International Journal of Electrical, Electronic and Communication Sciences
ISSN: 2517-9438
Vol:4, No:5, 2010

One-Class Support Vector Machines for Aerial Images Segmentation

Chih-Hung Wu, *Member, IEEE,* Chih-Chin Lai, *Member, IEEE,* Chun-Yen Chen, and Yan-He Chen

Abstract—Interpretation of aerial images is an important task in various applications. Image segmentation can be viewed as the essential step for extracting information from aerial images. Among many developed segmentation methods, the technique of clustering has been extensively investigated and used. However, determining the number of clusters in an image is inherently a difficult problem, especially when a priori information on the aerial image is unavailable. This study proposes a support vector machine approach for clustering aerial images. Three cluster validity indices, distance-based index, Davies-Bouldin index, and Xie-Beni index, are utilized as quantitative measures of the quality of clustering results. Comparisons on the effectiveness of these indices and various parameters settings on the proposed methods are conducted. Experimental results are provided to illustrate the feasibility of the proposed approach.

Keywords—Aerial imaging, image segmentation, machine learning, support vector machine, cluster validity index

I. Introduction

Aerial imaging is one of the most common and versatile ways of obtaining information from the Earth surface. Interpretation and annotation of aerial images is an important task in various applications. In order to successfully extract and understand interesting objects from aerial images, an automatical extraction procedure is essential [1], [2], [3]. Image segmentation is a basic but important technique, which partitions an image into physically meaningful regions and helps advanced analysis [4], [5]. Among many developed segmentation methods, the techniques of clustering have been extensively investigated and used.

The aim of cluster analysis is to partition a given set of data or objects into clusters. That data or objects in the same cluster have some properties in common and the ones across clusters are discriminated by other properties. Many clustering methods were proposed in the literatures [6], [7] and have been effectively used in many applications such as image processing, pattern recognition, information retrieval, data mining, etc.

Whatever the clustering method used, two key issues are critical. The first one is the efficiency of the clustering algorithm that can partition objects or data efficiently and reasonably. The second one is a reasonable estimation on the number of clusters to be used in the clustering algorithm. This estimation is an intrinsic part of the clustering procedure and a false estimate makes the clustering meaningless [8], [9]. Usually, this number is subjectively determined by the users. However, it is difficult for users to have exact ideas about the number of clusters in real-life analysis on aerial images. Hence, most of the users are forced to employ an trial-and-error means to determine the number of clusters when they use these clustering methods. It is a tedious task and the clustering result may be meaningless when the number of clusters is wrong. Therefore, determining a proper number of clusters and providing the appropriate clustering under this circumstance are real challenges. For aerial image analysis, these issues are extremely critical because an incorrect number of clusters may wipe out and neglect meaningful objects and merge them with the meaningless ones.

In this paper, a machine learning-based clustering method is applied to aerial images segmentation. The proposed approach employs support vector machines (SVMs) as the core clustering mechanism. One-class support vector machine (OCSVM) is a version of SVM [10]. Three cluster validity indices, including distance-based (DS) index, Davies-Bouldin (DB) index [11], and Xie-Beni (XB) index [12] are utilized as quantitative measures of the quality of clustering results. The feasibility of the proposed approach is demonstrated on several real aerial images.

The authors are all with Department of Electrical Engineering, National University of Kaohsiung, 700, Kaohsiung University Road, Nan-Tzu District, Kaohsiung 811, Taiwan
The corresponding author: Chih-Hung Wu, email: johnw@nuk.edu.tw

