LAMP2	12%	2%	**70%**	10%	7%	60
NOP4	0%	0%	0%	**88%**	13%	8
Tubulin	0%	0%	12%	4%	**85%**	26

*Due to rounding, rows do not always sum to 100%.

value for the stop data was at a minimum. At this point, the evaluation data were applied to the network and the output node of the network with the largest value was defined as the classification result for each evaluation example. Results are shown in Table 2. The average rate of correct classification for this method, 87%, is significantly better than the classification tree. The BPNN clearly enhanced our ability to classify the fluorescence images we had generated.

If this average classification rate seems inadequate, the following should be noted. First, a random classifier (one that is completely unable to discriminate between the image classes) would be expected to produce an average classification rate of only 20%. Second, it is possible to take advantage of the nature of the samples used for imaging to improve on this result. Specifically, if one prepares a homogeneous sample from a single class (i.e., identically prepared cells) and uses a majority rule classification scheme where the sample is classified using the result obtained for the majority of individual cells studied, it is possible to improve the classification rate. Treating the single-cell classifications as Bernoulli random variables (i.e., each cell is classified correctly or it is not) results in the following formula for the probability that the majority classification is correct

$$P_{\text{majority}}(n) = \sum_{x=\lfloor \frac{n}{2} \rfloor + 1}^{n} \binom{n}{x} p^x (1-p)^{n-x} \qquad (5)$$

where n is the number of cells examined and p is the probability of a correct classification of a single image. This analysis relies on the fact that there is no class for which the classifier achieves less than 50% correct classification. This level of performance is easily met by the backpropagation neural network classifier. For a sample size of 10 cells with a single-cell classification rate of 87%, a majority rule classifier will result in a 99% correct classification rate for the sample as a whole.

The results above utilized more than 300 images generated using 5 different labels and the acquisition of these images therefore represents a significant investment of time and resources. To gain some insight into how few images per class might be needed to train a useful classifier, we generated smaller training sets. Ten samples per class proved to be too few as the LAMP2 images were classified correctly only 40% of the time (data not shown), violating the requirement for a minimum of 50% corrrect classification discussed in the analysis surrounding Eqn. 5. Using 15 samples per class, however, provided reasonable results. In this case the correct classification rates were 83% for giantin, 93% for Hoechst, 62% for LAMP2, 63% for NOP4, and 73% for tubulin, for an average rate of 75%. These values, while not ideal for single-cell classification, can certainly be useful with the majority rule classification approach described above. For the localization patterns discussed here, these results also indicate that it is possible to train useful classifiers using less than 100 total images for 5 classes.

Classification Using Haralick Texture Features

We also explored the use of other types of numerical features for classification of protein localization patterns. To do so, we chose a set of descriptive features that are fundamentally different from the Zernike moments, the texture features described by Haralick (12). These were selected because they are invariant to translations and rotations, and because they describe more intuitive aspects of the images (e.g., coarse versus smooth, directionality of the pattern, image complexity, etc.) using statistics of the gray-level co-occurrence matrix for each image.

Since the BPNN classifier proved more effective than the classification tree when using the Zernike features, it was used with the texture features. This time the network had only 13 inputs but still had 20 hidden nodes and 5 output nodes, all fully connected. Training was carried out as

Table 3
*Confusion Matrix Generated From the Output of a Backpropagation Neural Network Trained and Evaluated With the Haralick Texture Features**

True classification	Output of neural network					Number of images
	Giantin	Hoechst	LAMP2	NOP4	Tubulin	
Giantin	**97%**	0%	3%	0%	0%	30
Hoechst	3%	**93%**	3%	0%	0%	30
LAMP2	8%	0%	**82%**	8%	2%	60
NOP4	0%	0%	13%	**88%**	0%	8
Tubulin	4%	0%	8%	8%	**81%**	26

*Due to rounding, rows do not always sum to 100%.

Table 4
*Confusion Matrix Generated From a Backpropagation Neural Network Trained and Evaluated With the 10 Best Features From the Zernike Moments and Haralick Texture Features, as Determined Using Eq. 3**

True classification	Output of neural network					Number of images
	Giantin	Hoechst	LAMP2	NOP4	Tubulin	
Giantin	**97%**	0%	3%	0%	0%	30
Hoechst	3%	**97%**	0%	0%	0%	30
LAMP2	12%	0%	**83%**	2%	3%	60
NOP4	0%	0%	13%	**88%**	0%	8
Tubulin	0%	0%	19%	4%	**77%**	26

*Due to rounding, rows do not always sum to 100%.

before and the results are shown in Table 3. The average performance of this feature set/classifier combination, 88%—corresponding to a predicted accuracy of 99.6% for majority rule on 10 images—is very close to that of the Zernike moment/BPNN approach. Note that this performance is accomplished with far fewer features describing each image, 13 texture features versus 49 Zernike moments.

