Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Pattern Recognition Letters 26 (2005) 1059–1068
Pattern Recognition Letters
www.elsevier.com/locate/patrec

Learning-based algorithm selection for image segmentation

Xia Yong [a,b,*], Dagan Feng [a,c], Zhao Rongchun [b], Maria Petrou [d]

[a] *Center for Multimedia Signal Processing, Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong*
[b] *School of Computer, Northwestern Polytechnical University, 710072 Xi'an, China*
[c] *School of Information Technologies, F09, University of Sydney, Sydney, NSW 2006, Australia*
[d] *School of Electronics, Computing, and Mathematics, University of Surrey, Guildford, GU2 5XH, UK*

Received 23 April 2004
Available online 11 November 2004

Abstract

Segmentation of nontrivial images is one of the most important tasks in image processing. It is easy for human being, but extremely difficult for computers. With the purpose of finding optimal segmentation algorithm for every image through learning from human experience, this paper investigates the manual segmentation process and thus presents a performance prediction based algorithm selection model to bridge the knowledge gap between images and segmentation algorithms. Derived from that model, a framework of learning-based algorithm selection system is proposed to automatically segment all images in a large database. A simulation system is designed to select the optimal segmentation algorithm from four candidates for synthetic images. The system is tested on 9000 images by comparing with the manual algorithm selection. The best algorithms are selected for 85% of the cases. If we also regard the second best algorithm as acceptable, more than 97% of images can be properly segmented. The satisfied result demonstrated that this study has provided a promising approach to achieve automated image segmentation.

Keywords: Image segmentation; Segmentation evaluation; Machine learning; Support vector machine

1. Introduction

Image segmentation is basically a problem of psycho-physical perception, and therefore, not susceptible to any purely analytical solution (Fu and Mui, 1981; Pal and Pal, 1993). In order to get a satisfying result, any segmentation algorithm, no matter what mathematical model it is based on,

* Corresponding author. Address: School of Computer, Northwestern Polytechnical University, Department of Computer Information and Engineering, P.O. Box 756, 710072 Xi'an, Shannxi, China. Tel.: +86 29 8849 4848/5454; fax: +86 29 8849 4000.
E-mail address: yxia@it.usyd.edu.au (X. Yong).

0167-8655/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2004.09.049

must be supplemented by heuristics which involve both semantic information and a priori knowledge about the images under consideration. As a result, though hundreds of segmentation algorithms have been presented in the literature during the past 40 years (Fu and Mui, 1981; Haralick and Shapiro, 1985; Sahoo et al., 1988; Pal and Pal, 1993), there is no a single general algorithm which may be considered good for all images, nor are all algorithms equally good for a particular image (Pal and Pal, 1993). Those algorithms all suffer from sensitivity to the properties of images, such as noise level, illumination condition, and the target size. When used to segment a large variety of images, none of them can succeed on all images, especially in case of greatly variant image characteristics. Up to now, a variety of attempts have already been made by researchers in recent years.

The most popular solution is to enhance the generalization ability and robustness of segmentation algorithms so that they can be successfully applied to more images (Francis et al., 1998; Yang and Yan, 2000; Gevers, 2002; Zhang and Desai, 2001). Although this strategy is widely used, and to some extent succeeds, we do not expect there is any algorithm which can produce equally good results for all images. Even if such a delicately designed all-purpose method existed, it would have been less efficient than the methods specifically designed for different conditions.

In many machine vision systems, rule-based integration methods are well-accepted approaches for image segmentation (Nazif and Levine, 1984; Li et al., 1988; Mussa et al., 1992; Zhang et al., 2002). That strategy successively generates more specific interpretations of data, and the generation is controlled by rules which encode the domain knowledge and the user's experience. A major drawback of rule-based systems is that a correct and accurate model of the application is needed to convert the knowledge into rules and a complete conversion is usually hard to achieve.

With the development of machine learning techniques, more and more learning-based systems have been proposed. Perner (1999) employs Case-Base Reasoning (CBR) to get segmentation parameters from training samples. Reynolds and Rolnick (1995) use Cultural Algorithms to learn the parameters for a neighborhood-based segmentation algorithm from the results of a region-growing method. Peng and Bhanu (1998) acquired the parameters for their multilevel close-loop system by using "delayed" reinforcement learning. Despite the fact that the parameters obtained from learning can adapt to different images, the improvement is limited by the essential limitation of the original algorithm.