II. Related Work

Image segmentation can be achieved by a variety of approaches. This section presents several methods based on soft-computing and machine learning. A new multistage method using hierarchical clustering for unsupervised image classification was presented [13]. In the first phase, the multistage method performs segmentation using a hierarchical clustering procedure which confines merging to spatially adjacent clusters and generates an image partition such that no union of any neighboring segments has homogeneous intensity values. In the second phase, the segments resulting from the first stage are classified into a small number of distinct states by a sequential merging operation. Xia et al. [14] presented an approach to perceptual segmentation of images through the means of clustering of spatial patterns. An image is modeled as a set of spatial patterns defined on a rectangular lattice. The distance between a spatial pattern and each cluster is defined as a combination of the Euclidean distance in the feature space and the spatial dissimilarity which reflects how much of the pattern's neighborhood is occupied by other clusters. Cao et al. [3] proposed a novel segmentation algorithm implemented within a level sets framework to deal with multi-class partitioning problem. Focusing on low resolution photographic gray aerial images, their method performs feature extraction at the

first stage, evolves curves according to the proposed algorithm at the next stage and gets the partition result in the end. An evolutionary-fuzzy clustering algorithm for automatically grouping the pixels of an image into different homogeneous regions was presented in [15]. The algorithm does not require a prior knowledge of the number of clusters. The fuzzy clustering task in the intensity space of an image is formulated as an optimization problem. An improved variant of the differential evolution algorithm has been used to determine the number of naturally occurring clusters in the image as well as to refine the cluster centers.

III. One-Class Support Vector Machine

Support vector machine (SVM) has been emerging as a popular classifier due to its efficiency and ability to handle complex problems. Its basic idea to transform data points from data space to a high dimensional feature space using a kernel function so that the data points in the feature space become linearly separable. This study employs one-class SVM (OCSVM) for aerial image segmentation in an incremental manner. OCSVM [16] is a kernel-based method for constructing a classifier only using a set of positive training patterns. For a set of training patterns, OCSVM finds a hyperplane to separate these patterns from the origin O with maximum margin $\mathbf{w} \cdot \phi(\mathbf{x}_i) = \rho$ where $\mathbf{x}_i$ lies on the hyperplane H. The distance between the origin O and H is $\rho/\|\mathbf{w}\|$. To allow for the possibility of outliers in the data set and to make the method more robust, the projection value from an image on $\mathbf{w}$ need not be strictly larger than ρ, but the small projection value should be penalized. Therefore, slack variables $\xi_i, i = 1, \ldots, l$, are introduced to account for small projection values, and the objective function and constraints are the following:

$$\begin{aligned} \min \quad & \frac{1}{2}\|\mathbf{w}\|^2 - \rho + \frac{1}{\nu l}\sum_{i=1}^{l}\xi_i \\ s.t. \quad & \mathbf{w} \cdot \phi(\mathbf{x}_i) \geq \rho - \xi_i, \xi_i \geq 0, \forall \mathbf{x}_i \in l \end{aligned} \tag{1}$$

where ϕ is a nonlinear mapping from the input space to the feature space, and $\nu \in (0, 1]$ is a parameter which gives a trade-off between the maximum margin and the errors. With a small ν, the penalty on small projection values become substantial, thus few outliers should exist and the margin is small. On the other hand, when ν is large, many outliers with small projection values may exist to take advantage of the small penalty and the margin is generally large. In practice, the size of ν is an upper bound on the fraction of outliers and a lower bound on the fraction of support vectors [16].

To solve the constrained optimization problem, Lagrangian is introduced. Also, to avoid working in the high-dimensional feature space, we pick a feature space where the dot product can be calculated directly using a kernel function K in the input space, and the Wolfe dual form, which is a quadratic function in $\alpha_i' s$, becomes:

$$\begin{aligned} \min \quad & \frac{1}{2}\sum_{i,j=1}^{l}\alpha_i\alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \\ s.t. \quad & 0 \leq \alpha_i \leq \frac{1}{\nu l}, \sum_{i=1}^{l}\alpha_i = 1. \end{aligned} \tag{2}$$

This constrained optimization problem can be solved using a standard QP solver. Three commonly used kernel functions are Polynomial, Sigmoid, and RBF. Throughout this paper, the RBF kernel is adopted. That is:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-q\|\mathbf{x}_i - \mathbf{x}_j\|^2) \tag{3}$$

where q is the width parameter. This equation implies that each input point is mapped on the octant surface of the unit ball in the high-dimensional feature space.

