

A Generic Knowledge-Guided Image Segmentation and Labeling System Using Fuzzy Clustering Algorithms

Mingrui Zhang, Lawrence O. Hall, *Senior Member, IEEE*, and Dmitry B. Goldgof, *Senior Member, IEEE*

Abstract—Segmentation of an image into regions and the labeling of the regions is a challenging problem. In this paper, an approach that is applicable to any set of multifeature images of the same location is derived. Our approach applies to, for example, medical images of a region of the body; repeated camera images of the same area; and satellite images of a region. The segmentation and labeling approach described here uses a set of training images and domain knowledge to produce an image segmentation system that can be used without change on images of the same region collected over time. How to obtain training images, integrate domain knowledge, and utilize learning to segment and label images of the same region taken under any condition for which a training image exists is detailed. It is shown that clustering in conjunction with image processing techniques utilizing an iterative approach can effectively identify objects of interest in images. The segmentation and labeling approach described here is applied to color camera images and two other image domains are used to illustrate the applicability of the approach.

Index Terms—Classification, color image processing, expert system, knowledge extraction.

I. INTRODUCTION

ONE of the important problems in a vision system is identifying the image regions that represent objects. This operation, which is so natural and easy for people, is surprisingly difficult for computers [1]. The first step toward a vision system for object identification is to partition or segment an image into meaningful regions. Ideally, a region represents an object or part of an object. A variety of segmentation techniques have been investigated [2]–[5]. Several conclusions were drawn from these efforts. The first was that supervised segmentation methods were subject to intra- and inter-observer variations. The variations were, for example, highly dependent in magnetic resonance images, on the tissue being investigated, segmentation method, and the imaging parameters. Another problem with supervised methods was that selecting good training examples could be time consuming. Unsupervised techniques based upon the fuzzy c -means (FCM) clustering algorithms had shown great promise in magnetic resonance (MR) image segmentation.

Manuscript received February 9, 2001; revised January 14, 2002. This paper was recommended by Associate Editor V. Govindaraju.

M. Zhang is with the Department of Computer Science, Winona State University, Winona, MN 55987 USA (e-mail: Zhang.Mingrui@winona.msus.edu).

L. O. Hall and D. B. Goldgof are with the Department of Computer Science and Engineering, University of South Florida, Tampa, FL 33620 USA (e-mail: hall@csee.usf.edu; goldgof@csee.usf.edu).

Publisher Item Identifier S 1083-4419(02)04992-0.

Clustering is a fundamental approach in pattern recognition. Given a finite set of data, X , the problem of clustering in X is to partition the data into c relevant classes of X . In classical cluster analysis, these classes are required to form a partition of X such that the degree of association is strong for data within a class and weak for data in different classes. One of the clustering methods is the FCM [6] algorithm, which creates a set of c class centroids and allows examples to partially belong to each one. The FCM algorithm can be utilized to partition the data into c classes.

The cluster centroid [6] produced from the FCM algorithm is a description of a fuzzy cluster. A centroid is represented by a set of image features, such as the intensity value of each color spectrum from an image. The set of all possible centroids forms a decision space. Similarity of clusters results in “close” proximity of their centroids in the feature space. For a color image composed of red, green, and blue color components, each component image is considered as a feature image in the process of clustering. In this case, the formed feature space is three-dimensional (3-D) and uses the same metrics as each color component. One can process the image to calculate a local texture feature, such as entropy, and then add the additional feature to the clustering process. Extraction of rules from the feature space of cluster centroids, and thereafter the labeling of clusters, can be accomplished by different approaches. Mascarilla [7] proposed an approach to extract fuzzy rules and eliminate the redundancy among the rules by eliminating redundant image features. The author investigated the use of fuzzy mutual information criterion (MIFS) to select a subset of image features.