As a matter of fact, using different algorithms to segment different images seems to be the most effective and straightforward solution. However, automated selection of an optimal algorithm according to image characteristics is easier said than done. In most practical image segmentation systems, the intervention of a human operator is often needed to choose the algorithm to be used (Olabarriaga and Smeulders, 2001; Matsuyama, 1989; Udupa et al., 1997). Zhang and Luo (2000) have attempted to achieve automated segmentation algorithm selection by using the heuristic knowledge and feedback of segmentation evalua-

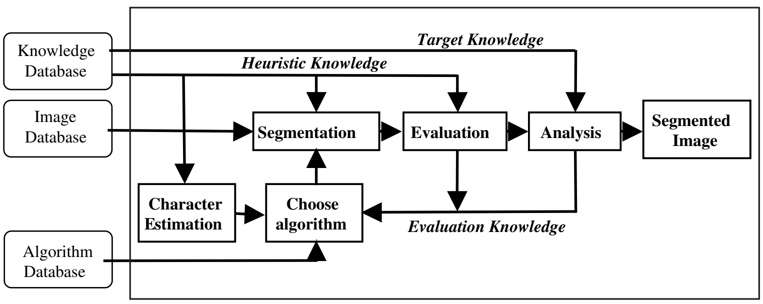

Fig. 1. The framework of performance evaluation based segmentation algorithm selection systems.

tion. In that system, as shown in Fig. 1, the iteration of algorithm selection, image segmentation and segmentation evaluation will not stop until a satisfying result is obtained. Nevertheless, up to now, there is no universally accepted method for either automated segmentation evaluation or human knowledge formulation. Therefore, the system can hardly be applied to real applications. Moreover, the trial-and-error method is computationally inefficient and not acceptable for real-time tasks.

Based on the idea of learning from the segmentation experience provided by users and selecting the optimal algorithm for every image by imitating experienced users, this paper proposes the framework of learning-based image segmentation system, which is derived from a performance prediction based algorithm selection model. A simulation system is constructed to select the optimal segmentation algorithm from four candidates for synthetic images. 9000 testing images are used to assess the performance of the system.

2. Performance prediction based algorithm selection model

It is interesting that image segmentation is quite easy for experienced users, but extremely difficult for computers. The utility of existing segmentation algorithms is badly limited by their narrow, highly specific image orientation. For most algorithms, it is unknown what kinds of properties are essential for the images, on which the best performance can be achieved. Furthermore, there are no universally accepted methods to measure all the properties of images quantitatively and accurately. But how users achieve image segmentation? Surely, they must have some knowledge on the image under consideration, the image analysis task, and some segmentation algorithms as well. According to their knowledge and experience, they can predict which algorithm may bring better result than others, and thus make the selection. This manual process can be depicted by the following algorithm selection model, as shown in Fig. 2.

As illustrated in Fig. 2, the most critical step in the whole algorithm selection process is predicting the performances of different algorithms, which is also the basis of decision-making. The prediction is based on the features of the image and supervised by the knowledge and experience of the users. If computer can predict the performance and make the decision by imitating users, we will be able to select optimal algorithms for different images automatically, and thus achieves better segmentation results. To this end, we attempt to investigate the knowledge gap between images and algorithms. First of all, we need to introduce two assumptions.

Assumption 1. There is a strong correlation between image characteristics, such as contrast, noise, illumination and target, and the performance of segmentation algorithms which are applied to the image.

Assumption 2. Applying the same segmentation algorithm to the images which share the same characteristics can produce results with the same performance.

Assumption 1 comes from the analysis of the algorithm selection model and is supported by our successful experience of algorithm performance

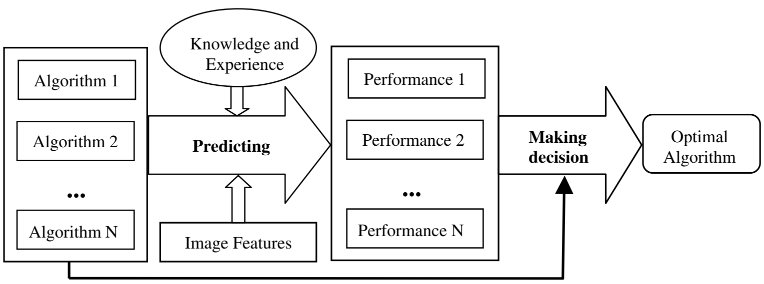

Fig. 2. Performance prediction based algorithm selection model.

prediction. Assumption 2 is the theoretical foundation of the automated prediction of segmentation performances. In fact, it is satisfied in most cases.