It is reported that SVM usually outperforms other soft-computing methods such as artificial neural networks, genetic algorithms, in efficiency, scalability, and performance [17], [18], [19]. Many machine learning algorithms suffer from common disadvantages such as considerably long training time and over-fitting. Most of the reasons of these disadvantages are from the fact that they employ the empirical risk minimization (ERM) principle to learn patterns. ERM minimizes risks resulting from the training data only; thus it may overfit a model. When a lot of training data, such as in an aerial image, are processed, the training time and performance become worse. Conversely, SVM uses the structural risk minimization (SRM) principle to train a model. SRM incorporates the model complexity into a learning process and may avoid the overfitting problem encountered in many learning algorithms.

IV. The Proposed Approach

An aerial image can be considered as a collection of data points featured by color, texture, or other image properties. Points with specific features may organize meaningful and interesting objects, such as rivers, buildings, forest, etc. The objects consisting of pixels in the same or very similar features and pixels in different features may represent another objects. That is, meaningful objects in an aerial image can be figured out provided that the pixels can be properly clustered.

OCSVM is the kernel for image segmentation in this study. SVM is widely used for supervised classification and regression problems; it can also be applied to data clustering. OCSVM performs aerial image segmentation in an incremental manner. First, an image is roughly partitioned into k regions where $k \geq 2$ is given by the user. OCSVM is invoked for k times, each run produces one clustering model for a region. The resulting k clustering models are united for a complete model for clustering. If the evaluating result is unsatisfactory, a new round of OCSVM is invoked for k=k+1 times. This procedure iterates until the cluster validity index reaches a satisfactory result. Though there have been several SVM-based methods proposed for unsupervised clustering, their performance and difficulty in determining the number of clusters still need some improvement. The use of OCSVM for aerial image segmentation is described as follows.

A. OCSVM for Clustering

Let I_0=$\{p_1, p_2, \ldots, p_n\}$ be an aerial image consisting of n pixels. Each p_i, $1 \leq i \leq n$ has m features $f_1, f_2, \ldots, f_m$. By the kernel function, $p_1, p_2, \ldots, p_n$ are mapped into a high-dimension feature space. Then, I_0 can be partitioned into a given number of k clusters by training OCSVM iteratively for

k times. In each iteration, a subset of image I_i, $1 \leq i \leq k$, is selected from I_0, where $I_i \cap I_j$=$\emptyset$, $1 \leq i,j \leq k$, $i \neq j$, and $\cup_i I_i$=I_0. I_i is used as the training dataset fed into OCSVM. The data which can be separated by the hyperplane satisfying Eq.(1)–Eq.(3) form a cluster. The support vectors associated with the corresponding boundaries of the hyperplane organize a clustering model C_i for I_0. By repeating the same procedure for k times, a k-clustering models C can be obtained such that C=$\{C_1, \ldots, C_k\}$.

It is assumed that data in each I_i are separable by one run of OCSVM. The data points included in a cluster C_i are featured by most significant features. The ones excluded by C_i may belong to another cluster C_j, $i \neq j$, which will be formed in a later-on iteration of OCSVM. Therefore, post-processing is needed to form a complete clustering model by joining all C_i and C_j. Notice that, it is possible that two clusters C_i and C_j obtained in different iterations may contain the same support vectors. This is because that the data points in I_i and I_j belong to the same area in I_0 and they both are featured by the same dominating features. In such cases, the number of clusters may be less than k.

B. Sampling Data for Training OCSVM

When using OCSVM, there are two important factors, i.e., selecting I_i from I_0 and determining the number k of clusters, to be carefully considered. Theoretically, if all data belonging to the same cluster can be completely extracted from I_0 and then retained in I_i, i.e., I_i includes only this cluster, the best OCSVM result can be obtained. This is possible only when a sophisticated clustering algorithm is performed on I_0 and can successfully extract a complete cluster in each iteration of OCSVM. Unfortunately, this is impossible since the clustering algorithm is not performed yet. Besides, it may need considerable overheads on running this pre-clustering algorithm which may degrade the overall performance of OCSVM. This study uses heuristics for fast determining which pixels are included in I_i. These sampling heuristics are simple but fast, which may only slightly delay the running of later-on OCSVM. They are presented as follows.