Feature Selection and Reduction of Classifier Complexity

In a further attempt to reduce the dimensionality of the feature set, we chose a subset of 10 features from the combined Zernike and Haralick features using the stepwise discriminant analysis functionality (i.e., the STEPDISC procedure) of SAS (SAS Institute, Cary, NC). This method uses Wilks' lambda statistic to iteratively determine which variables are best able to discriminate the classes. Using these 10 features with a BPNN containing 20 hidden nodes resulted in correct classification rates of 97% for giantin, 93% for Hoechst, 82% for LAMP2, 88% for NOP4, and 54% for tubulin. Although the performance on the first 4 classes is identical to the Haralick features alone, the performance on the tubulin images is significantly worse (81% vs. 54%) and drops the average classification rate to 83%. We considered the performance of the stepwise discriminant procedure to be unsatisfactory in this case.

As an alternative, we identified a different subset using Eqn. 3; this procedure selects those features that, on average, widely separate the classes from each other while at the same time keep the individual classes tightly clustered. Although 7 of the 10 features selected using this approach are the same ones selected using stepwise discriminant analysis, the performance of the BPNN using these 10 features (Table 4) was better (88% vs. 83%). This result is significant because it indicates that it is possible to achieve performance at least equal to the best single feature set using a smaller number of features selected from both feature sets.

To determine whether the Zernike and Haralick features could be used successfully with less complex neural networks, we measured the performance of networks having less than 20 hidden nodes. To expedite the testing of the various networks, we used the entire test set to both stop training and evaluate the classification performance

rather than splitting the test set into multiple stop/evaluate pairs as described in Materials and Methods. At no point were test samples used to modify the network weights, however. We found good correlation between the train/test and train/stop/evaluate approaches where they were used with the same training data and same number of hidden nodes. We therefore use the 2-set approach as a screening method when training networks under multiple conditions. Whereas the classification performance using the Zernike moments dropped from 87% with 20 hidden nodes to 83% with 10 to 78% with 5, the Haralick features maintained essentially constant performance, dropping only from 88% at 20 hidden nodes to 87% with 5. The Haralick result was confirmed using the more rigorous three set train/stop/evaluate method; the average performance was 84%. The maintenance of classification rate with fewer hidden nodes indicates that the classification problem is relatively "easier" with the Haralick features than with the Zernike moments. The decrease in feature number, from 49 to 13, and the decrease in the number of required hidden units, 20 vs. 5, both help to make the Haralick features the more desirable of the 2 feature sets studied here.

DISCUSSION

The localization of a protein to a particular subcellular structure or organelle is an important step in the study of that protein. It is common for investigators to use 1 of a number of protein tagging techniques (e.g., epitope tagging, fusion with green fluorescent protein, generation of antibodies), along with fluorescence microscopy, to visualize and record the localization pattern of a protein. The major reason for doing this is that the localization of a protein may provide insight into its function (e.g., the observation that the product of a gene implicated in vacuole biogenesis is located in the nucleus suggests that it is a transcription factor) or lend support to hypotheses regarding its function (the observation that a protein suspected to play a role in nuclear pore function localizes to the nucleus supports the hypothesis). The current state of the art in protein localization relies on individual investigators to make reasoned conclusions regarding the patterns obtained via the microscope. While this approach has worked adequately, improvements need to be made to

accommodate the rapidly increasing number of proteins that are discovered and characterized every year.

One way to improve upon the methods currently used in describing protein localization is to quantitatively describe the patterns. It is useful to make an analogy to the advances made in sequence analysis after quantitative comparison methods were developed. Analysis of new protein or nucleic acid sequences initially relied on visual inspection of sequences for regions of identity or homology to previously known sequences. Even after computerized methods for comparing sequences were developed, the statistical significance of matches was not always evaluated. Currently, it is a simple matter to sequence a gene or cDNA and send the resulting sequence to a server that is capable of comparing it to existing sequences in a wide variety of organisms. The results from this comparison can provide almost immediate insight into the possible structure and function of the new protein. With the work described here, we anticipate a time when visual comparison and analysis of protein localization patterns will be as rare as visual analysis of protein or nucleic acid sequences.

