Chapter XXI
Machine Learning In Morphological Segmentation

O. Lezoray
Université de Caen Basse-Normandie, France

G. Lebrun
Université de Caen Basse-Normandie, France

C. Meurie
INRETS-LEOST, France

C. Charrier
Université de Caen Basse-Normandie, France

A. Elmotataz
Université de Caen Basse-Normandie, France

M. Lecluse
Centre Hospitalier Public du Cotentin, France

H. Elie
Centre Hospitalier Public du Cotentin, France

ABSTRACT

The segmentation of microscopic images is a challenging application that can have numerous applications ranging from prognosis to diagnosis. Mathematical morphology is a very well established theory to process images. Segmentation by morphological means is based on watershed that considers an image as a topographic surface. Watershed requires input and marker image. The user can provide the latter but far more relevant results can be obtained for watershed segmentation if marker extraction relies on prior knowledge. Parameters governing marker extraction varying from image to image, machine learning approaches are of interest for robust extraction of markers. We review different strategies for extracting markers by machine learning: single classifier, multiple classifier, single classifier optimized by model selection.

INTRODUCTION

The visual evaluation of microscopic slides is a tedious task, which requires hard concentration of the pathologist screening the specimen under study. With the advent of image processing as an efficient way to extract objects of interest in images, the automatic analysis of images acquired from light microscopes has become an emerging and challenging image analysis application. Microscopic image analysis schemes are usually threefold: image segmentation, objects features computation, objects classification. We propose to focus on the first part of this scheme for the segmentation of microscopic images of bronchial cytology stained by the international coloration standard of Papanicolaou and acquired in light microscopy. The aim of the segmentation is to extract cells in images; cells being composed of a nucleus and a cytoplasm. Segmentation being in general a difficult task, machine learning has emerged as a key component of intelligent computer vision programs when adaptation is needed (Arif, 2007). In this Chapter, we consider the segmentation of microscopic images by morphological methods and show how to integrate machine learning into a morphological segmentation scheme.

BACKGROUND

Mathematical Morphology is a very well established theory to process images (Serra, 1988). The watershed is the basic tool of Mathematical Morphology for segmentation. It has proved to be a powerful tool and it is used in a large number of applications, such as, medicine, remote sensing, robotics, and multimedia (Meyer, 2001). The parameters for a watershed are marker and input images (Soille, 2004). The watershed grows the markers based on a flooding simulation process by considering the input image as a topographic surface. The problem is to produce the divide-line image on this surface (Roerdink, 2000). Each marker is associated to a color. The topography is flooded from below by letting colored water rise from the holes with its associated color, at an uniform rate across the entire image. When the rising water of distinct colors would merge, a dam is built to prevent the merging. Figure 1 illustrates such a process on a color hematology image with two different sets of markers (provided by the user or by a machine learning algorithm). The most difficult problem when using watershed is of course the definition of appropriate markers with minimal efforts (Rivest, 1992; Meyer, 2001). User provided markers can be attractive for interactive segmentation but for automatic segmentation other techniques have to be considered. An accurate extraction of reliable markers requires prior knowledge on the latter (color, texture, shape, etc.). To incorporate such prior knowledge for the automatic extraction of markers, machine-learning techniques (Derivaux, 2007; Lezoray, 2002; Levner 2007) are the most natural candidates. Figure 2 provides a schematic view of all components involved in the design of a morphological segmentation scheme relying on machine learning algorithms for marker extraction. To perform morphological color image segmentation, a machine learning based classification of pixel feature vectors is done. The result is labeled in connected components and refined by a color watershed. To infer a proper machine learning based pixel classifier, an image database with an associated ground truth is constructed and pixel feature vectors are shared among classes as a basis for supervised learning. In the following Sections, conceiving of each one of these components is described.

MACHINE LEARNING IN MORPHOLOGICAL SEGMENTATION OF MICROSCOPIC IMAGES

Machine Learning

Far more relevant results can be obtained for watershed segmentation if marker extraction relies

Figure 1. Segmentation by watershed: a) original hematology image, b) 3D view of the image as a topographic surface c) user-defined inner and outer marker after connected components labeling, d) user-defined markers superimposed, e) superimposed regions of color watershed with c) as markers, f) Machine Learning based marker extraction after connected components labeling, g) Machine Learning based markers superimposed, h) superimposed regions of color watershed with f) as markers. Regions colored in black in marker images ((c) and (f)) correspond to unlabeled pixels, other pixels correspond to region seeds.