1) Stripe sampling: I_0 is evenly partitioned alone the x- or y-axis into k stripes in the same size. The pixels in a strips form an I_i.
2) Block sampling: I_0 is evenly partitioned into $p \times q$ blocks $b_{x,y}$, $1 \leq x \leq p$, $1 \leq y \leq q$. $b_{x,y}$ is in the same size. Each I_i takes the block $b_{x,y}$ satisfying $((x+1)\times p+y) \bmod i = 0$.
3) Skew sampling: Similar to stripe sampling, but each I_i skewly picks up a stripe alone the diagonal.

With the extracted I_i, OCSVM is invoked for each I_i. Next, the concept of *cluster validity index* is introduced.

C. Cluster Validity Indices

Clustering validity is a measure for evaluating the goodness of clustering results. This section contains the description of three cluster validity indices that have been used.

1) Distance-based Index: The distance-based index is the most popularly used function that minimizes the total distance within cluster variation, which is defined as

$$DS = \sum_{i=1}^{k} \sum_{x \in C_i} d(x, \overline{x}_i) \tag{4}$$

where $\overline{x}_i$ denotes the centroid of the ith cluster, $d(x, \overline{x}_i)$ is the distance between the data point x and its nearest cluster center. Thus, this index attempts to minimize the distance of each data point from the center of the cluster to which the point belongs. The optimal number k is chosen when the distance-based index reaches its minimum.

2) Davies-Bouldin Index: The Davies-Bouldin index (DB-index) is a function of the ratio of the sum of within-cluster scatter to between-cluster separation. The scatter within the ith-cluster and the distance between the i-th and j-th cluster can be defined as

$$S_{i,q} = \left(\frac{1}{|C_i|} \sum_{x \in C_i} \{ \|x - \overline{x}_i\|_2^q \} \right)^{1/q}, \tag{5}$$

$$\overline{x}_i = \frac{1}{|C_i|} \sum_{x \in C_i} x, \tag{6}$$

and

$$d_{ij,t} = \|\overline{x}_i - \overline{x}_j\|_t = \left\{ \sum_{s=1}^{p} |\overline{x}_{is} - \overline{x}_{js}|^t \right\}^{1/t}, \tag{7}$$

where $\overline{x}_i$ and $|C_i|$ represent the centroid and the number of elements in the i-th cluster, respectively.

We use $S_{i,1}$, the average Euclidean distance of the vectors in the i-th cluster to its centroid, in the proposed approach. $d_{ij,t}$ is the Minkowski distance of order t between the centroids which characterize clusters C_i and C_j. Next, a term $R_{i,qt}$ for C_i is defined as

$$R_{i,qt} = \max_{j, j \neq i} \left\{ \frac{S_{i,q} + S_{j,q}}{d_{ij,t}} \right\}. \tag{8}$$

Then the DB-index for k-clustering is defined as

$$DB = \frac{1}{k} \sum_{i=1}^{k} R_{i,qt}, \tag{9}$$

where k is the number of clusters. Since it is desirable for the clusters to have the minimum possible similarity to each other; therefore, the objective is to minimize the DB-index for achieving proper clustering.

3) Xie-Beni Index: For hyper-spherical-shaped clustering algorithms, the most used cluster validity proposed by Xie and Beni is formulated as:

$$XB = \frac{\sum_{s=1}^{k} \sum_{j=1}^{N} u_{sj}^2 \|x_j - \overline{x}_l\|^2}{N \cdot \min_{ij} \|\overline{x}_i - \overline{x}_j\|^2}, \tag{10}$$

where u_{sj} is the membership of the j-th point to the l-th cluster, k is the number of clusters and N is the total number of points in the data set. The value of $\overline{x}_i$ represents the center of the i-th cluster. In Eq.(10), the smaller the separation validity function value is, the more probability there will be

(a) Image-imgA (b) Image-imgB

Fig. 1. The original aerial images

redundant cluster representative in the existed representatives. The optimal number k is then chosen when the XB-index reaches its minimum.