There are a number of advantages of an automated system for describing and classifying protein localization patterns. First, quantitative description of images facilitates a standardization that is not currently possible. Just as it is now possible to obtain a measure of homology between 2 sequences, we anticipate measuring the homology of the localization of 2 proteins. Second, databases can be constructed that will allow for immediate comparison of a new localization pattern with many existing patterns. In this way, it will be possible to see which other proteins localize in a manner similar to the one under study. Such information is currently unavailable. Third, the set of protein localization patterns obtained from classification of all known proteins will give insight into the complexity of protein localization mechanisms. At present, for example, the number of distinct intracellular patterns exhibited by surface receptors is unknown.

While we are not able to accomplish these ambitious goals at present, the work we describe here contributes in at least 2 ways towards attaining them: we describe 2 methods of numerically describing protein localization patterns, and we demonstrate that such descriptions are useful. Since the Zernike moments and the Haralick texture features were used to identify the corresponding images as being from 1 of a number of distinct classes, we know that the numbers calculated capture useful information. Given the large body of work in the field of pattern recognition, it is likely that other approaches to quantitative image description could be developed. Exploration of these approaches will be one focus of future work. These potential improvements are not necessary, however, to begin considering and solving biological problems using the approaches described here.

In addition to having potential utility for incorporating localization information into molecular biology databases, the methods we describe here may be of value in a number of automated or ''high-throughput'' screening approaches. First, automated methods are needed to screen the vast number of compounds now available as potential drugs. The classification of protein localization patterns will be of use here as a means of identifying those cells that have responded in a desired way to the application of a drug. It will be possible, for instance, to automatically identify only those cells in which the applied compound traps a surface receptor in the ER, or in which that compound prevents translocation of a transcription factor to the nucleus. Second, it should be possible to automatically screen for cells displaying a mutant phenotype. In this case, one might ask the computer to identify those cells that have a malformed Golgi apparatus. Third, one might screen a population of live cells for those members that are in a particular stage of the cell cycle, image those cells repeatedly until the event under study is complete, and then begin screening again.

A last potentially interesting application of automated localization analysis is in the area of gene discovery. By using molecular techniques to randomly insert visualizable tags into a wide variety of genes, it is possible to generate localization patterns for a large number of proteins, some known and some unknown. Once a large number of cells has had a single protein tagged, images of protein localization can be collected. This approach has been used to determine localization patterns for randomly-tagged genes from yeast using gene fusions with either LacZ (7) or Green Fluorescent Protein (26). Automation of the pattern analysis would potentially speed this approach, and one can then conceive of a localization database for all expressed proteins in yeast. This is beyond current capabilities for organisms with larger genomes, but screens for proteins with particular patterns can be imagined (e.g., ER proteins). While this may be carried out manually, the number of patterns requiring screening in this scheme is large and automated identification of patterns of interest is desirable. The results we present here suggest that such an approach is feasible.

LITERATURE CITED

1. Agard DA, Hiraoka Y, Shaw P, Sedat JW: Fluorescence microscopy in three dimensions. In: Fluorescence Microscopy of Living Cells in Culture, Methods in Cell Biology, Vol. 30, Taylor DL, Wang Y-L (eds). Academic Press, Inc., San Diego, CA, 1989, pp. 353–377.
2. Bailey RR, Mandyam S: Orthogonal moment features for use with parametric and non-parametric classifiers. IEEE Trans Pattern Anal Machine Intell 18:389–399, 1996.
3. Belkasim SO, Shridhar M, Ahmadi M: Pattern recognition with moment invariants: A comparative study and new results. Pattern Recognit 24:1117–1138, 1991.
4. Birdsong GG: Automated screening of cervical cytology specimens. Hum Pathol 27:468–481, 1996.
5. Born M, Wolf E: Principles of Optics, 2nd Edition. Pergamon Press Ltd., Oxford, England, 1964.
6. Breiman L, Friedman JH, Olshen RA, Stone CJ: Classification and Regression Trees. Wadsworth and Brooks/Cole, Monterey, CA, 1984.
7. Burns N, Grimwade B, Ross-Macdonald PB, Choi EY, Finberg K, Roeder GS, Snyder M: Large-scale analysis of gene expression, protein localization, and gene disruption in Saccharomyces cerevisiae. Genes Dev 8:1087–1105, 1994.