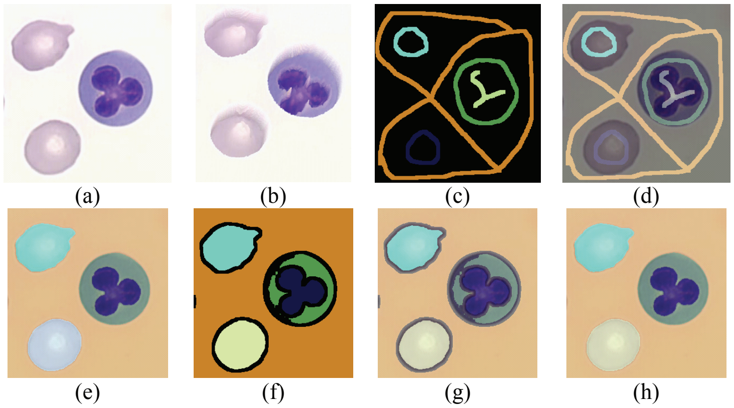
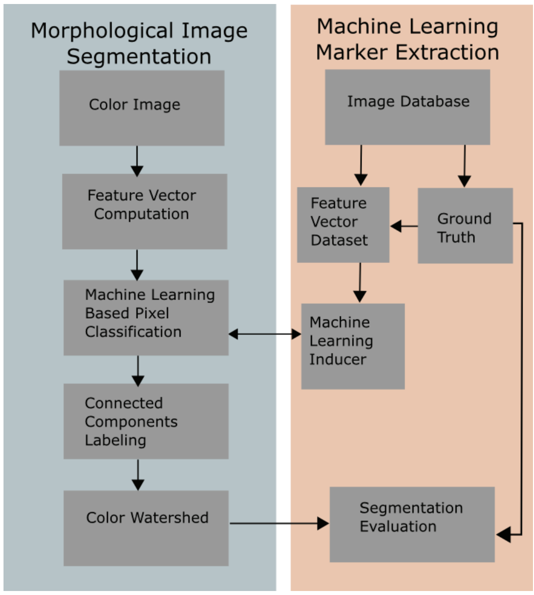


Figure 2. Schematic view of components involved in the design of a morphological segmentation scheme relying on machine learning algorithm for marker extraction.


Morphological Image Segmentation
Color Image
Feature Vector Computation
Machine Learning Based Pixel Classification
Connected Components Labeling
Color Watershed
Machine Learning Marker Extraction
Image Database
Feature Vector Dataset
Ground Truth
Machine Learning Inducer
Segmentation Evaluation

on prior knowledge. Parameters governing marker extraction varying from image to image, machine learning approaches are of interest for robust extraction of markers. However, with the use of machine learning algorithms for the extraction of seeds, one can consider either unsupervised or supervised learning approaches. Unsupervised approaches do not make use of any learning step and supervised methods do need a learning step to infer an appropriate model of the data. The two approaches (unsupervised or supervised learning) have benefits and defects, and it is often difficult to make a choice between both. A machine learning algorithm is a recognition tool called a classifier that provides class memberships information for a vector received in input. Therefore, a classifier aims at building a function *F* that maps the input feature space to an output space. Each example *(x, y)* consists of an instance $x \in X \subset \mathbb{R}^p$ and a label $y \in Y = \{\omega_1,...,\omega_K\}$ where *X* is the feature vector input space and *K* the number of classes to be discriminated. A classifier can be considered as a mapping *F* from instances to labels *F*: *X* → *Y*. Usually, classifiers provide class membership estimates $f(x, \omega_i)$ and the classification of an input feature vector *x* is performed by:

$$F(x) = \arg\min_{\omega_i} f\left(x, \omega_i\right)$$

We assume that a dataset *D* of *n* examples in a real-dimensional space $\mathbb{R}^p$ is provided. In the sequel, we review some classical unsupervised and supervised classifiers (Duda, 2000).

k-Means

Among unsupervised clustering formulations that are based on minimizing a formal objective function, the most widely used is probably k-means clustering (Linde, 1980). It consists in finding a set of *k* examples of $\mathbb{R}^p$ called centers, so as to minimize the mean squared distance from each data example to its nearest neighbor. To classify an input feature vector *x*, the class memberships are estimated in terms of Euclidean distance:

$$f\left(x, \omega_i\right) = \left\| x - c_{\omega_i} \right\|$$

where c_{ω_i} denotes the center of class ω_i and ‖.‖ is the Euclidean distance.

Fisher Linear Discriminant Analysis (FLDA)

Fisher's linear discriminant is an unsupervised classification method that projects high-dimensional data onto a line (denoted by *w*) and performs classification in this one-dimensional space (Fisher, 1936). To find this projection, one maximizes the following objective:

$$J(w) = \frac{w^T S_B w}{w^T S_W w}$$

where S_B is the between classes scatter matrix and S_W the within classes scatter matrix. To classify an input feature vector *x*, the class memberships are estimated by:

$$f\left(x, \omega_i\right) = w^T x + b_i$$

where b_i is a threshold deduced from prior probabilities.

Bayesian Classifier

This classifier is based on the Bayesian decision theory (Duda, 2000). It is a supervised statistical approach to pattern classification that assumes that the decision problem is expressed in probabilistic terms. For multivariate distributions, mixtures of Gaussian distribution models are used. To classify an input feature vector *x*, the class memberships are estimated for each class by the equation in Box 1. where μ_{ω_i} denotes mean attribute vector, Σ_{ω_i} is conditional covariance matrix of class ω_i

Box 1.

$$f\left(x,\omega_i\right) = -\frac{1}{2}\left(x-\mu_{\omega_i}\right)^T \sum\nolimits_{\omega_i}^{-1}\left(x-\mu_{\omega_i}\right) - \frac{1}{2}\log\left(\left|\Sigma_{\omega_i}\right|\right) - \log\left(p_{\omega_i}\right) + \frac{K}{2}\log\left(2\pi\right)$$

(normally distributed) and p_{ω_i} prior probability of class ω_i.

k-NN

The *k* Nearest Neighbors method is a well known supervised algorithm in the field of machine learning (Michie, 1994). Given a training set and a distance defined in the feature space, the basic k-NN rule consists in searching for the *k* nearest neighbors of an input feature vector. The estimated class probabilities are proportional to the number of classes among the *k* nearest neighbors:

$$f\left(x,\omega_i\right) = \frac{\left|S\left(x,\omega_i,k\right)\right|}{k}$$

where $S(x, \omega_i, k)$ denotes the set of patterns among the *k* nearest neighbors to point *x* that belong to class ω_i and |.| denotes the cardinal of a set.