Notice that these indices are not comparable since they are in different basis. They are all as small, approaching to 0, as better. However, they may be pretty large and become incomparable (e.g., $\gg$1). In order to make them quantitatively comparable in the experiments, they are scaled into $[0, 1]$ as

$$\textbf{Scaled index} = \frac{1}{\textbf{(1+Index)}}, \tag{11}$$

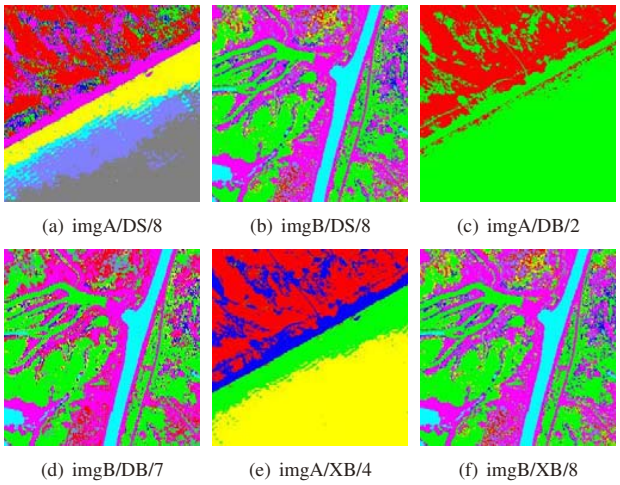
where **Index** can be DS, DB, or XB. In Eq.(11), an index value more approaching to 1 represents a better clustering result.

V. Experiments

This section presents several experiments for demonstrating the performance of the proposed method. An aerial image segmented into either too few or too many clusters is both less informative. In the experiments, the maximum number `MaxC` of clusters is 8. an image segmented into 2 or `MaxC` of clusters is in such cases. As stated in many image processing literatures, the performance of image segmentation usually depends on the clustering methods and the associated parameters. It is also domain dependent. The three aerial images shown in Figure 1 are taken from the public domain [1] and used as the test bench. Each image is 128×128 in size. We include libsvm[20] as part of the OCSVM engine. For parameters in Eq.(1) and Eq.(3), throughout this paper ν = 0.05 meaning that there are at most 5% of outliers in the training patterns and q=0.25 for the RBF kernel function. Basic attributes such as pixel's location and color are used as features for clustering. The cluster validity indices are normalized by Eq.(11) and are used as the criterion to evaluate the number of clusters. Currently, the OCSVM segmentation process is implemented in C#. All experiments are conducted on a personal computer equipped with a P4 2.0GHz processor and 512 MB memory running Windows XP. The figures presented in Figure 2–Figure 4 are the best results. Each figure is denoted by x/y/z which means that the image x is segmented into z clusters using y as the cluster validity index.

First, the sampling strategy for one-run of OCSVM is discussed. There are three sampling strategies, i.e., stripe, block, and skew. As stated in Section IV-B, if all data belonging to the same cluster are completely extracted from the original image and used to train OCSVM, the best clustering result can

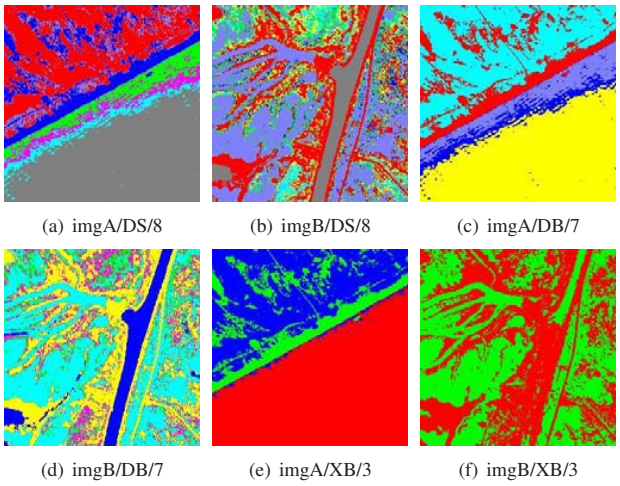
[1] http://www.texmaps.com/texmaps-free.html and http://www.usgs.gov/



