1K2-IOS-1b-1

Using Soft Case-Based Reasoning in Model Order Selection for Image Segmentation Ensemble

Pakaket Wattuya*1 Nuanwan Soonthornphisaj*2 Xiaoyi Jiang*3

*1 *2 Department of Computer Science, Kasetsart University, Thailand

*3 Department of Computer Science, University of Münster, Germany

The desired number of clusters in clustering problem is generally not known in advance. In this work, we propose to use case-based reasoning as a novel problem solving technique for automatic model order selection with application to image segmentation ensemble. Soft computing technique is integrated in our case-based reasoning to handle ambiguity and uncertainty in image data. Given the fact that we do not know the optimal number of regions for a particular image in advance, the comparative performance of our approach is remarkable and reveals its potential in dealing with the difficult model order selection without ground truth. Moreover, our approach can be easily integrated into a general class of image segmentation system that prevents a segmentation algorithm from exhaustively searching for optimal segmentations. Extensive experiments on 300 images have been conducted and our preliminary results show the effectiveness of our approach.

1. Introduction

For most clustering problems there is little prior information (e.g., statistical models) available about the data. Thus, the desired number of clusters is not known in advance and is often specified by a human user. Recently, the idea of using multiple segmentations has emerged in the area of model order selection problem. Cho and Meer [Cho 97] proposed a new approach of image segmentation based on a co-occurrence probability field derived from the consensus of a set of different segmentation outputs on a single input image. Starting from an over-segmented result, the algorithm computed a final segmentation result by iteratively merging together the pixel pairs with high co-occurrence probability. Rabinovich et al. [Rabinovich 06] developed a model order selection schema based on cluster stability and used it to find a shortlist of most stable segmentations from a large number of possible segmentations. Wattuya et al. [Wattuya 08] proposed also an optimization approach for selecting one optimal segmentation from a given set of segmentation results. In their framework, they produced several initial segmentations of the same input image by varying values of segmentation algorithm parameters. Then, a segmentation combination algorithm is applied to combine these initial segmentations to produce a final combination segmentation result. Since the optimal number of regions is not known, they forced the segmentation combination algorithm to produce a series of combination results with different number of regions, k, in a meaningful range $[k_{\min}, k_{\max}]$. Normally, the range $[k_{\min}, k_{\max}]$ is not small, so that it can adequately cover all possible meaningful segmentations. Then, optimality criterion based on generalized median concept is applied to select the optimal result out of the set of combination solutions. The extension of this work [Wattuya 10] proposed to used objecttive function based on minimum description length instead of generalized median concept. The results showed slightly improvement over the previous work.

In this work, we propose to use a case-based reasoning as a problem solving technique for automatic model order selection. We begin with the observation that the complexity of one image should reflect something about its own number of regions. Images with high visual complexity should naturally contain more salient regions (objects) than images with low visual complexity. In other words, images with similar degree of visual complexity should contain similar number of regions. A case-based reasoning is built upon this assumption to infer the potential, small range of a number of regions.

The proposed case-based reasoning can be integrated into the existing image segmentation system, in order to reduce a search space/a problem size which needs to be processed. For example, we can integrate our case-based reasoning to [Rabinovich 06] by presenting a small set of potential segmentations to a model order selection module to find a shortlist of most stable segmentations, instead of starting from a large set of possible segmentations. By doing this way, the model order selection computation may converge to an optimality criterion faster and yield better optimal results. Our case-based reasoning approach is also possible to be integrated into the segmentation combination framework proposed in [Wattuya 08], which would (i) significantly reduce the amount of work needed for producing a large set of combination results with different $k \in [k_{\min}, k_{\max}]$ to a smaller set of potential segmentations, and (ii) significantly reduce the amount of time needed to search for a final op-

Contact: P. Wattuya, N. Soonthornphisaj are with Department of Computer Science, Kasetsart University, 50 Ngam Wong Wan Rd., Chatuchak, Bangkok Thailand 10900, Phone +662 5625444, Fax +662 9428488, Emails:{fscipkw, fscinws}@ku.ac.th

Contact: X. Jiang is with Department of Computer Science, Department of Computer Science, University of Münster, Einsteinstr. 62 D-48149 Münster, Germany, Phone +49 2518333759, Fax +49 2518333755, Email: xjiang@math.uni-muenster.de

timal segmentation by restricting a large search space of a meaningful range $[k_{\min}, k_{\max}]$ to an smaller subspace of a potential range reasoned by a case-based reasoner.

We verify the effectiveness of our case-based reasoning in an application of image segmentation combination proposed in [Wattuya 08]. The experiments show good results on our approach. Moreover, they demonstrate a valuable side benefit of our proposed case-based reasoning in alleviating the poor performance caused by the instability of the objective function used in an optimal segmentation selection method. Novel features of our case-based reasoning include the ability to predict a very small range of valid number of regions without the need of ground truth segmentations and the ability to integrate into a general class of image segmentation that free a segmentation algorithm from exhaustively searching for potential segmentations.

In image processing and computer vision, case-based reasoning techniques have been successfully used to improve the performance in various applications, for example, Perner [Perner 99] used case-based reasoning framework for the high-level unit of an image interpretation system. Frucci et al. [Frucci 08] used case-based reasoning for handling parameter selection problem in image segmentation algorithm. To our knowledge, our work is the first attempt of using case-based reasoning to handle the problem of model order selection.

In the next section, we continue with an overview of our proposed case-based reasoning. The full details of building it are explained in Section 3. In Section 4, experimental validation and discussion are reported, followed by some discussions to conclude this paper.

2. Overview of the Proposed Soft Case-Based Reasoning

Case-based reasoning in general involves solving new problems by identifying and adapting solutions to similar problems stored in a library of past experiences. In our specific problem domain, a case-based reasoning is built upon the knowledge derived from original image and its segmentation ensemble. It takes an image and its segmentation ensemble as input, and gives a small potential range of k as an output. Once the case-based reasoning unit is given an input image and its corresponding segmentation ensemble, the relevant image features are extracted from both inputs. The case-based reasoner will select from a case base the cases having most similar features to those of the current input image. Then, the solutions of retrieved cases are appropriately adapted to fit the new current problems before fed into an image segmentation combination unit to further process.