Support Vector Machines (SVM)

SVM are supervised classifiers based on the structural risk minimization principle from statistical learning theory (Cristianini, 2000; Vapnik, 1998). SVM express predictions in terms of a linear combination of kernel functions on a subset of the training data, known as support vectors. SVM map an input vector *x* into a high-dimensional feature space *H* through some nonlinear mapping function ϕ(.) and builds an optimal separating hyper-plane in that space. The mapping is performed by a kernel function *k(.,.)* that defines an inner product in *H*. A typical kernel is Gaussian kernel:

$$k\left(x_1,x_2\right) = \exp\left(-\frac{\left\|x_1-x_2\right\|^2}{2\sigma^2}\right)$$

This reduces the training of a SVM to maximizing a convex quadratic form subject to linear constraints. The maximum margin separating hyper-plane can be represented as a linear combination of training points called support vectors (SV):

$$w = \sum_{i=1}^{n} \alpha_i^* y_i \phi\left(x_i\right)$$

An example of the training set is a support vector if $\alpha_i^* \geq 0$. Many specific algorithms can solve the convex quadratic problem of SVM, the most competitive being Sequential Minimal Optimization (Platt, 1998). The training algorithm produces a decision function where each support vector has a α_i value characterizing his weight on the hyper plane position. The output of a SVM for a given input feature vector *x* is:

$$f(x) = \sum_{i \in SV}^{n} \alpha_i^* y_i k\left(x_i, x\right)$$

The output of a SVM is not a probabilistic value, but non-calibrated distance measurement of an example to the separating hyper-plane. Platt proposed a method to map SVM outputs into positive class posterior probabilities by applying a sigmoid function to the SVM output (Platt, 1999):

$$f\left(x,\omega_i\right) = \frac{1}{1+\exp\left(Af\left(x\right)+B\right)}$$

where *A* and *B* are obtained by minimizing the negative log like hood under a test set. Finally, SVM are binary classifiers and multi-class decision functions are usually designed by combining several two-class SVM decision functions (Hsu, 2002).

Multi Layer Perceptrons (MLP)

A MLP is a supervised classifier expressed by a network of simple neurons called perceptrons. The perceptron computes a single output from multiple real-valued inputs by forming a linear combination according to its input weights and then possibly putting the output through some nonlinear activation function (Rosenblatt, 1958). A single perceptron is not very useful because of its limited mapping ability. The perceptrons can, however, be used as building blocks of a larger, much more practical structure. A typical multilayer perceptron (MLP) network consists of a set of source nodes forming the input layer, one or more hidden layers of computation nodes, and an output layer of nodes. The input signal propagates through the network layer-by-layer (Dreyfus, 2005). The computations performed by such a feed forward network with a single hidden layer are:

$$f\left(x,\omega_i\right)=g\left(\sum_{j=0}^{n_h} w_{ij}\, g\left(\sum_{k=0}^{n_d} w_{jk} x_k\right)\right)$$

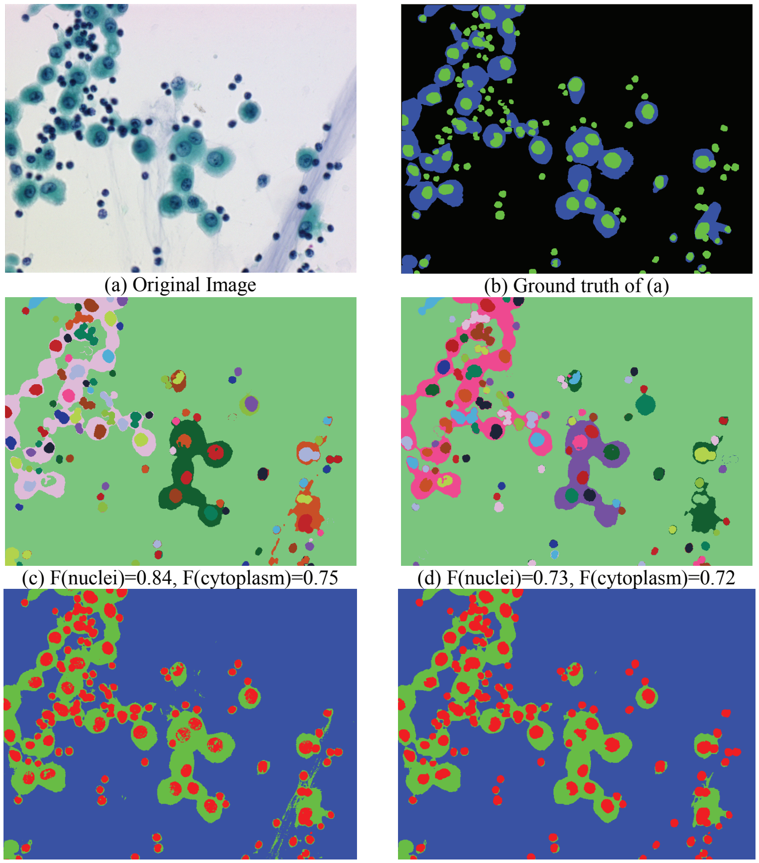
Where w_{ij} denotes the weight between neurons *i* and *j*, n_h the number of hidden units, n_d the number of input units and *g* is a nonlinear activation function (e.g. a sigmoid). The supervised learning problem of the MLP can be solved with the back-propagation algorithm.