(a) imgA/DS/8 (b) imgB/DS/8 (c) imgA/DB/2

(d) imgB/DB/7 (e) imgA/XB/4 (f) imgB/XB/8

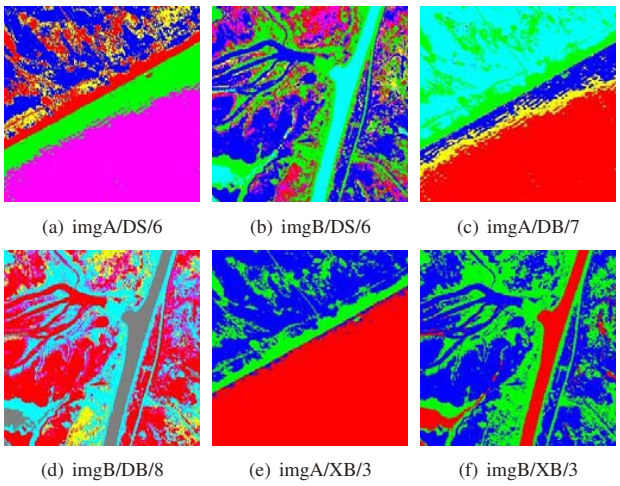
Fig. 2. OCSVM using stripe partitioning



(a) imgA/DS/8 (b) imgB/DS/8 (c) imgA/DB/7

(d) imgB/DB/7 (e) imgA/XB/3 (f) imgB/XB/3

Fig. 3. OCSVM using block partitioning



(a) imgA/DS/6 (b) imgB/DS/6 (c) imgA/DB/7

(d) imgB/DB/8 (e) imgA/XB/3 (f) imgB/XB/3

Fig. 4. OCSVM using skew partitioning

be obtained. Observe Figure 1, the distribution of data points belonging to different areas in Figure 1(a) more concentrate than that in Figure 1(b). That is, whatever a strategy selects a set of data points from Figure 1(b) for OCSVM, the dataset may contain various data points from different area so that OCSVM can not produce a good cluster. Conversely, the stripe and skew selecting strategies are useful for working on Figure 1(a). Block sampling strategy degrades the performance when blocks are finer-grained. In this experiment, the number of blocks is 128×128. Finer-grained partitioning makes the selected data points to have similar distributions as that in the original image.

Next, the performance of the cluster validity indices is presented. From the experimental results, DS-index intends to guide the clustering mechanism to as many clusters as possible. In subfigures (a)–(b) of Figure 2–Figure 4, imgA and imgB are segmented into many clusters, where 4 out of 6 results are 8 clusters. It is reasonable, by referring to Eq.(4), that the distance between two data points in smaller clusters is smaller as well. An image segmented into many clusters can offer many smaller clusters. DB-index and XB-index consider advanced information and outperform than DS-index. However, in these experiments, their performance is variable. XB-index seems to intend to simplify the segmentation results, while DB-index intend to figure out the details. From these experimental results, it is found that the proposed approach with XB-index can produce a spatially smooth class map with a more distinctive configuration of the classes than the approach with DB-index and DS-index. For example, we can distinguish lands from the ocean. The contour of a river is clearly recognized by the proposed approach by XB-index. Observe subfigures (c)–(f) of Figure 2–Figure 4, 4 out of 6 results are 7 clusters in DB-index and 4 out of 6 results are 3 clusters in XB-index.