One of the important issues related to building case-based reasoning is how to definitively describe the problem specification in order to ensure that a case will be retrieved in the most appropriate context. This requires the cases to contain the relevant features of the problem and its context that influenced the outcome of the solution. Based on our assumption between the degree of visual complexity and its natural link to a number of regions in an image, we have to look for features that can characterize and discriminate the visual complexity between images well. However, it is difficult to precisely model the complexity of an image, particularly, related to the number of regions. Case-based reasoning can simplify our problem since it does not require any explicit model or rules. The knowledge acquisition tasks of case-based reasoning consist primarily of collection of relevant existing cases and their representation and storage. However, once cases are retrieved from the case base, it is possible that they are not identical to the current case, possibly caused by using indescriptive or incomplete features. A case-based reasoner can deal with this situation, although these factors may cause a slight degradation in performance (due to the increased disparity between the current and retrieved cases). In this work, we attempt to prevent performance degradation by integrating soft computing into case-based reasoning. The experimental results show that soft computing technique can manage this situation well.

The use of fuzzy logic in case-based reasoning systems dates back to the early 1990s, when researchers started to use attributes with fuzzy values and a fuzzy pattern matcher for case retrieval. Soft case-based reasoning has been well documented in [Pal 04]. In image processing and computer vision, there are numerous works that use soft computing to deal with uncertainty and vagueness of image data, for example, a survey of soft computing approach to image segmetation can be found in [Senthilkumaran 09].

3. Soft Case-Based Reasoning for Model Order Selection

The important steps in the inference cycle of case-based reasoning are to retrieve cases from the library which are most relevant to the problem at hand and adapt the retrieved cases to the current input. In this section we introduce how to design these essential components and processes that make up a case-based reasoning for solving the model order selection problem. Our attempt is to predict a small range with reasonable number of regions that is large enough to capture all salient objects in an image.

3.1 Case Representation

Cases in a case base are usually represented as two unstructured sets of attribute-value pairs that represent the problem and solution features. In this work, a case is represented by a flat feature-value vector comprised of problem features describing image characteristics of visual complexity, $< f_1, ..., f_p >$, and the solution that stores a potential, small range of number of region, $< r_{\min}, r_{\max} >$.

Problem Features

The development of a case base requires objective measures of visual complexity of an image. The problem of evaluating the complexity of an image is of a certain relevance to both cognitive and computer science studies, although in broader contexts the general problem of visual complexity measurement is ill-defined [Cardaci 06]. In the context of our problem, we define image complexity as a measure of image information that causes regions in the image. Images possessing large amount of such information should inherently lead to great perceived complexity (i.e. a lot of objects or regions). Hence, images that share most common such information should have a number of regions in the same small range of k. In order to effectively capture visual complexity of an image, we take into account both (i) information from low-level features (i.e. texture) extracted directly from image data, and (ii) valuable information (i.e. average number of regions) derived from a segmentation ensemble, which play an important role in our approach.

Texture Feature. Texture is one of the important characteristics used in identifying objects or regions of interest. Texture features contain information about the spatial distribution of tonal variations within a band. The texture features we used in this work are based on the gray-tone spatial-dependence matrices proposed by Haralick et al. [Halkidi 73]. The gray-tone spatial-dependence matrices can reveal certain properties about the spatial distribution of the gray levels in the texture image. For example, if most of the entries in the the gray-tone spatial-dependence matrices are concentrated along the diagonal, the texture is coarse with respect to the specified offset. Four statistical measures, which are *contrast*, *correlation*, *homogeneity*, and *entropy*, extracted from the the gray-tone spatial-dependence matrices are used. *Contrast* is a measure of the amount of local variations present in an image. *Correlation* is a measure of how correlated a pixel is to its neighbor over the whole image. Contrast and correlation are zero for a constant image. *Homogeneity* measures uniformity of the gray value in an image. *Entropy* measures characterize the complexity and nature of gray-level transitions which occur in the image. Images with more gray levels have lower average homogeneity but higher average entropy.

Average Number of Regions. We propose to use a new high-level feature involving an approximate natural number of regions by employing knowledge from image segmentation ensemble. A different number of regions among the different initial segmentations are a good approximation about a natural number of regions of a given image. We conjecture about general characteristics of segmentation results computed by an arbitrary segmentation algorithms that the good quality segmentations of the same image are quite alike (more or less), while the bad segmentations are arbitrarily bad in its own way. If we assume that about the half of initial segmentations in an ensemble have acceptable good quality, the average number of regions among all initial segmentations is a good statistical approximation one. Since segmentations with minimum/maximum number of regions in an ensemble would have high possibility to be bad segmentations in an ensemble, directly use of minimum/maximum values would cuase high possibility to falsify the natural number of regions. However, in practice our assumption may be wrong in some situations since we did not control the quality of initial segmentations in an ensemble. The majority population of a segmentation ensemble could be relatively bad for difficult-to-segment images.

All five image features defined in this section will be used for indexing and retrieving of a set of cases close to the current problem, based on a proper similarity measure. However, the task of defining relevant image features and how to weigh their importance are difficult and need further study. One can use more sophisticated image analysis techniques to explore more relevant features with higher discriminative ability, for example, the work [Cardaci 09] proposed to evaluate the image complexity using fuzzy approach.

Case Solution

As we mentioned earlier, the consensus of different initial segmentations provides valuable information for approximation of natural number of regions in a given image. Even though direct use of minimum/maximum number of regions of an ensemble could misinterpret a natural number of regions, these two extreme values are the best available approximation of a lower bound and an upper bound of a reasonable range of possible k. In other words, a natural number of regions should fall within this range. Hence, we decide to use the range $[r_{\min}, r_{\max}]$ as a solution of a case.

3.2 Case Generation and Retrieval

Case selection and retrieval is usually regarded as the most important step within the case-based reasoning cycle since the remaining operations of adaptation and evaluation will succeed only if the selected cases are of relevant ones. In general, in the process of case matching and retrieval, the searching space is the entire case base. Too many cases stored in a case base will make the task costly and inefficient. Thus, many classification and clustering algorithms have been proposed to address such problem. After the cases are partitioned into several subclusters, a set of representative cases is selected beforehand based on clustering result. Then, the task of case matching and retrieval boils down to matching the new case with one of the several subclusters, and finally, the desired number of similar cases can be obtained.