Image Database

When one wants to conceive a segmentation method for a given type of microscopic color images, it is essential to utilize a data set of representative images. This shows several key benefits for the conception and the evaluation of a complete segmentation scheme. For the considered class of microscopic images, a microscopy expert has to choose judicious images that well describe the whole segmentation problem: all the objects to be extracted (and further segmented) are present on at least one image. Once the representative images are determined, they are manually segmented in several classes and objects of interest are extracted. This enables the constitution of a database of segmented images (a ground truth). This ground truth is associated to a set of pixel feature vectors shared in several classes that a machine-learning algorithm has to learn to categorize. Therefore, a ground truth database can be used for the learning step of supervised machine-learning algorithms (Bayes, k-NN, SVM, MLP) and also as a reference segmentation to evaluate the relevance of an automatic segmentation. In the sequel, we will consider a publicly available database of 8 images from bronchial cytology that have been manually segmented (Meurie, 2005).

The use of machine-learning algorithms to extract seed from images comes to perform pixel feature vector classification. Each pixel of the ground truth is associated to a feature vector *x* and a class *y*. All the couples *(x,y)* associated to ground truth pixels define a dataset of examples and a machine learning algorithm has to infer a mapping function *F* as close as possible to ground truth. A machine-learning algorithm used to categorize pixels in images produces image classification and not image segmentation. In an image classification, classes are assigned to pixels that are not necessarily spatially connected. To obtain segmentation from an image classification result, one has therefore to perform a labeling of connected components. Figure 3 (a)-(b) show a microscopic color image from bronchial cytology and its ground truth where pixels have been classified into three classes (background, cytoplasm, nuclei). One the opposite, Figure 3 (c)-(d) show

Figure 3. Morphological segmentation of a microscopic color image (a) with a Bayesian classifier for extracting seeds: in YCh_1Ch_2 (c) and RGB (d) color spaces. Figure 3(b) presents the ground truth of Figure 3(a) where three classes of pixels are shown (black for background, blue for cytoplasm, and green for nuclei). F-measures for the nuclei and cytoplasm classes are provided. Figure 3(e) and (f) respectively present results of a pixel classification by SVM and further refined by a watershed. q_{shape} and q_{seg} measures are provided to globally quantify the segmentation quality.



(a) Original Image

(b) Ground truth of (a)

(c) F(nuclei)=0.84, F(cytoplasm)=0.75

(d) F(nuclei)=0.73, F(cytoplasm)=0.72

(e) Classification by SVM (q_{shape}=*0,769* and q_{seg}=*7,33*)

(f) Machine Learning based morphological segmentation (q_{shape}=*0,492* and q_{seg}=*4,58*)

typical image segmentations where connected components have distinct labels.

Moreover, since color is the main information used to classify pixels, different feature vectors can be considered by using different color representations in different colors spaces (Lukac, 2006). This change of color representation can have high influence on final results. Figure 3(c)-(d) show two different segmentation results where the feature vector associated to pixels is either a vector in YCh_1Ch_2 color space (Figure 3(c)) or a vector in *RGB* color space (Figure 3(d)).

Classification and Segmentation Evaluation

An important issue when designing a segmentation method is the evaluation of results. Having a ground truth obviously facilitates evaluation but several precautions have to be undertaken. A classical way to assess the performance of a classifier that produces a classification on image pixels is to compute a classification rate for each class.

Let F be a machine learning algorithm given feature vector inputs X and class outputs Y. To quantify the error rate of a machine-learning algorithm, a loss function is defined to assess if the prediction realized is close to the ground truth. The error rate (*ER*) is defined by (Duda, 2000):

$$ER(X,Y,F)=\frac{1}{n}\sum_{n} l\left(F\left(x_i\right),y_i\right)$$

and the loss function is usually defined by

$$l(x,y)=\begin{cases}0 & \text{if } x=y\\ 1 & \text{if } x\neq y\end{cases}$$

This involves counting the misclassification error if input feature vector x is wrongly classified. The classification (success) rate is then defined as *1-ER(X,Y,Z)*.

However, if the proportion of elements among classes is not well balanced, the classification rate is no more confident. This is generally the case for feature data sets coming from microscopic ground truth images where most of pixels belong to background in a very high proportion (higher than 80%). A classifier that classifies all pixels as background will have a classification rate close to 80% although it performs very badly. To correct this, it is recommended to use a Balanced Error Rate (BER) that computes the average of the error rates on each class. BER is defined by:

$$BER(X,Y,Z)=\frac{1}{|Y|}\sum_{\omega_i\in Y}\left(\frac{1}{|\omega_i|}\sum_{\left(x_j,y_j\right),y_j=\omega_i} l\left(F\left(x_j\right),y_j\right)\right)$$

Other measures to evaluate classification results exist. The most used are based on Precision and Recall measures. They are obtained from an analysis of the classification confusion matrix. Let *TP* denote *True Positives*, *FP False Positives*, and *FN False Negatives* that are defined for each class by:

$$TP\left(\omega_i\right)=\left|\left\{\left(x_i,y_i\right),y_i=\omega_i \text{ and } F\left(x_i\right)=\omega_i\right\}\right|$$

$$FN\left(\omega_i\right)=\left|\left\{\left(x_i,y_i\right),y_i=\omega_i \text{ and } F\left(x_i\right)\neq\omega_i\right\}\right|$$

$$FP\left(\omega_i\right)=\left|\left\{\left(x_i,y_i\right),y_i\neq\omega_i \text{ and } F\left(x_i\right)=\omega_i\right\}\right|$$