Each round of image segmentation stops when the indices reach a local optima. In order to compare the effectiveness of cluster validity indices, the values of these indices in all possible settings of k are listed in Table I. From these values, it is found that not all best results appear in the same settings. However, the trends seem that most best results concentrated at cases when stripe or skew partitioning is used. For example, 5 out of 7 column best values in imgA are from stripe and skew partitioning. In both images, the best values of cluster validity index are from skew partitioning. Interestingly, column best in imgB appear at the case where block partitioning is applied. The same reason mentioned previously about the distribution of data points in imgB can explain this phenomenon. In all images, XB-index seems to give OCSVM better performance than the other two indices. 6 out of 6 best values come from the cases where XB-index are applied. The XB-index is considered more reliable than other cluster validity criteria because it considers all the information in the partition and adopts the concept of compactness-to-separation ratio. The numerator of the XB-index is a compactness function that fits the objective function of the clustering algorithm to reflect the compactness of clusters. The denominator of the XB-index is a separation function that measures the separation status of clusters. However, these are not strict conclusions. There are more factors to be considered.

It is well known that evaluating segmentation results and comparing the related methods are not simple tasks. The better evaluation is to compare the segmentation results with the associated ground truth images. However, if the ground truth images are unavailable, Haralick and Dhspito [21] provided four declarative segmentation criteria for reference. They are (1) segmented regions should be uniform and homogeneous, (2) regions interiors should be simple and without many small holes, (3) adjacent regions should have significantly difference, and (4) boundaries of each region should be simple and spatially accurate. According to these criteria, it may conclude that the segmentation results shown in Figure 2–Figure 4 with XB-index are better than those of obtained with other indices.

VI. Conclusion

In this paper, OCSVM is employed for the segmentation of aerial images. The proposed segmentation approach can automatically determine the proper number of clusters and then a partitioning of the given image is done. From the experimental results, the proposed approach with XB-index can actually find the appropriate number of regions as well as proper segmenting of an image.

Acknowledgment

This work was supported by the National Science Council, Taiwan, under grants NSC 98-2221-E-390-026 and NSC 98-2221-E-390-027. The authors would also like to thank Mr. Chi-Yuan Yeh for his comments for improving the performance of OCSVM.

References

[1] Y. Hata, S. Kobashi, S. Hirano, H. Kitagaki, and E. Mori, "Automated segmentation of human brain MR images aided by fuzzy information granulation and fuzzy inference," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 30, no. 3, pp. 381–395, Aug 2000.

[2] V. Letournel, B. Sankur, F. Pradeilles, and H. Maître, "Feature extraction for quality assessment of aerial image segmentation," in *Proceedings of the ISPRS Technical Commission III Symposium 2002, Photogrammetric Computer Vision (PCV'02)*, Graz, Austria, 2002, pp. 141–163.

[3] G. Cao, Z. Mao, X. Yang, and D. Xia, "Optical aerial image partitioning using level sets based on modified chan-vese model," *Pattern Recognition Letters*, vol. 29, no. 4, pp. 457 – 464, 2008.

[4] Z. Iscan, A. Yksel, Z. Dokur, M. Korrek, and T. lmez, "Medical image segmentation with transform and moment based features and incremental supervised neural network," *Digital Signal Processing*, vol. 19, no. 5, pp. 890 – 901, 2009.

[5] S. Kavitha, S. M. M. Roomi, and N. Ramaraj, "Lossy compression through segmentation on low depth-of-field images," *Digital Signal Processing*, vol. 19, no. 1, pp. 59 – 65, 2009.

[6] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, 1999.

[7] J. T. Tou and R. C. Gonzalez, *Pattern Recognition Principles*. Addison-Wesley Pub. Co., Reading, Mass.,, 1974.

[8] M. Herbin, N. Bonnet, and P. Vautrot, "Estimation of the number of clusters and influences zones," *Pattern Recognition Letters*, vol. 22, no. 14, pp. 1557–1568, 2001.

[9] J. Kang, L. Min, Q. Luan, X. Li, and J. Liu, "Novel modified fuzzy c-means algorithm with applications," *Digital Signal Processing*, vol. 19, no. 2, pp. 309 – 319, 2009.

[10] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. New Jersey: Cambridge University Press, 2000.