Based on those two assumptions, we are able to formally define the performance prediction based algorithm selection model mathematically. Let $X = (x_1,x_2,\ldots,x_m)$ denote the image features and r_i denote the performance rank of the ith algorithm on that image. According to Assumption 1, there exists a map from image features X to algorithm performance rank r_i.

$$f_i' : X \mapsto r_i \tag{1}$$

Assumption 2 means the performance rank of a given segmentation algorithm on an image is accurately a function of the image features, and the above map may be refined as

$$r_i = f_i(X) \tag{2}$$

For a set of different algorithms, the scalar function (2) could be substituted by a multi-variable vector function

$$R = F(X) \tag{3}$$

where $R = (r_1,r_2,\ldots,r_n)$ and n is the number of algorithms. The index of the optimal algorithm is

$$i = \mathrm{Arg}\,\underset{0<i<n}{\mathrm{Max}}\, r_i \tag{4}$$

3. Performance prediction based algorithm selection

The performance prediction based algorithm selection model gives us a possible explanation of how people accomplish image segmentation by using different algorithms adaptively. As indicated by this model, algorithm selection is based on the predicted performance, which is determined by the prediction function, as shown by Eq. (3). The prediction function, which contains the heuristic knowledge and human experience needed for algorithm selection, can be approximated by using machine learning techniques to learn from human experience. Based on this idea, a framework of a learning-based optimal algorithm selection system is proposed to achieve automated image segmentation.

As shown in Fig. 3, the system consists of three major modules: the performance predictor, the performance evaluator, and the feature extractor. The performance predictor is designed to estimate the prediction function by learning from the segmentation experiences gained in training stage. Various machine learning techniques can be employed to fulfill this task. It provides to users a convenient tool to automatically acquire, store and utilize human knowledge in an implicit way. The performance evaluator is used to rank the segmentation results of the sample images during training. Generally, automated segmentation evaluation is not a well-solved problem. The proposed system adopts the off-line learning which enables interactive segmentation evaluation. Users can subjectively rank the performances of all candidate algorithms with the help of some objective criteria so that the evaluation, which is more consistent with human vision system (HVS), can be obtained. The feature extractor must be powerful enough for the extracted features to provide sufficient information to direct the selection of the optimal algorithm. It is hard to design a general feature extractor. Fortunately, it is easy for non-texture images. Taking into account of the fact that many thresholding algorithms, which only use the gray level histogram, usually can get acceptable segmentation results, we believe that histograms are fairly good features for most of those images.

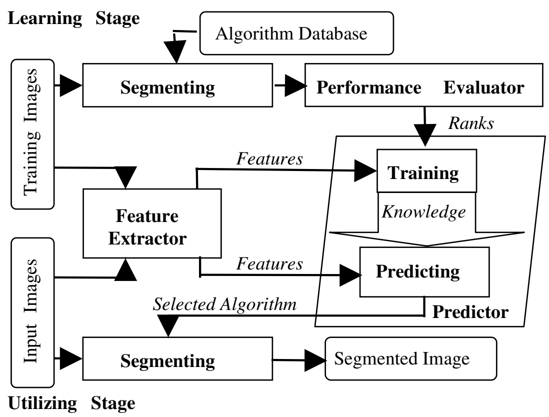

Fig. 3. The framework of learning-based segmentation algorithm selection system.

The operation of the proposed system can be divided into two stages: off-line learning and online using. In the learning stage, every sample image is segmented by all candidate algorithms, and the obtained results are ranked interactively. Then, both the performance ranks and the features extracted from the image are used to train the predictor and estimate the prediction function. After training, the knowledge needed for algorithm selection has already been stored in the predictor. When the system is put to use, features of each input image are extracted first. Then, based on those features, the performance ranks of all algorithms on the image are computed by using the prediction function. Finally, as indicated by Eq. (4), the algorithm which has the best performance is selected as the optimal one and applied to that image.

4. Simulated system

In order to assess the feasibility of the proposed learning-based optimal algorithm selection system, we constructed a simulated system to select optimal segmentation algorithms from four candidates for synthetic images.