In this work, the above schema is also applied but for different purpose. Our use of clustering algorithm mainly aims for *generalization* of the case base. The essential reason for the need of case base generalization is that since it is hard to precisely define all relevant features and their importance in our problem domain, case descriptions may contain misprecision or incomplete data. Thus, by using an individual case to reason the new current problem could indeed yield an unsatisfactory result. We propose to handle this problem by applying soft computing technique.

Case Generalization using Fuzzy c-Means Clustering

We achieve a case base generalization by using fuzzy c-means (FCM) algorithm. FCM has proven to be very effective for solving many cluster analysis problems and is widely used for case clustering. In our work, distribution of cases in a case base has no clear cut boundaries between clusters. They partially overlap among them in the feature space. We use FCM to handle uncertainties arising from such overlapping clusters, since it allows one case to belong to multiple clusters with similar or different degree of belongness.

To develop FCM algorithms, we hypothesize that cases are represented in an n-dimensional feature space where the axes represent the variables (features) and cases become points (vectors) in the space. Let $X = \mathbf{x}_1, ..., \mathbf{x}_n$ be the set of given cases and let c be the number of clusters ($1 < c < n$). The basic idea of determining the fuzzy clusters is by minimizing the following objective function [Bezdek 81]:

$$J(U, C) = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d_{ij}^2 \tag{1}$$

where $C = \{c_1, ..., c_c\}$ is the set of cluster centers (or cluster prototypes), d_{ij} is the distance between $\mathbf{x}_j$ and cluster centers $\mathbf{c}_i$, the parameter m, $m > 1$, is called weighting exponent used to determine the fuzziness of the classification, and U is a $c \times n$ matrix, where u_{ij} is the ith membership value of the jth input sample x_j. The membership values satisy the following conditions:

$$0 \le u_{ij} \le 1, \quad i = 1, ..., c; j = 1, ..., n$$

$$\sum_{i=1}^{c} u_{ij} = 1, \quad j = 1, ..., n$$

$$0 < \sum_{i=1}^{c} u_{ij} < n, \quad i = 1, ..., c$$

Euclidean distances between each input case and its corresponding cluster center are weighted by the fuzzy membership values. The algorithm is iterative and uses the following equations:

$$c_i = \frac{1}{\sum_{j=1}^{n} u_{ij}^m} \sum_{ij}^{n} u_{ij}^m x_{ij}, \quad i = 1, ..., c \tag{2}$$

$$u_{ij} = \frac{\left[1/\left|x_j - c_i\right|^2\right]^{\frac{1}{(m-1)}}}{\sum_{k=1}^{c} \left[1/\left|x_j - c_k\right|^2\right]^{\frac{1}{(m-1)}}}, i = 1, ..., c; k = 1, ..., n \tag{3}$$

Given the number of cluster centers c, and the exponent weight m, FCM clustering procedure for generalizing a case base consists of the following steps:

Step 1. Initialize $U^{(0)}$ randomly; initialize $C^{(0)}$ and calculate $U^{(0)}$. Set the iteration counter $\alpha = 1$.

Step 2. Compute the cluster centers $C^{(\alpha)}$ according to equation (2) given the membership value matrix, $U^{(\alpha)}$.

Step 3. Update the membership values $U^{(\alpha)}$ according to equation (3) given the set of cluster centers, $C^{(\alpha)}$.

Step 4. Stop the iteration if the maximum number of iterations is reached or when the improvement between two consecutive iterations is less than the minimum amount of improvement specified (ϵ).

$$\max \left| u_{ij}^{(\alpha)} - u_{ij}^{(\alpha-1)} \right| \le \epsilon$$

else let $\alpha = \alpha + 1$ and go to step 2.

Step 5. We then used the resulting cluster centers $C = \{c_1, \ldots, c_c\}$ as generalized version of cases in the case base.

There are two main reasons why the cluster centers are suitable for representing the cases in the case base: (i) FCM finds the most characteristic point in each cluster as the 'center' of the cluster. (ii) FCM can capture the actual data distribution (by using the training data) which will interpret the future data well.

By doing this way, the number of cases in the case base, as well as the number of case matching, is reduced to c. After a set of matching cases were retrieved, the solutions stored in the retrieved cases will be adapted into a final solution which is fit for the current problem. An algorithm used to carry out this task is described in Section 3.3.

Similarity Measure and Case Retrieval

Case retrieval is the process of finding within the case base the prototypes that are the most promising classes to the current case. The retrieval algorithm is begun by deciding to which classes the present case is closest to. The certainties of the case prototypes and the matching degree are expressed by means of Euclidean distance. The two closest case prototypes are selected here for using in a case adaptation which has been argued that adaptation may be the most important step of case-based reasoning since it adds intelligence to what would otherwise be simple pattern matchers.

3.3 Case Adaptation

In problem solving CBR, case adaptation is the process of transforming a solution of retrieved cases into a solution appropriate for the current problem. In this section we describe an algorithm for deriving the final solution, $[\tilde{k}_{\min}, \tilde{k}_{\max}]$, from each individual case solution $[r_{\min}, r_{\max}]$, in order to make it suitable for solving the current problem.

We begin by noting that when cases in a case base are partitioned into c clusters with different degree of membership, it is not necessary that all case members in the same cluster share the same solution. The differences between solutions associted with cluster members can vary from small to large. Membership value derived from FCM allows us to take into account the relative level of importance of each case member in the cluster. Thus, cases that share fewer features of the cluster (i.e. has higher distances from case prototype or lower membership values) will have fewer contribution (or less influence) to a new solution than cases highly close to a case prototype. Furthermore, one case can give a contribution to more than one cluster (but in different degrees), especially, cases that lie on transition between two clusters. This property comes in useful to compensate the ambiguity of case desciption in our domain problem.