From the latter, Precision, Recall and F-measure, can be defined by:

$$\text{Precision}\left(\omega_i\right)=\frac{TP\left(\omega_i\right)}{TP\left(\omega_i\right)+FP\left(\omega_i\right)}$$

$$\text{Recall}\left(\omega_i\right)=\frac{TP\left(\omega_i\right)}{TP\left(\omega_i\right)+FN\left(\omega_i\right)}$$

$$\text{F-measure}\left(\omega_i\right)=\frac{2\text{Precision}\left(\omega_i\right)\text{Recall}\left(\omega_i\right)}{\text{Precision}\left(\omega_i\right)+\text{Recall}\left(\omega_i\right)}$$

For instance, a specific F-measure for the evaluation of microscopic image classification has been proposed in (Meurie, 2003). Figure 3(c)-(d) provides such a F-measure for nuclei and cytoplasm classes. One can see that this measure well reflects how good the classification is with respect to ground truth (the higher the F-measure, the better).

All previous methods are dedicated to pixel classification results evaluation: they do not take into account the spatial information in images. Therefore, for the case of segmentation evaluation, other specific measures have to be taken into account. An excellent review of segmentation evaluation methods can be found in (Chabrier, 2006). However, all these methods are not always suited for evaluating microscopic image segmentation results. A more specific method has been proposed in (Lebrun, 2007) to evaluate in a single measure the segmentation of cells in microscopic images. This cell segmentation quality criterion q_{seg} takes into account the adequacy q_{shape} between the shape of the objects produced by an automatic segmentation I_a and an expert segmentation I_e.

That criterion also takes into account the number of missing objects $n_{missing}$ and the number of artifact objects $n_{artifact}$. Definition of that criterion is, with $\lambda \in [0,1]$:

$$q_{seg} = q_{shape} + \lambda n_{\text{missing}} + (1 - \lambda) n_{\text{artifact}}$$

Constant λ makes it possible to favor a segmentation that limits the number of missing objects as compared to the number of artifact objects. In the case of cell segmentation, it is essential that no cell is lost, even if that forces to keep some artifacts, so a typical value is $\lambda = 0{,}9$. The shape adequacy q_{shape} is defined as following:

$$q_{shape} = \frac{1}{|I_a|} \sum_{I_a(p) \neq I_e(p)} \min\left(d_e\left(p, I_e\right), d_{\max}\right)^2$$

In the latter, $d_e(p,I_e)$ corresponds to the distance between the pixel p and its nearest pixel belonging to the shape edges in expert segmentation I_e. d_{max} value aims at restricting the effect of weighting decrease when pixels are close to expert segmentation boundaries. In Figure 3, values of q_{seg} and q_{shape} are provided for two images obtained from image classification by SVM, and further refined by a watershed. Obtained values efficiently quantify segmentation results (the lower q_{seg} value, the better) as confirmed by visual analysis.

Multiple Classifier Fusion

In this Section, a complete morphological segmentation scheme based on machine learning techniques for marker extraction is designed for the automatic segmentation of bronchial color microscopic images. First, the abilities of different machine learning algorithms are studied for sole pixel classification. Given an image as input, each classifier processes an image by assigning a label to each pixel. Unsupervised classification directly treats each image without exploiting any model inferred from an image database whereas this is the case for supervised classification. Whatever the classifier, its hyper-parameters and feature vector used to represent a color pixel have serious influence on final results. It is therefore essential to choose the best representation and parameters. This is performed in cascade: first, each classifier has its parameters optimized, and second the best pixel color features are determined, both steps with respect to F-measure. Once this is done, a set of different classifiers is obtained, operating on different pixel representations and having different abilities to extract seeds. Table 1 presents, for each of the abovementioned classifiers, the retained color feature vector used to represent pixels so as to obtain the best results in terms of F-measure for the extraction of cytoplasm and nuclei. This F-measure is measured and averaged over all the comparisons with ground truth images. As shown in Table 1, the color representation has high influence on results and classifiers do not respond in the same way to similar feature vectors. Moreover, supervised classifiers (SVM, Bayes, MLP and kNN) tend to provide better pixel classification results in terms of F-measure. Figure 4 presents classification results for a color microscopic image with classifiers of Table 1.

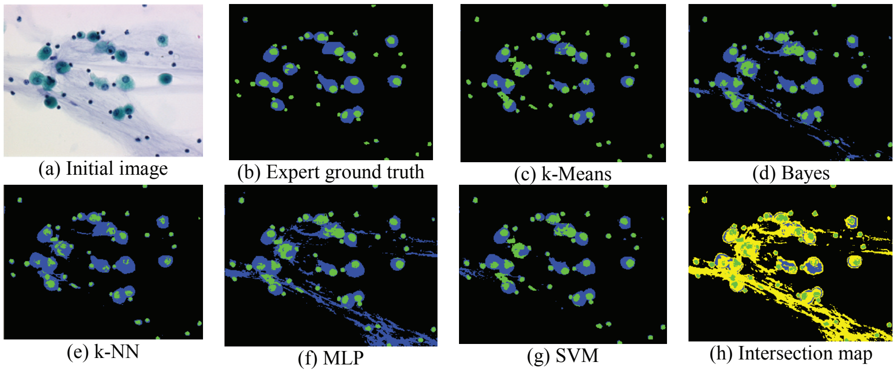
Previous results have shown that pixel classification is a good candidate for marker extraction. However, it remains difficult to choose only one single classifier for extracting markers since results obtained by some of them are very close. Despite this, SVM, Bayes, kNN and k-means can be retained as the most reliable classifiers. Therefore, this first step of classifier evaluation was essential to retain the best machine-learning candidates for marker extraction regarding the problem under consideration.