A wide variety of image processing algorithms have been devised in the history of digital image processing. Their utility has been proven in various application areas, such as medical image processing, remote sensing, and computer vision. Heath and Sarkar *et al.* [8] investigated and presented a paradigm, which was based on experimental psychology and statistics, to gather human ratings of the results of low-level vision algorithms. They used this methodology to compare the performance of four edge detectors, namely, Canny, Nalwa-Binford, Sarkar-Boyer, and Sobel. One of the conclusions they reached was that the optimal parameter settings of an edge detector were dependent upon the image. The dilemma which remains is how to effectively choose different algorithms and parameters for them to carry out image processing tasks. The knowledge-guided image processing system may provide an answer.

Knowledge-based medical image segmentation and/or interpretation has been investigated by several researchers [11]–[15]. In [15], a knowledge-guided image thresholding approach was

presented to automatically extract soft tissues from multispectral brain MRI images and display their 3-D shapes. The thresholding was adjusted for each slice of the image. A dynamic thresholding process was used in processing computerized tomography (CT) images for organ identification [13]. The location, size, adjacency, and shape constraints of the organs were used as anatomical knowledge. Other than medical image segmentation and interpretation, McKeown and Harvey *et al.* [16] developed a rule-based aerial image understanding system for airport scenes. It contained an image/map database as its knowledge source. In their system, all extracted image features were first transformed into the map space, in which all reasoning for object recognition was performed.

This paper presents a generic knowledge-guided segmentation and labeling (KGSL) approach that, after training, is capable of automatically segmenting and analyzing previously unseen multifeature images. The KGSL methodology uses an unsupervised fuzzy clustering algorithm along with basic image processing techniques under the guidance of a knowledge base. In the approaches described in [11]–[14], knowledge was acquired and “hard-coded” with human intervention. The KGSL approach can be applied to automatically process the ground-truth images and extract cluster labeling rules. The cluster labeling rules are stored in the system’s knowledge base. The approaches presented in [17] and [18] were successful in their chosen areas, but they were focused on system implementations for a specific domain area and lack generalization. This paper describes a generic approach to segment images taken at a fixed location using clustering as a base algorithm. Different types of knowledge and their utilization are discussed. Furthermore, the use of machine learning mechanisms, such as automatic rule extraction and knowledge base refinement, are discussed. The knowledge base can be refined later by adding or removing cluster labeling rules.

The refinement is especially useful for applications such as the identification of brain tumors with MRI and monitoring the environmental changes using satellite imagery. For both applications, the acquisition of ground-truth images will be over a (long) period of time. Whenever new ground-truth images with interesting features are available, the KGSL approach can be applied to segment the images, automatically extract cluster labeling rules and add them to the knowledge base to improve its performance. The promise of the approach was demonstrated by processing a set of color images and ocean satellite images.

The remainder of the paper is structured as follows. Section II introduces the overall structure of the approach. Section III describes the experiment to collect a set of color images. Section IV discusses the types of knowledge used in the approach. We describe the construction of a knowledge base for the system in Section V, and refinement of the knowledge base in Section VI. In Section VII, we will show the application of the approach to a set of color images taken outdoors at the same location over a period of time and give a discussion of the application of the approach in other domains. Section VIII is a summary of results.

II. A KNOWLEDGE-GUIDED SYSTEM

Fuzzy clustering algorithms are useful in the early stages of a segmentation system, where they may be used to group pixels of