In this study, the image database we used consists of 1000 synthetic images, which are denoted by $T1, T2, \ldots, T1000$. The design and generation of these images is similar to that given by Zhang and Gerbrands (1992). The object in every image is a square, a circle, an ellipse, or a combination of these primitives. The size of the object is ranging from 0.5% to 60% of the entire image. The gray lever of the objects is 144, whereas that of background is 112. To make the image more realistic, a 3×3 mean filter is applied to create a transition region between objects and background. With the aim of simulating the deformation caused by noise, a two dimensional discrete Gaussian grid with mean zero is added to every image. The standard deviation of the Gaussian grid is ranging from 4 to 36. As defined by Kitchen and Rosenfeld (1981), the Signal-to-Noise Ratio (SNR) is

$$\mathrm{SNR} = \left(\frac{\Delta g}{\sigma}\right)^2 \tag{5}$$

where Δg is the difference of gray level between objects and background in the noise-free image, and σ is the standard deviation of the noise. The SNR levels of these synthetic images are ranging from 64 to 0.79, which cover the range of many applications (Kitchen and Rosenfeld, 1981). The noise level, target size and target shape are all randomly determined for every sample image to simulate the randomness of images. 12 examples from the image database are shown in Fig. 4.

Generally, the algorithm candidates must be carefully determined so that every image may be properly segmented by at least one of them. This requires a diverse algorithm set, where each algorithm should be able to properly segment the images with certain properties. According to the generation of the image database, the target and background in all synthetic images can be approximately differentiated by gray level. Therefore, we choose four thresholding methods as algorithm candidates, which are denoted by $A1$, $A2$, $A3$, and $A4$, respectively, and detailed in Table 1. Algorithm 1 is the widely used Otsu's thresholding method. However, its performance declines with decreasing target size (Kittler and Illingworth,

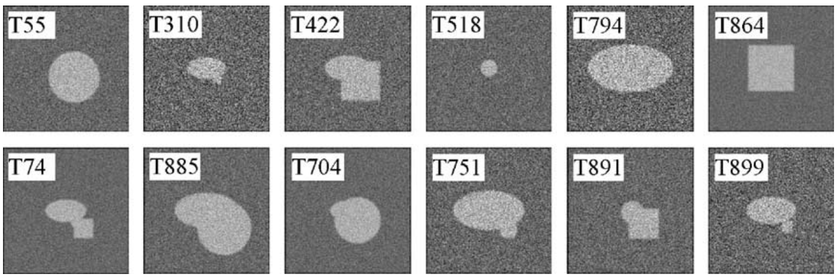

Fig. 4. Twelve examples from the image database.

Table 1
Candidate image segmentation algorithms

Index	Description
$A1$	Otsu's thresholding algorithm (Otsu, 1979)
$A2$	Thresholding based on enhanced histogram, which only count the edge points (Weszka et al., 1974)
$A3$	Thresholding based on the gray-level and average gray-level scatterplot and Region Growing (Xia and Zhao, 2002)
$A4$	Wiener denoising and Kapur's maximum entropy thresholding (Kapur et al., 1985)

1985). In Algorithm 2, only those edge points, where the gradient value is larger than some threshold, are used to estimate the threshold by employing Otsu algorithm. By this means, the impact of the target size is almost eliminated. The other two algorithms take account of the deformation caused by noise, and thus achieve better robustness. In Algorithm 3, an approximate threshold is calculated based on the gray-level and average gray-level scatterplot. Then, all pixels are divided into three classes: target, background and edge. Finally, edge pixels are further classified by a region growing technique. In Algorithm 4, before Kapur's maximum entropy criterion, which is not as sensitive to the target size as the Ostu algorithm, is adopted to seek the optimal threshold, Wiener filtering is used to denoise the original image. Those four algorithms are enough for this simulation experiment, since every image in the database can get at least one acceptable segmentation result by applying them to it.

Sample images are divided into two sets: the training set and the testing set. In the learning stage, the performances of all segmentation algorithm candidates on every training sample are evaluated by users and ranked as {Best, Good, Bad, Worst}, respectively. If it is hard to tell which result is better, the ratio of the number of misclassified pixels to the area of the target, denoted by ε, is calculated to facilitate the comparison. If two results share the same ε, the one with less computational cost will be ranked higher. After evaluation, both the performance ranks of algorithms and the histogram of the image are gathered to train the predictor.

The performance predictor must learn from small training samples in a very high dimensional feature space. Because users are usually reluctant to provide a large number of training examples in practical terms, and the histogram, which is used as the image feature, has a dimension of 256. To solve this problem, the Principal Components Analysis (PCA) (Jackson, 1991) is adopted to reduce the feature dimensionality, and a series of Support Vector Machines (SVM) (Vapnic, 1995) are employed to learn the heuristics.