Since we are working on the space of uncertain and/or incomplete data, directly using the solution of only one (closest) retrieved case may lead to unsatisfactory results. The basic idea underlying our algorithm is that: Once the new case $\hat{c}$ presented in the feature space, it certainly lies between the two closest cluster centers c_i and c_j. This means that the features of the current case is likely to be similar to the features of either cluster protype c_i or c_j, and the solution of the new case $\hat{c}$ should be contributed by the two cluster protype c_i or c_j. Thus, we define the new solution of the new case $\hat{c}$ as $[\tilde{k}_{\min}, \tilde{k}_{\max}]$, where $\tilde{k}_{\min} = \min(\bar{k}_i, \bar{k}_j)$, $\tilde{k}_{\max} = \max(\bar{k}_i, \bar{k}_j)$, and $\bar{k}_i$ and $\bar{k}_j$ are *fuzzy mean region* of cluster protype c_i and c_j, respectively. *Fuzzy mean region* can be derived from all fuzzy solutions $[\tilde{r}_{\min}, \tilde{r}_{\max}]$ of all case members in the cluster. In this work, we assume that every case in the case base belongs to every cluster $C_1, ..., C_n$. Thus, a fuzzy mean regions $\bar{k}_i$ of cluster protype c_i can be computed as

$$\bar{k}_i = \left(\tilde{r}^{(j)}_{\min} + \tilde{r}^{(j)}_{\max}\right)/2, \quad j = 1, ..., n \tag{4}$$

where $[\tilde{r}^{(j)}_{\min}, \tilde{r}^{(j)}_{\max}]$ is a *fuzzy solution* of case member j. *Fuzzy solution* for each case can be derived by weighting its original solution $[r_{\min}, r_{\max}]$ with its corresponding membership value as

$$\tilde{r}^{(j)}_{\min} = \frac{\sum_{j=1}^{n} u_{ij} r^{(j)}_{\min}}{\sum_{j=1}^{n} u_{ij}} \quad i = 1, ..., c; j = 1, ..., n$$

$$\tilde{r}^{(j)}_{\max} = \frac{\sum_{j=1}^{n} u_{ij} r^{(j)}_{\max}}{\sum_{j=1}^{n} u_{ij}}, \quad i = 1, ..., c; j = 1, ..., n$$

The new deriving solution $[\tilde{k}_{\min}, \tilde{k}_{\max}]$ will be given to the segmentation combination algorithm for combining a given segmentation ensemble. Once $\tilde{k}_{\max} - \tilde{k}_{\min} + 1$ combination results were computed, the optimal segmentation selection method will select the optimal segmentation among them to produce the final result.

4. Experiments

We first describe the dataset, the segmentation ensemble framework and the evaluation method used in the experiments. Then the series of experimental results are reported and discussed.

4.1 Experimental Settings

Image Dataset

The natural images with human segmentations from the Berkley segmentation dataset [Martin 01] are used. The dataset consists of 300 color images of size 481×321 or vice versa, each having multiple manual segmentations. The human segmentations are used only for evaluating segmentation results produced by our approach.

Automatic Evaluation of the Segmentation Results

We follow the concept of mutual information to quantify the statistical information shared between two clusterings. The quality of combination results can be evaluated in terms of consistency with the ground truth images. The normalized version of mutual information, $\phi^{(\mathrm{NMI})}$, which is used to quantitatively evaluate the segmentation quality against the ground truth, is defined as

$$\phi^{(\mathrm{NMI})}(S_a, S_b) = \frac{\sum_{h=1}^{|S_a|} \sum_{l=1}^{|S_b|} |R_{h,l}| \log \frac{n \cdot |R_{h,l}|}{|R_h| \cdot |R_l|}}{\sqrt{\sum_{h=1}^{|S_a|} |R_h| \log \frac{|R_h|}{n} \sum_{l=1}^{|S_b|} |R_l| \log \frac{|R_l|}{n}}}$$

where R_h and R_l are regions from S_a and S_b, respectively, $R_{h,l}$ denotes the common part of R_h and R_l, and n is the image size. The value domain of $\phi^{(\mathrm{NMI})}$ is $[0, 1]$. Larger NMI values indicate better segmentation that shares more information with the ground truths, and thus can be considered as higher quality. When there are more than one ground truth per input image, one segmentation result is compared to all manual segmentations and the average normalized mutual information (ANMI) is reported.

$$\phi^{(\mathrm{ANMI})}(\hat{S}, \Lambda) \;=\; \frac{1}{N} \sum_{q=1}^{N} \phi^{(\mathrm{NMI})}(\hat{S}, S_q)$$

Segmentation Ensemble Generation

An ensemble $\Lambda = \{S_1, ..., S_N\}$ of N segmentations of the same image can be produced in several ways, for example, by using different algorithms or with the same algorithm but different parameter values. In our experiments multiple segmentations are obtained by varying the parameter values of the same segmentation algorithm in an appropriate range. In this work we use the efficient graph-based image segmentation algorithm [Felzenszwalb 04] for segmenting an image because of its competitive segmentation performance and high computational efficiency. The algorithm has three parameters: smoothing parameter (σ), a threshold function (k), and a minimum component size (min_area). We obtained 24 segmentations of an image by varying the parameter values (fixing $min\_area = 1500$ and varying $\sigma = 0.4$, 0.5, 0.6, 0.7, 0.8, 0.9 and $k = 150$, 300, 500, 700). The sampled values of parameters within these ranges are chosen so as to yield segmentations with perceptible differences and acceptable average quality.

Random Walker-Based Segmentation Combination Algorithm

In this study we used the random walker-based segmentation combination algorithm proposed in [Wattuya 08]. The reasons are (i) the algorithm, in contrast to the few early works, is able to work with the most general class of segmentation combination (i.e. each input segmentation can have an arbitrary number of regions, which is a fundamental property required for computing the case base solution), (ii) it is efficient enough to be capable of evaluating a series of possible combination results with different k regions. (iii) it allows us to choose methods to select an optimal segmentation (from a set of combination output) independently from the combination algorithm. Methods for selecting optimal segmentation will be discussed in following subsection.

The segmentation combination algorithm begins with automatically extracting k seeds (corresponding to k image regions) from Λ. Pixels with high co-association values among

N initial segmentations are likely to be selected. Given a set of seeds and a graph G constructed from the initial segmentations (by using co-association values as weights of graph edges), the random walker algorithm performs the calculation by assigning to each pixel a k-tuple vector that specified the probability (i.e., co-association values) that a random walker starting from each unseeded pixel will first reach each of the k seeds. A final segmentation is derived from these k-tuples by assigning each pixel the label of the largest probability. Full algorithm details are given in [Wattuya 08].