Since it is difficult to choose a single classifier to perform a marker extraction task, an alternative lies in combining outputs of several classifiers. Figure 4(h) shows an intersection map of several pixel classifications obtained from different classifiers. In this image, pixels colored in yellow present cases where at least one classifier predicted a different class from the other classi-

Table 1. Pixel classification results with different machine learning algorithms and feature vectors. The first four rows concern unsupervised classifiers and the last four rows supervised classifiers. Best F-measures are bold faced.

Classifier	Color feature vector	100*F(cytoplasm)	100*F(nuclei)
k-means	YCh_1Ch_2	69,5	74,4
FLDA	*RGB*	50,8	72,4
FLDA	$I_1I_2I_3$	57,3	71,9
FLDA	*HSL*	59,9	69,8
SVM	YCh_1Ch_2	77,4	74,2
Bayes	YCh_1Ch_2	72,2	**74,6**
MLP	YC_bC_r	56,9	73
kNN	*HSL*	**79,9**	70

Figure 4. Several pixel classifications ((c) to (g)) of an original image (a) and the intersection map of all these classifications (h) illustrating the way they agree altogether.



(a) Initial image (b) Expert ground truth (c) k-Means (d) Bayes

(e) k-NN (f) MLP (g) SVM (h) Intersection map

fiers. To alleviate these incoherencies, a classical way is to fuse the outputs of several classifiers to take a final decision. This comes to do multiple classifier fusion. Given a set of *m* classifiers, a combination rule *g* can be used to fuse results. This combination rule is used to estimate class membership estimates by (Kuncheva, 2004):

$$p(\omega = \omega_i \mid x) = g\left(\left\{\phi_j(\omega_i) f_j(x, \omega_i), \forall j \in [1, m[\right\}\right)$$

where $f_j(x,\omega_i)$ and $\phi_j(\omega_i)$ respectively denote class membership estimates by classifier *j* for class ω_i, and the confidence in the classification performed by classifier *j* for class ω_i. The confidence is a value assessing the reliability of a classifier in its recognition of a given class. Typical values for confidence are provided by evaluation measures. Table 2 presents results that show the interest in multiple classifier fusion with respect to using single classifiers. The confidence of each classifier is evaluated with F-measures and only three classifiers are combined (SVM, Bayes and k-means). Two combination rules *g* are considered (sum and majority vote). First, multiple classifier combination enables to obtain better final classification whatever the combination rule, sum combination rule outperforming majority vote. Therefore, multiple classifier combination is a simple and

Table 2. F-measures for cytoplasm and nuclei extraction by different combination rules of pixel classifications (k-means, Bayes, SVM) and by classification refined by watershed (i.e. a complete morphological segmentation making use of machine-learning algorithms for extracting markers).

Classification scheme	100*F(cytoplasm)	100*F(nuclei)
Classification by sum rule combination	78,3	74,9
Classification by majority vote rule combination	78,1	74,8
Classification refined by Watershed	**100*F(cytoplasm)**	**100*F(nuclei)**
k-means + Watershed	72,8	76,2
SVM + Watershed	73,2	75,8
Bayes + Watershed	71,1	76,3
Sum rule combination + Watershed	76,5	76,4

efficient method to merge the outputs of several classifiers with close accuracies.

As mentioned earlier, pixel classification is only one step in a machine-learning based morphological segmentation scheme. To assess the quality of marker extraction by machine-learning means, one needs evaluation of the whole segmentation scheme (classification followed by watershed) and not only of the sole pixel classification. These results are presented in Table 2. In this case, one evaluates the final segmentation obtained through the whole scheme of Figure 2. First, whatever the classifier, multiple classifier combination for marker extraction of a watershed performs always better than single classifier marker extraction. Second, spatial refinement by watershed enables increasing of the nuclei detection rate.

Finally, there are few differences between the obtained results. Even if one classifier performs slightly better than another one, this is not of high importance for the next morphological segmentation step, watershed. However, the previous scheme has pointed out which classifier performs the best in average: Support Vector Machines. In next Section, we show one can globally optimize such a single supervised classifier.

Single Classifier Optimization

Working with machine learning algorithms for pixel classification involves taking into account not only the recognition rate of the base inducer but also the processing time needed to perform a single pixel classification (Lebrun, 2008). SVM are powerful classifiers having high generalization abilities, but the decision function build by SVM has a complexity that increases with training set size (Steinwart, 2004). As a consequence, using SVM directly on a huge pixel dataset is not directly tractable to produce fast and efficient pixel classifier (Lebrun, 2008). Therefore, it is essential to perform an efficient model selection of SVM that achieves a trade-off between recognition rate and low complexity of the inducer (the decision function). Such a trade-off can be expressed via a criterion to optimize (Lebrun, 2007) that will be called Decision Function Quality (DFQ) in the sequel. A natural way to reduce the complexity of decision functions produced by SVM is to control the number of support vectors. Since the latter is related to training set size, one can control complexity by modifying training set size through Vector Quantization (Gersho, 1991). As opposed to the approach described in previous Section that operates in cascade, it is more natural to choose for a SVM, in a single optimization process, the values of the SVM hyper-parameters, and the simplification level of the training set, in order to optimize the proposed DFQ criterion. Such an optimization process is usually named model selection. Exhaustive search for model selection being not tractable, meta-heuristic have to be used, e.g. taboo search (Glover, 1999). Such a model selection for pixel classification has been proposed in (Lebrun, 2007) to design SVM decision function of high recognition rates while being parsimonious. To perform pixel classification, three different binary decision functions are induced, each one discriminating one class of pixel against the others. This is called a one-against-all decomposition.