The purpose of PCA is to reduce the complexity of the multivariate data (p dimension) into the principal components space and then choose the first q principal component ($q < p$) that explain most of the variation in the original variables. In our experiment, only 6 principal components are used, and histograms are projected into the 6-dimensional feature space, where the reconstruction accuracy is about 96%.

A SVM is a universal learning machine whose decision surface is parameterized by a set of support vectors, and by a set of corresponding weights. Theoretically, it provides much better generalization and tends to work better on small sample sets. Due to those desired characteristics, SVM is used to approximate Eq. (3).

For the convenient of computation, the performance ranks, which originally belong to {Best, Good, Bad, Worst}, are mapped into {1.5, 0.5, −0.5, −1.5}. The better performance an algorithm achieves, the higher rank will be given. The compressed features are normalized in the following way

$$f'_{ij} = \frac{f_{ij} - m_i}{\sigma_i}, \quad i = 1, 2, \ldots, 6, \; j = 1, 2, \ldots, M \tag{6}$$

where m_i and σ_i are the mean and standard deviation of training features in the ith dimension. For every algorithm, all the normalized features and ranks are used to construct a SVM. As given by Eq. (11), we choose the kernel function to be the Exponential Radial Basis Function (ERBF), which is reported to perform better than the other three typical kernel functions, namely polynomial, Gaussian radial basis function, and multilayer perception (Guo and Li, 2003).

$$K(x, x_i) = \exp\left(-\frac{\|x - x_i\|_1}{2\sigma^2}\right) \tag{7}$$

where, $\| \, \|_1$ denotes the vector 1-norm. After all SVMs are generated, the knowledge and experience provided by users are stored as a set of weights.

5. Experimental results

After training, the system can be applied to the segmentation of testing images. For every under-segmented image, the gray level histogram is calculated and projected into the feature space. Then the obtained features are normalized by Eq. (6) and input into the SVMs. The outputs of the SVMs are the predicted performance ranks of all algorithm candidates. Finally, the optimal algorithm determined by Eq. (4) will be applied to the image.

To demonstrate the effectiveness of the proposed system, the performance of the simulated system on two samples ($T450$ and $T569$) are evaluated by comparing it with the ground truth, which is acquired by manually ranking the segmentation results according as ε, the ratio of the number of misclassified pixels to the area of the target. The original images and their segmentations by applying all algorithm candidates are shown in Fig. 5. Both the predicted performance ranks of all algorithm candidates and the corresponding ground truth are listed in Table 2. For test sample $T450$, the algorithm $A1$ is ranked as "Best" because the minimum ε is reported. Meanwhile, the predicted performance rank of algorithm $A1$ is much higher than that of others. That means the selection of the optimal algorithm conducted by the simulated system is definitely correct. The same conclusion can be drawn for testing image $T569$. In this experiment, the system is trained by sample set $\{T1, T2, \ldots, T100\}$.

To get a more reliable result, the 10-folder validation scheme is adopted to further evaluate the performance of the simulated system. All sample images are divided into 10 groups, each has 100

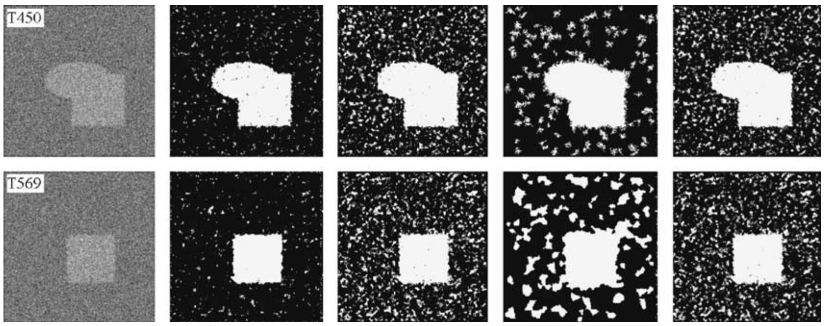

Fig. 5. 2 testing samples (T450 and T569) and their segmentation by applying (the 2nd column) $A1$, (the 3rd column) $A2$, (the 4th column) $A3$, and (the 5th column) $A4$.