TABLE I
VALUES OF THE CLUSTER VALIDITY INDICES OF THE BEST SEGMENTATION RESULTS (NOTE: **bold face**: MAX. VALUE IN THE ROW; ( ): MAX. VALUE IN THAT THE COLUMN k; *: MAX. VALUE IN THE SAMPLING STRATEGY; **: MAX. VALUE IN THE IMAGE OF ALL MEANS)

Image	Method	Index	k=2	k=3	k=4	k=5	k=6	k=7	k=8
imgA	stripe	DS	0.87406	0.88130	0.88623	0.88959	0.88938	0.89122	**0.89194**
		DB	**0.95431**	0.95040	0.95255	0.95149	0.95224	0.95348	0.95284
		XB	(0.96389)	0.96492	***(0.96588)**	0.96547	0.96545	0.96550	(0.96520)
	block	DS	0.86371	0.86969	0.87240	0.87078	0.87101	0.87120	**0.87328**
		DB	0.95345	0.95815	0.95200	0.95834	(0.96748)	**0.97005**	0.95821
		XB	0.96208	***0.97237**	0.96473	(0.96974)	0.96450	0.96447	0.96408
	skew	DS	0.86731	0.87142	0.87320	0.87205	**0.87480**	0.87225	0.87433
		DB	0.95379	0.95814	0.95646	0.95692	0.95283	**(0.97046)**	0.96378
		XB	0.96246	****(0.97263)**	0.96475	0.96439	0.96457	0.96465	0.96432
imgB	stripe	DS	0.86760	0.86959	0.87321	0.87402	0.87467	0.87485	**0.87525**
		DB	0.95041	0.95024	0.95091	0.95163	0.95182	**0.95236**	0.95222
		XB	0.95905	0.95888	0.95979	0.95983	0.96002	0.96004	***0.96008**
	block	DS	0.86981	0.87203	0.87565	0.87611	0.87763	0.87704	**0.87801**
		DB	0.95698	0.95011	0.95433	0.95511	0.95438	**(0.96778)**	0.96102
		XB	****(0.96977)**	0.95936	(0.96031)	0.96011	(0.96032)	0.96032	0.96031
	skew	DS	0.86878	0.87130	0.87521	0.87635	**0.87675**	0.87657	0.87666
		DB	0.95044	0.95495	0.95070	0.95515	0.95357	0.95891	**(0.96317)**
		XB	0.95943	***(0.96733)**	0.96013	(0.96017)	0.96020	0.96015	0.96014

[11] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 2, pp. 224–227, 1979.
[12] X. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 8, pp. 841–847, 1991.
[13] S. Lee and M. M. Crawford, "Unsupervised multistage image classification using hierarchical clustering with a bayesian similarity measure," *IEEE Transactions on Image Processing*, vol. 14, no. 3, pp. 312–320, 2005.
[14] Y. Xia, D. Feng, T. Wang, R. Zhao, and Y. Zhang, "Image segmentation by clustering of spatial patterns," *Pattern Recognition Letters*, vol. 28, no. 12, pp. 1548–1555, 2007.
[15] S. Das and A. Konar, "Automatic image pixel clustering with an improved differential evolution," *Applied Soft Computing*, vol. 9, no. 1, pp. 226–236, 2009.
[16] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the Support of a High-Dimensional Distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1472, July 2001.
[17] D. Li, R. M. Mersereau, and S. Simske, "Blind image deconvolution through support vector regression," *IEEE Transactions on Neural Networks*, vol. 18, no. 3, pp. 931–935, 2007.
[18] Z. L. Wu, C. H. Li, J. K. Y. Ng, and K. R. Leung, "Location estimation via support vector regression," *IEEE Transactions on Mobile Computing*, vol. 6, pp. 311–321, 2007.
[19] L. Cao, "Support vector machines experts for time series forecasting," *Neurocomputing*, vol. 51, pp. 321–339, 2003.
[20] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," http://www.csie.ntu.edu.tw/∼cjlin/libsvm, 2001.
[21] R. Haralick and L. G. Dhspito, "Image segmentation techniques," *Applications of Artificial Intelligence II*, vol. 548, pp. 2–9, 1985.