Results obtained with this methodology are shown in Table 3. Recognition rate is determined using a balanced error rate. Results show that training time stays tractable in all cases. Mean classification time per image is also tractable (only few seconds) as compared with no dataset quantization (classification time higher than 1 hour). If we compare this to results presented in previous Section, SVM was probably the best pixel classifier but also the slowest. A careful model selection is therefore essential. As attended, color space has also an impact on recognition rate.

Moreover, one can see that high confidence in class memberships is obtained. Classification result is close to ground truth but with processing times largely lower than for classifiers described in previous Section. Finally, to obtain image segmentation from image classification, a watershed is performed with as markers the classification result of a SVM-based pixel classification after model selection. Table 4 shows benefits of refining image segmentation obtained by pixel classification: better results are always obtained regarding sole pixel classification. Figure 5 shows segmentation results with this segmentation scheme in comparison with expert segmentation. Globally, automatic segmentations have good matchings with expert segmentations.

FUTURE TRENDS

Future works will concern the adaptation of machine-learning based algorithms to the clas-

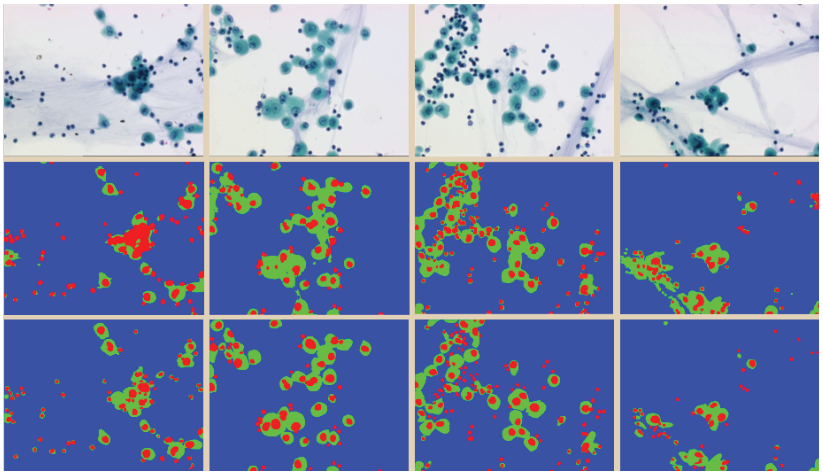
*Table 3. Recognition rate (***1-BER***), total number of support vectors |SV|, training time and mean classification time (in seconds) per image are given for multi-class decision functions produced with nine different color spaces.*

Color space	*1-BER*	*|SV|*	Training time	Mean classification time
RGB	86.55 %	479	2639	10.32
XY_1Z	86.80 %	1364	12017	29.22
L*a*b*	86.74 %	745	3856	16.80
L*u*v*	86.35 %	2680	5761	61.98
LCH_1	85.97 %	1239	6785	27.40
YCh_1Ch_2	**87.09 %**	**303**	**6404**	**6.58**
$I_1I_2I_3$	86.85 %	2589	4760	54.11
HSL	86.02 %	2520	2899	55.52
YC_bC_r	86.67 %	519	2668	11.08
Average	86.56 %	1382	5310	30.34

Table 4. Cell segmentation quality (q_{seg}), shape quality (q_{shape}) and missed artifact trade-off quality ($q' = q_{seg} - q_{shape}$) with 8 microscopic images for pixel classification and pixel classification refined by watershed.

	Pixel classification			**Pixel Classification refined by watershed**		
Image	q_{seg}	q_{shape}	q'	q_{seg}	q_{shape}	q'
0	17.24	1.04	16.20	**4.60**	**0.80**	**3.80**
1	6.11	1.01	5.10	**4.36**	**0.96**	**3.40**
2	11.00	0.90	10.10	**6.26**	**0.66**	**5.60**
3	24.59	2.19	22.40	**4.89**	**1.49**	**3.40**
4	6.34	0.24	6.10	**2.30**	**0.20**	**2.10**
5	6.42	0.52	5.90	**2.45**	**0.45**	**2.00**
6	7.33	0.73	6.60	**4.58**	**0.68**	**3.90**
7	13.99	0.79	13.20	**3.17**	**0.57**	**2.60**
average	11.63	0.93	10.70	**4.08**	**0.73**	**3.35**

Figure 5. Cell microscopic images (first row), segmentations produced by SVM after model selection refined by watershed (middle row) and expert segmentation (last row).



sification of regions. It is much more natural, and presumably more efficient, to work with perceptually meaningful entities obtained from low-level grouping process. This will transform the problem of pixel classification into a problem of region classification. However, even if this can be attractive in terms of complexity reduction, this introduces other problems in terms of region description that have to be studied in depth.

CONCLUSION

Machine Learning algorithms have emerged as powerful techniques to introduce adaptation into the conception of image processing algorithms. For the special case of Mathematical Morphology making use of watershed from markers, far more relevant results can be obtained with markers extracted by pixel classification by machine learning algorithms. In this chapter, we described how to exploit machine learning for morphological segmentation.

ACKNOWLEDGMENT

This work was partially supported under a research grants of the ANR Foundation (ANR-06-MDCA-008-01/FOGRIMMI), FEDER and FNADT funds, and doctoral grants of the Conseil Régional de Basse-Normandie and of the Cœur et Cancer association.