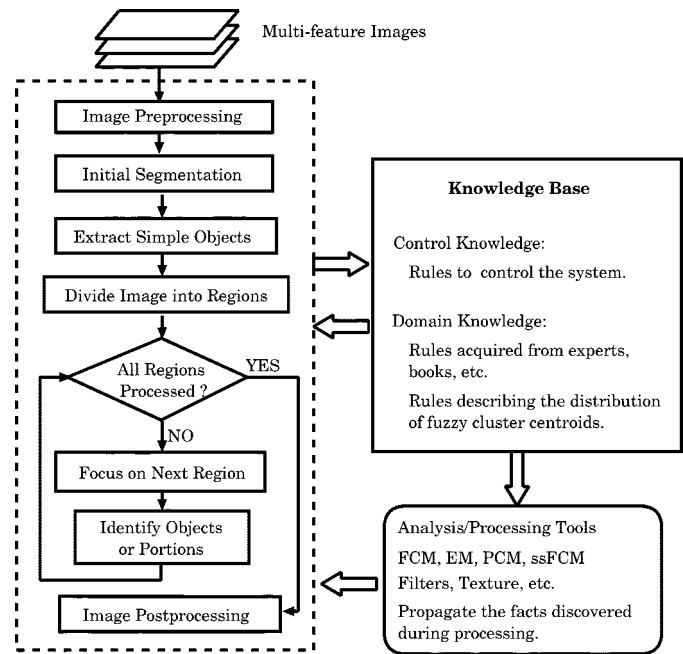


Fig. 1. Control structure of a knowledge-guided segmentation and labeling approach.

the image into “meaningful” clusters. Extraction of rules from the feature space formed with cluster centroids and thereafter the labeling of clusters may be accomplished later. The KGSL approach performs two types of tasks: 1) *segmentation*, to extract meaningful regions from an image; and 2) *identification*, to establish the correspondence between image regions and object(s) searched for. Segmentation is performed by fuzzy clustering. The membership of classes is useful information; any clustering algorithm [19]–[21] that provides fuzzy or probabilistic information will suffice.

Fig. 1 illustrates the control structure of the KGSL approach. A KGSL approach begins the initial segmentation using a fuzzy clustering algorithm and extracts clusters corresponding to easily detectable objects or portions of objects. An easily detectable object is defined as an object that can be segmented successfully using a one-step segmentation method. On the contrary, the detection of objects with less prominent appearance requires accurate *a priori* information about their properties: areas, locations, etc. Cluster labeling rules may be used to label clusters belonging to the easily-detectable objects if a FCM algorithm is used. The concept of easily-detectable objects can be used in many image processing applications, such as remote sensing and medical image processing. Geographic features like buildings and roads will remain in a time series of satellite image of the same location. The human skull and ventricles can be identified first on MR brain images of different patients. The relative locations of the objects in the image do not change.

Based on the extracted objects, it is necessary to set up a geometric frame of reference relative to them with the whole image then divided into subimages. The division of the whole image into subimages enables the system to focus on a subimage and to apply different algorithms or to choose different parameters for the same algorithm when applied to the different subimages. Each subimage is focused upon by the system. In a process

of clustering, identifying objects and reclustering, a complete scene description is gradually constructed. Different processing algorithms will be used. The process is controlled under the guidance of the knowledge base in the KGSL approach.

The idea of *focus-of-attention* can also be achieved through re-clustering, when a fuzzy clustering algorithm is used. The basic idea behind knowledge-based (re-)clustering is to iteratively cluster, identify the regions of interest and recluster. The term “reclustering” means that the whole image is first clustered by a fuzzy clustering algorithm, then all clearly identified objects or portions of objects are extracted and labeled. The remaining pixels are to be reclustered. This approach is necessary because different classes may be very close in some features and have a tendency to be segmented into mixed classes. Through reclustering, we can deliberately separate similar classes.

The use of knowledge clearly delineates the KGSL approach from statistical pattern recognition, in which all processing is done purely based on attributes of objects. Knowledge about the scene is classified into two types, namely, control knowledge and domain knowledge. The control knowledge guides the use of other knowledge utilized by the system and is not directly applicable to the recognition of the objects. It may include the rules to determine the best image processing algorithm along with its parameters, what objects should be detected/searched for first, which subimage to be analyzed first, and to control the (re-)clustering, etc. The effective use of domain knowledge, such as where or which domain knowledge should be applied, relies upon the control knowledge of the system.

Part of the domain knowledge can be acquired by interviewing field experts, or it can be obtained from a published book, map, etc. It is an important part of the knowledge base in the KGSL approach, and is represented by rules. This aspect of the approach can be viewed as setting up an “expert system” shell for image analysis [22]. Because the KGSL approach uses a fuzzy clustering algorithm as an analysis tool, the distribution of the resultant fuzzy cluster centroids become another important knowledge source for the system. The cluster labeling rules are extracted from the feature space formed with fuzzy cluster centroids of training images. Successful extraction of cluster labeling rules is critical to the success of a KGSL approach.