Automatic Optimal Segmentation Selection

The best segmentation in the set of combination results can be either judged by human or automatic evaluated by optimization methods. In this work we choose the MDL-based selection approach proposed in [Wattuya 10] to automatically select an optimal segmentations. The major reason is that the computation in the MDL-based approach is independent of the quality of initial segmentations in an ensemble. The generalized median-based selection approach proposed in [Wattuya 08] is able to achieve its fully performance in the situation where more than half of initial segmentations in an ensemble have good quality. When majority population in an ensemble is of poor quality, it degrades the power of the approach. This may be one of the causes why the accuracy of the results produced by the MDL-based approach is slightly better than the accuracy of the results produced by the generalized median-based approach [Wattuya 10].

The MDL-based selection method [Wattuya 10] was adapted from the MDL-based objective criterion for image segmentation proposed by Rao et al. [Rao 09]. The criterion is based on the minimum description length principle to encode both the texture and boundary information of a natural image and defines the optimal segmentation of an image as the one that minimizes its total coding length.

Adaptive Texture Encoding: Rao et.al construct texture vectors that represent homogeneous textures in image segments as follows. Let the w-neighborhood $\mathcal{W}_w(p)$ be the set of all pixels in a $w \times w$ window centered at pixel p. They construct a set of features X by taking the w-neighborhood around each pixel in I across the three color channels, and then stacking each window as a column vector:

$$X = \left\{ x_p \in \Re^{3w^2} : x_p = \mathcal{W}_w(p)^S \text{ for } p \in I \right\}.$$

For ease of computation, they reduce the dimensionality of these features by projecting the set of all features X onto their first D principal components. They denote the set of features with reduced dimensionality as $\hat{X}$ and choose to assign $D = 8$. Subsequently, the texture information is encoded using a Gaussian distribution. First Rao et.al consider a single region R with N pixels. For a fixed quantization error ϵ, the expected number of bits needed to code the set of N feature window $\hat{X}$ up to distortion ϵ^2 is given by:

$$L_{w,\epsilon}(R) = (\frac{D}{2}+\frac{N}{2w^2})\log_2 \det(I+\frac{D}{\epsilon^2}\hat{\Sigma}_w)+\frac{D}{2}\log_2(1+\frac{\|\hat{\mu}_w\|^2}{\epsilon^2}). \quad (5)$$

Adaptive Boundary Encoding: Rao et.al apply a well-known scheme, the *Freeman chain code*, for representing boundaries of image regions. In this coding scheme, the orientation of an edge is quantized along eight discrete directions. Let $\{o_t\}_{t=1}^{T}$ denote the orientations of the T boundary edges of R. Since each chain code can be encoded using three bits, the coding length of the boundary of R is

$$B(R) = 3\sum_{i=0}^{7} \sharp(o_t = i).$$

Given the prior distribution $P[\Delta o]$ of difference chain codes, $B(R)$ can be encoded more efficiently using a lossless Huffman coding scheme:

$$B(R) = -\sum_{i=0}^{7} \sharp(\Delta o_t = i)\log_2(P[\Delta o = i]). \quad (6)$$

Minimizing Coding Length: Suppose an image I can be segmented into non-overlapping regions $\mathcal{R} = R_1, ..., R_k, \cup_{i=1}^{k} R_i = I$. Based on the coding length functions developed in (5) and (6), the total coding length of the image I is

$$L_{w,\epsilon}^{S}(\mathcal{R}) = \sum_{i=1}^{k} L_{w,\epsilon}(R_i) + \frac{1}{2}B(R_i). \quad (7)$$

Note that the boundary term is scaled by a half because we only need to represent the boundary between any two regions once. The optimal segmentation of I is the one that minimizes equation (7).

4.2 Experimental Results

We have conducted a series of experiments to verify the effectiveness of our approach in 3-fold cross-validation. The original data set was randomly partitioned into three mutually exclusive subsets, each contains 100 images. Of the K subsets, a single subset is retained as the test set for testing our case-base reasoning, and the remaining $K-1$ subsets are used as training data (for building a case base). The cross-validation process is repeated K times, with each of the K subsets used exactly once as the testing data on which the final performance evaluations are carried out. The K results from the folds are reported as well as averaged value for an overall performance. In the experimental reports, we will refer to the three subsets as '*testset1*', '*testset2*' and '*testset3*', respectively.

Traditional Image Segmentation Combination Approach

This approach is used as a baseline segmentation combination framework to verfy our approach. The steps of computing the final combination result in the original image segmentation combination framework [Wattuya 08] are as follows:

Table 1: Average performance measures of 'Full-Space-Search' and '*Optimal-Subspace-Search*' approaches.

Test set	*Full-Space-Search* ANMI	*Optimal-Subspace-Search* ANMI	T
1	.6112(±.1430)	.6066(±.1377)	1.70
2	.6107(±.1347)	.6079(±.1329)	1.77
3	.6283(±.1382)	.6187(±.1361)	1.66
Avg.	0.6167 (±.1386)	.6111 (±.1356)	1.71

Step 1. For each image in each test set, a 24-segmentation ensemble is generated by applying 24 parameter combinations to the efficient graph-based image segmentation algorithm.

Step 2. For each ensemble in each test set, the random walker-based segmentation combination algorithm is run $k_{\max} - k_{\min} + 1$ rounds to produce a series of $k_{\max} - k_{\min} + 1$ combination results.

Step 3. The MDL-based segmentation selection algorithm is used to search for an optimal final result as follow:

- For each combination result in the search space Compute the MDL-value according to (7).
- The optimal segmentation combination is given by the one that minimizes the objective function.

The algorithm needs to run, in total, $k_{\max} - k_{\min} + 1$ interations in Step 2 and 3. We will refer to this number as the size of search space T, $T = k_{\max} - k_{\min} + 1$, and refer to this approach as *Full-Space-Search* for reporting experimental results. The range $[k_{\min}, k_{\max}]$ is set to [2, 50], the same as in [Wattuya 08].