Table 2
Experimental results of 2 testing samples (T280 and T450): Ground Truth and the Predicted Performance Ranks

Images	Ground truth				Predicted performance ranks				Results evaluation
	$A1$	$A2$	$A3$	$A4$	$A1$	$A2$	$A3$	$A4$	
$T450$	$\varepsilon = 0.13$ Best	$\varepsilon = 0.68$ Worst	$\varepsilon = 0.44$ Bad	$\varepsilon = 0.22$ Good	$r_1 = 0.66$ Best	$r_2 = -0.28$ Bad	$r_3 = -0.56$ Worst	$r_4 = 0.18$ Good	Right
$T569$	$\varepsilon = 0.28$ Good	$\varepsilon = 1.45$ Bad	$\varepsilon = 1.90$ Worst	$\varepsilon = 0.23$ Best	$r_1 = 0.25$ Good	$r_2 = -0.24$ Bad	$r_3 = -0.62$ Worst	$r_4 = 0.61$ Best	Right

Table 3
The experimental results of the four algorithms on the test sets

Images	Ground truth				Predicted performance ranks				Correct selection	
	Best	Good	Bad	Worst	Best	Good	Bad	Worst	Number	Ratio (%)
$A1$	**5175**	1917	1791	117	**5337**	1999	1560	104	4720	91.21
$A2$	**2196**	3276	2988	540	**2203**	3161	3109	527	1834	83.52
$A3$	**261**	1602	3051	4086	**232**	1545	3004	4219	122	46.74
$A4$	**1368**	2205	1170	4257	**1228**	2295	1327	4150	1016	74.27

Table 4
Results of algorithm selection by using SVM as predictor

Predictor	Best	Good	Bad	Worst
SVM	85.47	11.98	2.32	0.23

images. Every time, only one group of images is used for training and the other 900 images are used for testing. After 10 iterations, all images have already been used for training and 9000 images for testing. The experimental results of the four algorithms on the test sets used for evaluating the proposed method are detailed in Table 3, and the algorithm selection results of the proposed system are shown in Table 4.

Besides the number of correct selection, Table 3 also gives the number of testing samples, on which each algorithm was ranked as "Best", "Good", "Bad" and "Worst", respectively. For the convenience of comparison, both the predicted performance ranks and the ground truth are shown in the above table. For example, it is clear that the algorithm $A1$ is ranked as "Best" by users on 5175 testing samples, and it is predicted as "Best" by the system on 5337 samples, among which the optimal algorithms are correctly selected for 4720 cases and the correct ratio is about 91.21%.

After learning the heuristics knowledge provided by users, the algorithm selection system works fairly well. As shown in Table 4, for 85.47% of testing images, the selected algorithms are actually the optimal algorithms. And for another 11.98% of the cases, the results are the second best algorithms. If we regard the second best choice as acceptable algorithm, more than 97.45% of testing images can be appropriately and automatically segmented by the simulation system.

6. Discussion

The key issues in addressing the proposed algorithm selection system as a supervised learning problem are the feature extractor and the performance predictor. In order to demonstrate the preferable of the simulation system depicted in the previous section, comparative experiments have been carried out by adopting another group of features and three other learning techniques.

The new features consist of four statistics of the image, namely mean value, standard deviation, skewness, and kurtosis. To reduce the computation cost, the original image are sampled down to 32×32 and regarded as a one dimensional vector before the statistics are calculated. Due to its low dimensionality, those features are, without being compressed by the PCA, directly normalized by Eq. (6) and used for training and testing the system.

The learning techniques used in the comparative experiments are, respectively, Back-Propagation Neural Network (BPNN) (Rumelhart et al., 1986), Case-based Learning (CBL) (Mitchell, 1997), and Subspace Support Vector Machine (SSVM) (Zhang et al., 2002). For BPNN, a 3-layer feed-forward network with 6 inputs, 16 hidden nodes and 4 outputs is used. A Tan-Sigmoidal activation function is chosen for all hidden nodes and outputs. The Levenberg-Marquardt algorithm (More, 1977) is adopted to approach second-order training speed. The CBL algorithm we use here is the distance weighted k-nearest neighbor algorithm (MacLeod et al., 1987). Adopting the idea of SSVM, we use a Self-Organization Map (SOM) (Kohonen, 1995) to cluster all training samples, and thus divide the sample space into 4 sub-spaces. The SVM-based learning is performed in every subspace individually.