REFERENCES

Arif, M., & Rajpoot, N. M. (2007). Detection of nuclei by unsupervised manifold learning. *Proceedings of Medical Image Understanding and analysis*.

Chabrier, S., Emile, B., Rosenberger, C., & Laurent, H. (2006). Unsupervised performance evaluation of image segmentation. *EURASIP Journal on Applied Signal Processing*, (pp. 1-12).

Cristianini, N., & Shawe-Taylor, J. (2000). *Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press.

Derivaux, S., Lefèvre, S., Wemmert, C., & Korczak, J. J. (2007). On machine learning in watershed segmentation. *IEEE International Workshop on Machine Learning in Signal Processing* (pp. 187-192).

Dreyfus, G. (2005). *Neural Networks, Methodology and Applications*. Springer.

Duda, R., Hart, P., & Stork, D. (2000). *Pattern Classification*, Second Edition.

Fisher, R. A. (1936). The Use of Multiple Measurements in Taxonomic Problems. *Annals of Eugenics*, *7*, 179-188.

Gersho, A., & Gray, R. M. (1991). *Vector Quantization and Signal Compression*. Kluwer Academic edition.

Glover, F., & Lagun, M. (1999). *Tabu search*. Boston, MA: Kluwer Academic Publishers.

Hsu, C.-W., & Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, *13*, 415-425.

Kuncheva, L. (2004). *Combining Pattern Classifiers - Methods and Algorithm*s. Wiley.

Lebrun, G., Charrier, C., Lezoray, O., & Cardot H. (2007). A fast and efficient segmentation scheme for cell microscopic image. *Cellular and Molecular Biology*, *Biomedical Signal and Image Processing*, *53*(2), 51-61.

Lebrun, G., Charrier, C., Lezoray, O., & Cardot, H. (2008) Tabu search model selection for SVM. *International Journal of Neural Systems*, *18*(1), 19-31.

Levner, I., & Zhang, H. (2007). Classification-Driven Watershed Segmentation. *IEEE Transactions on Image Processing*, *16*(5), 1437-1445.

Lezoray, O., & Cardot, H. (2002). Cooperation of color pixel classification schemes and color watershed : a study for microscopical images. *IEEE transactions on Image Processing*, *11*(7), 783-789.

Linde, Y., Buzo, A., & Gray, R. M. (1980). An Algorithm for Vector Quantizer Design. *IEEE Transactions on Communications*, (pp. 702-710).

Lukac, R., & Plataniotis, K. (2006). *Color Image Processing: Methods and Applications*. Boca Raton, FL: CRC Press / Taylor & Francis.

Meurie, C., Lebrun, G., Lezoray, O., & Elmoataz, A. (2003). A supervised segmentation scheme for cancerology color images. *IEEE ISSPIT* (pp. 664-667).

Meurie, C., Lezoray, O., Charrier, C., & Elmoataz, A. (2005). Combination of multiple pixel classifiers for microscopic image segmentation. *International Journal of Robotics and Automation*, *20*(2), 63-69.

Meyer, F. (2001). An Overview of Morphological Segmentation. *IJPRAI, 15*(7), 1089-1118.

Michie, D., Spiegelhalter, D., & Taylor, C. (1994). *Machine learning, neural and statistical classification*.

Platt, J. (1998). Sequential minimal optimization: A fast algorithm for training support vector machines. In *Advances in Kernel Methods - Support Vector Learning*, (pp. 185-208).

Platt, J. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, (pp. 61-74).

Rivest, J.-F., Beucher, S., & Delhomme, J. P. (1992). Marker-controlled segmentation: an application to electrical borehole imaging. *Journal of Electronic Imaging*, *1*(2), 136-142.

Roerdink, J. B. T. M., & Meijster, A. (2000). The watershed transform: definitions, algorithms, and parallellization strategies. *Fundamenta Informaticae*, *41*, 187-228.

Rosenblatt, F. (1958). The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*, *65*(6), 386-408.

Serra, J. (1988). *Image Analysis and Mathematical Morphology: Theoretical Advances*. London: Academic Press.

Soille, P. (2004). *Morphological Image Analysis: Principles and Applications*, Springer-Verlag, Berlin and New York, corrected 2nd printing of the 2nd edition.

Steinwart, I. (2004). Sparseness of support vector machines -some asymptotically sharp bounds. *Proceedings of NIPS* (pp. 169-184).

Vapnik, V.N. (1998). *Statistical Learning Theory*. Wiley.

KEY TERMS

Classification: The process of deriving a mathematical function that can predict the membership of a class based on input data.

Classifier Combination: Classifier combination consists in combining results obtained from a set of classifiers to achieve higher performance than each single classifier.

Ground Truth: A ground-truth database is a database that provides a list of the objects in each image.

Machine Learning: As a broad subfield of artificial intelligence, machine learning is concerned with the design and development of algorithms and techniques that allow computers to "learn".

Mathematical Morphology: Mathematical morphology (MM) is a theoretical model for digital images built upon lattice theory and topology. It is the foundation of morphological image processing, which is based on shift-invariant (translation invariant) operators based principally on Minkowski addition.

Model Selection: Selection of an optimal model to predict outputs from inputs by fitting adjustable parameters.

Support Vector Machines: SVM map input vector to a higher dimensional space where a maximal hyperplane is constructed.

Watershed: Segmentation by watershed designs a family of segmentation methods that consider an image as a topographic relief the flooding of which is simulated.