The average performance of 'Full-Space-Search' for all three test sets are reported in Table 1 column 2. We did not put values of T for this approach in the table, since it is equal to 49 for all cases.

Integrating with Soft Case-Based Reasoning Approach

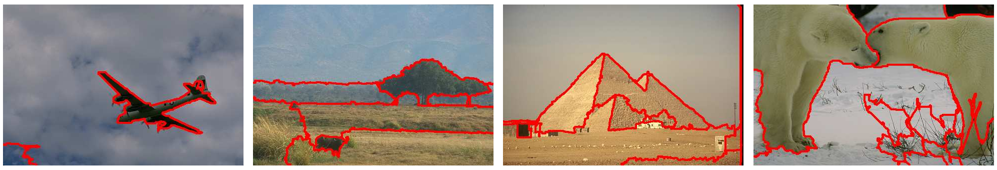
By integrating our case-based reasoning into the original framework, it can reduce the search space T to $\tilde{k}_{\max} - \tilde{k}_{\min} + 1$. We will refer to this approach as *Optimal-Subspace-Search* in the experimental reports. In the experiment, we empirically set the number of clusters to 50. However, validity criteria, such as [Halkidi 02], can be applied to search for optimal clusterings which may improve the accuracy of the current results (remains future work). For each test set we performed 50 runs of the FCM algorithm with random initialization of cluster centers, and retained only the results that corresponded to the best results (maximum ANMI value attained) for comparison purposes, however, a small variance on the performance over these 50 experiments was observed.

Figure 1 shows some examples of model order selection results produced by our method. The resulting $[\tilde{k}_{\min}, \tilde{k}_{\max}]$ are shown under each of its input image. Note that all results have $\tilde{k}_{\min} = \tilde{k}_{\max}$, where the optimal segmentation selection is not required. The segmentation results shown side by side with the solution of the case-base are for visual evaluation of the obtained results, in comparison with the image contents. The segmentation lines are superimposed on the original image for visualization. As shown in the figure, our case-based reasoner can predict a reasonable range of k, based on the underlying assumption. Case-based reasoner gives relatively high $\tilde{k}_{\min}$, $\tilde{k}_{\max}$ values to input images with high variation in intensity values, and gives relatively low $\tilde{k}_{\min}$, $\tilde{k}_{\max}$ values to input images with nearly constant regions. It is also interesting to note that there are also the situations that the case-based reasoner can predict a reasonable range of k. Unfortunately, either a segmentation or a segmentation combination algorithm fail to obtain meaningful regions, which degrades our formulation suboptimal.

Column 3 and 4 in Table 1 summarizes the average performances and the average T obtained over 100 images of three test sets. A case-based reasoner can deliver a very tight ranges $[\tilde{k}_{\min}, \tilde{k}_{\max}]$ according to our expectation, which substaintialy reduces the search space, namely, from $T = 49$ to $T = 1.71$ in average over all three test sets, or approximately 29 times reduction. However, the average accuracy is slightly degraded. Two possible causes of degradation are due to (i) the estimation errors in our method, for example, by using the mean values (i.e. fuzzy mean region) for estimating the solution $[\tilde{k}_{\min}, \tilde{k}_{\max}]$. (ii) the small body of knowledge, namely, the cases stored in the case base is too small to cover the problem space. An effective way to solve the first problem is by case base solution relaxation (described below) and to solve the latter problem by case learning, which will be discussed later in this section.

In this work, we propose a simple strategy to solve the degradation caused by the estimation error in our method by using a relaxation theshold, τ, to relax the very tight range. Thus, the range $[\tilde{k}_{\min}, \tilde{k}_{\max}]$ becomes $[\tilde{k}_{\min} - \tau, \tilde{k}_{\max} + \tau]$. Table 2 shows the extension results of the results reported in Table 1 for $\tau = 1$, 2, 3 and 4, respectively. The average performance whose value higher than the average performance of *Full-Space-Search* are highlighted in bold (i.e. $\tau = 4$ for *testset1*, $\tau = 2$ for *testset2*, $\tau = 3$ for *testset3*, with $\tau = 3$ for overall performance). It is interesting to note that performance of *'Optimal-Range* is more stable than performance of *Full-Space-Search* in all cases (We will discuss about stability of the approaches later in this section.). As we expected, the average performances of *Optimal-Subspace-Search* improves as τ increases. However, it is important to note that it is not necessary that increasing τ will monotonically increases the average performance. The average performance can fluctuate along the way and, eventually, converges to the average performance of *Full-Space-Search* when the optimal range $[\tilde{k}_{\min}, \tilde{k}_{\max}]$ is expnded to the full range $[k_{\min}, k_{\max}]$. As suggested by the experimental results, the value of $\tau \in [1, 4]$ yields satisfactory performance, in terms of both accuracy and T. The last row of Table 2 shows the average performance and the average T over the three test sets. *Optimal-Subspace-Search* can overcome *Full-Subspace-Search* when $\tau = 3$ and can significantly reduce T from 49 to 3.71, 5.71, 7.71 and 9.10 or reduction approximately by 13, 8, 6 and 5 times, for $\tau = 1, 2, 3$ and 4, respectively, while preserves the quality

,



ANMI = 0.5765	ANMI = 0.7846	ANMI = 0.7238	ANMI = 0.7211
$k = 5 \in [5, 5]$	$k = 8 \in [8, 8]$	$k = 9 \in [9, 9]$	$k = 13 \in [13, 13]$

,

ANMI = 0.6565	ANMI = 0.5580	ANMI = 0.6052	ANMI = 0.3378
$k = 16 \in [16, 16]$	$k = 19 \in [19, 19]$	$k = 20 \in [20, 20]$	$k = 21 \in [21, 21]$

Figure 1: Example of segmentation results of the *Optimal-Subspace-Search*. The quantities under images are ANMI value, number of regions, and the optimal range $[\tilde{k}_{\min}, \tilde{k}_{\max}]$ computed by our proposed CBR.

of the results.