Table 5
Comparative experimental results of algorithm selection by using different features and learning techniques

Predictor	Features: histogram				Features: statistics			
	Best	Good	Bad	Worst	Best	Good	Bad	Worst
BPNN	74.05	17.98	7.00	0.97	68.04	22.64	8.07	1.24
CBL	85.33	10.88	3.62	0.17	80.04	14.07	5.70	0.19
SSVM	85.18	12.32	2.31	0.19	80.16	16.51	3.19	0.14
SVM	85.47	11.98	2.32	0.23	81.01	16.03	2.84	0.11

Table 5 gives the comparative experimental results of algorithm selection by using different features and learning techniques. Due to the lack of a theory to determine the structure of the network and the stop condition of training, the generalization of BPNN sometimes suffers from over-fitting. Though widely used, it is not suitable for this learning problem, and the poorest performance has been acquired. Despite the fact that it yields fairly good results, the CBL-based predictor is not a good choice either. As a lazy learning method, CBL does not exploit iterative training, and learning will not be performed until the system is used. Therefore, it is computationally inefficient, especially in the case of a large training set and a large algorithm set. Though reported superior to SVM in separating two twisted data sets, SSVM does not achieve better performance than, though very similar to, SVM in this experiment. The reason may lie in the fact that training samples are far from sufficient for some subspaces. The SVM-based system gives the best performance on both groups of features. On the other hand, the histogram has better ability of characterizing images than those four statistics, as more accurate algorithm selections were achieved by using histograms as features, no matter what learning method is used. It is clearly revealed by the comparative experiments that the simulation system is preferable to other approaches.

7. Conclusions

This paper has proposed a learning-based algorithm selection model and a framework of algorithm selection system for automatic image segmentation. A simulated system has also been constructed to select the optimal segmentation algorithm from four candidates for synthetic images. The system was tested on 9000 images by comparing with manual selection. The results show that over 85% of images, which share great variance in characteristics, can be successfully and automatically segmented.

Using the proposed system, users only need to randomly choose some images as training samples, apply all candidate algorithms to them and rank the segmentation results. Then the performance predictor will try to learn and use the heuristic knowledge provided by the user. After training, no manual intervention is needed any more and the selection will be made in real time. Besides image segmentation, the proposed system may also be applied to other image processing problems.

Further work will be focused on more powerful feature extractors for texture images. In addition, some objective performance evaluation criteria, which have a strong correlation with the judgment of HVS, should also be investigated to free users from the tedious training stage.

Acknowledgement

This research is partially supported by the HK-RGC, ARC, the NSFC under Grant No. 60141002, and the Nation Defense "Tenth Fives" Scientific Research Projects under Grant No. 413160103.

References

Francis, H.Y., Chan, F.K., Lam, H.Z., 1998. Adaptive thresholding by variational method. IEEE Trans. Image Process. 7 (3), 468–473.

Fu, K.S., Mui, J.K., 1981. A survey of image segmentation. Pattern Recognition 13, 3–16.

Gevers, T., 2002. Adaptive image segmentation by combining photometric invariant region and edge information. IEEE Trans. Pattern Anal. Machine Intell. 24 (6), 848–852.

Guo, G., Li, S.Z., 2003. Content-based audio classification and retrieval by support vector machines. IEEE Trans. Neural Networks 14 (1), 209–215.

Haralick, R.M., Shapiro, L.G., 1985. Image segmentation techniques. Comput. Vis., Graph., Image Process. 29, 100–132.

Jackson, J.E., 1991. A User's Guide to Principal Components. Wiley Series on Probability and Statistics. John Wiley and Sons, Chennai.

Kapur, J.N., Sahoo, P.K., Wong, A.K.C., 1985. A new method for gray level picture thresholding using the entropy of histogram. Comput. Vis., Graph., Image Process. 29, 273–285.

Kitchen, L., Rosenfeld, A., 1981. Edge evaluation using local edge coherence. IEEE Trans. Syst., Man, Cybern. 11, 597–605.

Kittler, J., Illingworth, J., 1985. On threshold selection using clustering criteria. IEEE Trans. Syst., Man, Cybern. 15, 652–655.

Kohonen, T., 1995. Self-Organization Maps. Springer-Verlag, Berlin Heidelberg.

Li, X., Wang, K., Li, Z., 1988. A rule based image segmentation system (RBISS). In: Proc. IEEE International Conference on Systems, Man, and Cybernetics 1988 (SMC), vol. 1, Beijing, China, pp. 652–653.