8. Duda RO, Hart PE: Pattern Classification and Scene Analysis. John Wiley & Sons, New York, 1973.

9. Farkas DL, Baxter G, DeBiasio RL, Gough A, Nederlof MA, Pane D, Patek DR, Ryan KW, Taylor DL: Multimode light microscopy and the dynamics of molecules, cells, and tissues. Ann Rev Physiol 55:785–817, 1993.

10. Fleming MG: Design of a high resolution image cytometer with open software architecture. Anal Cell Pathol 10:1–11, 1996.

11. Granger BL, Green SA, Gabel CA, Howe CL, Mellman I, Helenius A: Characterization and cloning of lgp110, a lysosomal membrane glycoprotein from mouse and rat cells. J Biol Chem 265:12036–12043, 1990.

12. Haralick RM: Statistical and structural approaches to texture. Proc IEEE 67:786–804, 1979.

13. Hornick K, Stinchcombe M, White H: Multilayer feedforward networks are universal approximators. Neural Netw 2:359–366, 1989.

14. Hu M-K: Visual pattern recognition by moment invariants. IRE Trans Inf Theory IT 8:179–187, 1962.

15. Jaggi B, Poon SS, MacAulay C, Palcic B: Imaging system for morphometric assessment of absorption or fluorescence in stained cells. Cytometry 9:566–572, 1988.

16. Khotanzad A, Hong YH: Invariant image recognition by Zernike moments. IEEE Trans Pattern Anal Machine Intell 12:489–497, 1990.

17. Khotanzad A, Hong YH: Rotation invariant image recognition using features selected via a systematic method. Pattern Recognit 23:1089–1101, 1990.

18. Khotanzad A, Lu J-H: Classification of invariant image representations using a neural network. IEEE Trans Acoust Speech Signal Process 38:1028–1038, 1990.

19. Klecka WR: Discriminant analysis. In: Quantitative Applications in the Social Sciences, Vol. 19, Sullivan JL (ed). Sage University Paper, Beverly Hills and London, 1980.

20. Lee BR, Haseman DB, Reynolds CP: A digital image microscopy system for rare-event detection using fluorescent probes. Cytometry 10:256–262, 1989.

21. Linstedt AD, Hauri HP: Giantin, a novel conserved Golgi membrane protein containing a cytoplasmic domain of at least 350 kDa. Mol Biol Cell 4:679–693, 1993.

22. Lockett SJ, Jacobson K, Herman B: Quantitative precision of an automated, fluorescence-based image cytometer. Anal Quant Cytol Histol 14:187–202, 1992.

23. McClelland JL, Rumelhart DE: Parallel Distributed Processing: Explorations in the Microstcuture of Cognition. Vol. 1. The MIT Press, Cambridge, MA, 1986.

24. Nederlof PM, van der Flier S, Verwoerd NP, Vrolijk J, Raap AK, Tanke HJ: Quantification of fluorescence in situ hybridization signals by image cytometry. Cytometry 13:846–852, 1992.

25. Perantonis SJ, Lisboa PJG: Translation, rotation, and scale invariant pattern recognition by high-order neural networks and moment classifiers. IEEE Trans Neural Netw 3:241–251, 1992.

26. Sawin KE, Nurse P: Identification of fission yeast nuclear markers using random polypeptide fusions with green fluorescent protein. Proc Natl Acad Sci USA 93:15146–15151, 1996.

27. Sun C, Woolford JL, Jr.: The yeast NOP4 gene product is an essential nucleolar protein required for pre-rRNA processing and accumulation of 60S ribosomal subunits. EMBO J 13:3127–3135, 1994.

28. Teague MR: Image analysis via the general theory of moments. J Opt Soc Am 70:920–930, 1980.

29. Vrolijk H, Sloos WC, van de Rijke FM, Mesker WE, Netten H, Young IT, Raap AK, Tanke HJ: Automation of spot counting in interphase cytogenetics using brightfield microscopy. Cytometry 24:158–166, 1996.

30. Zernike F: Beugungstheorie des Schneidenverfarhens und Seiner Verbesserten Form, der Phasenkontrastmethode. Physica 1:689–704, 1934.