Comparison Discussion

We begin a comparison discussion by argueing that there is high consistency between the two approaches supported by the results. Figure 2 shows example segmentation results that were, exactly the same, selected by both *Optimal-Subspace-Search* (with $\tau = 1$) and *Full-Space-Search* approaches. The images with underlying descriptions under them indicates that they were also selected by *Optimal-Subspace-Search* without relaxation ($\tau = 0$). This is somewhat surprising. However, selecting the same results as *Full-Space-Search* approach did not necssarily means that we select the correct (optimal) results. The performance of *Full-Space-Search* depends heavily on the effectiveness of the used MDL-based objective function, which sometimes leads to very poor performance. This is possibly due to the instability of the objective function. Becuase of the large degree of variability and complexity encountered in real-world images, it is quite impossible to understand completely the unpredictable behaviour of the objective function (e.g. what factors that lead to the succces/failure of a solution). Some example results in this situation are shown in Figure 3. The first row (comparing along side on the left the results of *Optimal-Subspace-Search*) shows example results where the MDL-based objective function selects poor results with too high number of regions, and the second row shows selected results with too low number of regions. It is quite obvious that our case-based reasoning is substaintially profitable in this situation. It indirectly helps the objective function not to select unpleasant results by limiting the search in the whole search space to the potential subspace. In other words, it filters out the weak segmentations from a consideration, while leaving only the potential solutions to be selected. This situation shows a valuable side benefit of our case-based reasoning in alleviating a performance degradation caused by instability of objective function.

Given the fact that we do not know the optimal number of regions for a particular image in advance, the comparative performance of our approach is remarkable and reveals its potential in dealing with the difficult model order selection without ground truth.

Case Learning and Case-base Maintenance

As we mentioned earlier the performance degradation caused by the small body of knowledge. This problem can be solved by case learning. One of the key successes of a problem-solving case-based reasoning is its *coverage* of the problem space in the target domain. The case-based reasoning system should contain all the essential cases that could be used to generate solutions to all possible current cases (i.e. $[k_{\min}, k_{\max}]$ in our domain problem). As case-based reasoning systems are used, they encounter a wider range of problem situations. Thus, case learning must be performed to help in solving future problems. A simple learning method is the addition of a new problem and its solution to the case base. The new cases must be tested and determined its successfulness level for solving the problem before added into the case base. However, the addition of cases may not necessarily improve the coverage of a case base. Thus, additional primary tasks, such as identification of the essential cases and missing knowledge, are needed.

As cases increase, the greater the problem space covered. However, too many cases stored in a case base will degrade retrieval efficiency. Thus, case-base maintenance, such as redundant case deletion, is needed. Case learning and case-base maintenance remain our future work.

5. Conclusions

Our contribution of this work is to investigate the usefulness of a soft case-based reasoning for solving the model order selection problem. Our proposed case-based reasoning has been verified in an application of image segmentation combination. Preliminary experiments show good results of our approach that can predict a shortlist of potential number of regions. The benefits of integrating the case-based reasoning into original segmentation combination framework not only significantly reduce the search space without degrading the accuracy performance, but also

Table 2: Average performance measures of *Optimal-Subspace-Search* with $\tau = [1, 4]$.

Test	$\tau = 1$		$\tau = 2$		$\tau = 3$		$\tau = 4$	
set	ANMI	T	ANMI	T	ANMI	T	ANMI	T
1	.6082($\pm$.1382)	3.70	.6086($\pm$.1381)	5.70	.6105($\pm$.1380)	7.70	**.6115**($\pm$.1394)	9.70
2	.6103($\pm$.1311)	3.77	**.6124**($\pm$.1312)	5.77	.6117($\pm$.1311)	7.77	.6120($\pm$.1289)	9.77
3	.6259($\pm$.1352)	3.66	.6273($\pm$.1344)	5.66	**.6300**($\pm$.1335)	7.66	.6303($\pm$.1344)	9.62
Avg.	.6148($\pm$.1348)	3.71	.6161($\pm$.1345)	5.71	**.6174**($\pm$.1342)	7.71	.6179($\pm$.1342)	9.70

significantly improve the accuracy of particular results affected by the instability objective function used in optimal segmentation selection method. Novel feature of our proposed case-based reasoning is the ability to predict the optimal range of k without the need of ground truth segmentations, which are generally not available in practice. The focus of our current work is image segmentation combination. It should be mentioned that our case-based reasoning is a general framework and can be integrated to a general class of segmentation combination system.

Acknowledgments. This work is supported by Thailand Research Fund under Grant no. MRG5480193

References

[Bezdek 81] Bezdek, J. C.: Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press (1981)

[Cardaci 06] Cardaci, M., Gesú V. D., Petrou, M., and Tabacchi, M. E.:On the evaluation of images complexity: A fuzzy approach, *Lecture Notes in Artificial Intelligence*, Vol. 3849, pp. 305–311 (2006)

[Cardaci 09] Cardaci, M., Gesú, V. D., Petrou, M., and Tabacchi, M. E.: A fuzzy approach to the evaluation of image complexity, *Fuzzy Sets and Systems*, Vol. 160, No. 10, pp. 1474–1484 (2009)

[Cho 97] Cho, K. and Meer, P.: Image segmentation from consensus information, *Computer Vision Image Understanding*, Vol. 68, No. 1, pp. 72–89 (1997)

[Felzenszwalb 04] Felzenszwalb, P. and Huttenlocher, D.: Efficient graph-based image segmentation, *International Journal of Computer Vision*, Vol. 59, No.2, pp. 167–181 (2004)

[Frucci 08] Frucci, M., Perner, P., and Sanniti, G.:Case-based-reasoning for image segmentation, *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 22, No. 5, pp. 1–14 (2008)

[Halkidi 02] Halkidi, M., Batistakis, Y., and Vazirgiannis, M.: Cluster validity methods: Part I and part II, *SIGMOD Record*, Vol. 31, No. 2, pp. 19–27 (2002)

[Halkidi 73] Haralick, R. M., Shanmugam, K., and Dinstein, I.: Textural features for image classification, *IEEE Transactions on System, Man, and Cybernetics*, Vol. 3, No. 11, pp. 610–630 (1973)

[Martin 01] Martin, D. R., Fowlkes, C., Tal, D., and Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in *Proceesdings of the International Conference on Computer Vision*, pp. 416–425 (2001)