MacLeod, J., Luk, A., Titterington, D., 1987. A re-examination of the distance-weighted *k*-nearest-neighbor classification rule. IEEE Trans. Syst., Man, Cybern. 17, 689–696.

Matsuyama, T., 1989. Expert systems for image processing: knowledge-based compositions of image analysis processes. Comput. Vis., Graph., Image Process. 48, 22–49.

Mitchell, T.M., 1997. Machine Learning. McGraw-Hill, New York.

More, J.J., 1977. The Levenberg–Marquardt algorithm: Implementation and theory. In Numerical Analysis. In: Watson, G.A. (Ed.), Lecture Notes in Mathematics 630. Springer-Verlag, New York.

Mussa, A.W., Marshall, S., Chapman, R., 1992. An object-oriented rule-based image segmentation system for underwater autonomous vehicles. In: Proc. International Conference on Image Processing and its Applications, Netherlands, Holland, pp. 167–171.

Nazif, A.M., Levine, M.D., 1984. Low level image segmentation: an expert system. IEEE Trans. Pattern Anal. Machine Intell. 6 (5), 555–577.

Olabarriaga, S.D., Smeulders, A.W.M., 2001. Interaction in the segmentation of medical images: A survey. Medical Image Analysis 5, 127–142.

Otsu, N., 1979. A threshold selection method from gray level histograms. IEEE Trans. Syst., Man, Cybern. 9, 62–66.

Pal, N.R., Pal, S.K., 1993. A review on image segmentation techniques. Pattern Recognition 26, 1227–1249.

Peng, J., Bhanu, B., 1998. Closed-loop object recognition using reinforcement learning. IEEE Trans. Pattern Anal. Machine Intell. 20 (2), 139–154.

Perner, P., 1999. An architecture for a CBR image segmentation system. Engineering Applications of Artificial Intelligence 12 (6), 749–759.

Reynolds, R.G., Rolnick, S.R., 1995. Learning the parameters for a gradient-based approach to image segmentation from the results of a region growing approach using cultural algorithms. In: Proc. IEEE International Conference on Evolutionary Computation, vol. 2, Perth, Australia, pp. 819–824.

Rumelhart, D.E., Hinton, G.E., Williams, R.J., 1986. Learning internal representations by error propagation. In: Rumelhart, D.E., MeClelland, J.L. (Eds.), Parallel Distributed Processing: Explorations in the microstructure of Cognition, Vol. 1, MIT Press, Cambridge.

Sahoo, P.K., Soltani, S., Wong, A.K.C., Chen, Y.C., 1988. A survey of thresholding techniques. Comput. Vis., Graph., Image Process. 41, 232–260.

Udupa, J.K., Wei, L., Samarasekera, S., Miki, Y., van Buchem, R.I., Grossman, R.I., 1997. Multiple sclerosis lesion quantification using fuzzy-connectedness principles. IEEE Trans. Medical Imaging 16 (5), 598–609.

Vapnic, V., 1995. The Nature of Statistical Learning Theory. Springer, New York.

Weszka, J.S., Nagel, R.N., Rosenfeld, A., 1974. A threshold selection technique. IEEE Trans. comput. 23, 1322–1326.

Xia, Y., Zhao, R.C., 2002. A new thresholding algorithm using spacial information. Chinese Journal of Stereology and Image Analysis 7 (4), 235–239.

Yang, Y., Yan, H., 2000. An adaptive logical method for binarization of degraded document images. Pattern Recognition 33, 787–807.

Zhang, M., Hall, L.O., Goldgof, D.B., 2002. A generic knowledge-guided Image segmentation and labeling system using fuzzy clustering algorithms. IEEE Trans. Syst., Man, Cybern. 32 (5), 571–582.

Zhang, X., Desai, M.D., 2001. Segmentation of bright targets using wavelets and adaptive thresholding. IEEE Trans. Image Process. 10 (7), 1020–1030.

Zhang, Y.J., Luo, H.T., 2000. Optimal selection of segmentation algorithms based on performance evaluation. Optical Engineering 39 (6), 1450–1456.

Zhang, Y.J., Gerbrands, J.J., 1992. On the design of test image for segmentation evaluation. In: Proc. EUROSCO, vol. 1, pp. 551–554.

Zhang, Y.N., Zhao, R.C., Leung, Y., 2002. An efficient target recognition method for large scale data. Acta Electronica Sinica 30 (10), 1533–1535.