[Pal 04] Pal, S. K. and Shiu, S. C. K.: Foundations Of Soft Case-based Reasoning. John Wiley & Sons (2004)

[Perner 99] Perner, P.: An architecture for a cbr image segmentation system, *Journal on Engineering Applications in Artificial Intelligence*, Vol. 12, No. 6, pp. 749–759 (1999)

[Rabinovich 06] Rabinovich, R., Belongie, S., Lange, T., and Buhmann, J. M.: Model order selection and cue combination for image segmentation, in *Proceedings of Computer Vision and Pattern Recognition - Volume 1*, pp. 1130–1137 (2006)

[Rao 09] Rao, S., Mobahi, H., Yang, A., Sastry, S., and Ma, Y.: Natural image segmentation with adaptive texture and boundary encoding, in *Proceedings of the 9th Asian Conference on Computer Vision*, pp. 135–146 (2009)

[Senthilkumaran 09] Senthilkumaran, N. and Rajesh, R.: Image segmentation - a survey of soft computing approaches, in *International Conference on Advances in Recent Technologies in Communication and Computing*, pp. 844–846 (2009)

[Wattuya 10] Wattuya, P.: Combination of Multiple Segmentations, PhD thesis, University of Münster, Germany (2010)

[Wattuya 08] Wattuya, P., Jiang, X., Prassni, J., and Rothaus, K.: A random walker based approach to combining multiple segmentations, in *Proceedings of the International Conference on Pattern Recognition*, pp. 1–4 (2008)

,

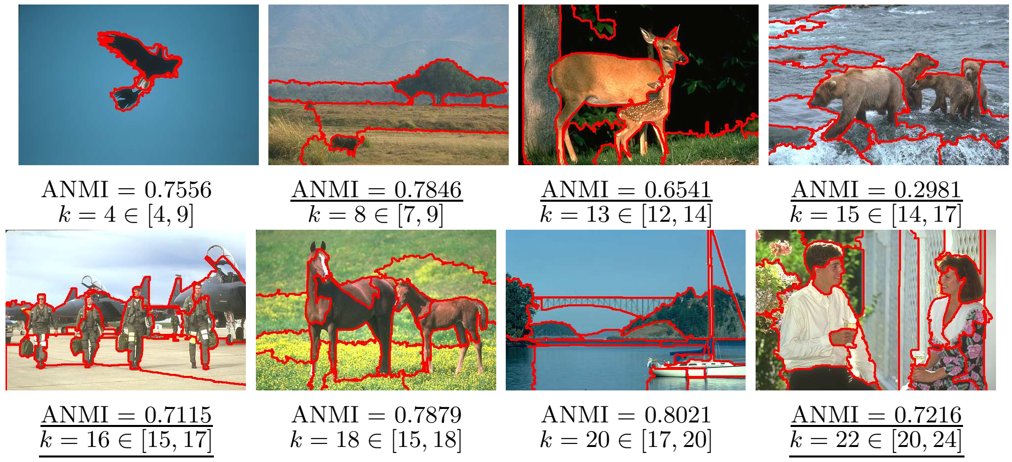


Figure 2: Example of segmentation results: the *Optimal-Subspace-Search* (with $\tau = 1$) selects the same results as *Full-Space-Search*. The quantities under images are ANMI value, number of regions, and the optimal range $[\tilde{k}_{\min}, \tilde{k}_{\max}]$ computed by our proposed CBR. The segmentation is superimposed on original image.

,

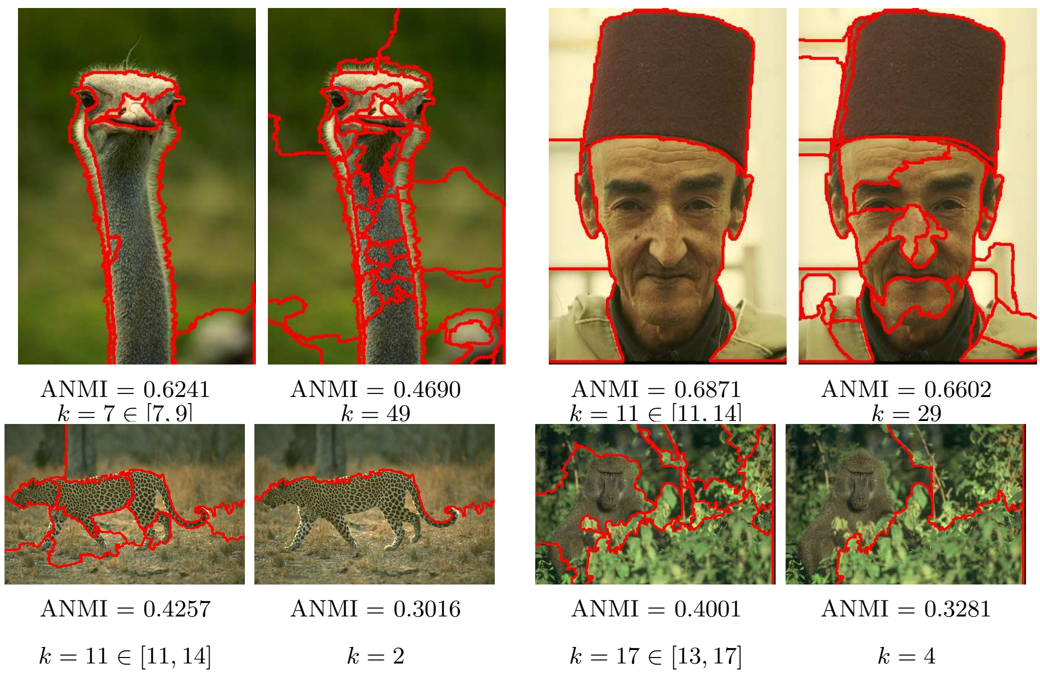


Figure 3: Example of segmentation results: *Optimal-Subspace-Search* (with $\tau = 1$) selects better results (on the left) than *Full-Space-Search* (on the right). The quantities under images are (on the left) ANMI value, number of regions, and the optimal range $[\tilde{k}_{\min}, \tilde{k}_{\max}]$, and (on the right) ANMI value and number of regions.