III. EXPERIMENT ON SEGMENTING COLOR PHOTOGRAPHS

To explain the general ideas behind the approach presented here, we implemented and applied a KGSL system to 93 color images collected outdoors. As we proceed to the latter sections of this paper, these color photos will be used as an explanation aid. The color photos were collected using a Cannon Elan II camera and Kodak 35-mm color film. The camera’s position was fixed throughout photo collection. Photos were collected under different light conditions: morning, noon and afternoon; sunny, cloudy, and rainy (see Fig. 2). We have included a color MPEG attachment which contains four images taken at different times under different conditions. This will be available as an attachment within IEEE Xplore (<http://ieeexplore.ieee.org>).

Each photo was scanned by a color scanner, HP ScanJet 4C, with the color resolution of “millions of colors” and size 500×390 pixels. The digitized color photographs originally



Fig. 2. Sample of color image printed in gray level.

have three specific color spectrums: red (R), green (G), and blue (B). The transformation from RGB to hue (H), saturation (S) and intensity (I) was used to convert all color images into an appropriate form for this study [23]. All 93 images were numbered and divided into two groups: 1) a training set and 2) a testing set. The implementation of a KGSL system started with 45 training images. Objects in the training images were identified and manually outlined. The rest of images (48) were left unstudied and used for testing the system performance.

Shadows are unavoidable in photos taken outdoors under natural light conditions. Even though it is not difficult for humans to distinguish shadows from objects, identifying shadows by computer involves solving many difficult problems. Several methods have been developed for identifying shadows [25], [24]. Nagao and Matsuyama *et al.* [25] used the histogram of an image to identify shadows in aerial images. They assumed that the histogram is bimodal, with the valley between the two peaks in the histogram chosen as a threshold to determine shadows. Similarly, we computed the histogram of the intensity (I) of the color image for each subimage. The shadow and unshadowed pixels are clustered separately using a fuzzy clustering algorithm after thresholding.

Certainly, fuzzy clustering algorithms can be used instead of histogram thresholding to separate shadow pixels. However, fuzzy clustering, which requires heuristic rules to label clusters, is less efficient in time than a simple histogram thresholding for this purpose. This step can be viewed as a fast “gross” clustering/grouping. Histogram thresholding is applied to each subimage before any fuzzy clustering algorithm is applied. Even if there is no shadow in the image (e.g., photos taken on cloudy days), the histogram thresholding technique will still threshold an image into two groups of pixels corresponding to dark or bright “objects.” Instead of clustering one group of pixels, two groups are clustered separately. As the size of each group is decreased, the total computation time should not be changed dramatically. In fact, it may be decreased.

IV. KNOWLEDGE ABOUT THE SCENE

The knowledge used in a system is represented by rules. The utility of production rules in representing knowledge has been

widely recognized for various expert systems [26]. Knowledge representation through production rules offers a great flexibility in coding knowledge from different sources and provides a reasonably effective control strategy. A common use in image analysis of rules is to represent functions (computer programs) together with the domain or control information about when and where to apply the functions. A production rule is generally described by

IF (precondition) THEN (action).

In the interpretation process, first the *precondition* of each rule is examined and, if it evaluates to true, the procedure in the *action* part is executed to generate and insert a new fact into the system or trigger the execution of another function. Both *precondition* and *action* can be functions.

A. Domain Knowledge

Domain knowledge is usually described using terminology defined in the real world: names of objects and their constituent parts, their spatial relationships, the objects' shapes, etc. For example, the domain knowledge about grass and trees is that they are all green in color, but trees have larger leaves than blades of grass. This domain related knowledge reveals that telling the tree leaves from grass by color would be difficult (if possible), but the differences in their shape might be significant. An explicit interpretation of the knowledge leads us to utilize texture features of trees and grass, when there are objects like trees and grass which need to be distinguished. Texture features, such as energy, entropy, etc., can be used to describe the texture of an object as smooth, coarse, grainy, and so on [1].

In this work, spatial information is categorized into domain knowledge, since it is heavily dependent upon the scene under consideration. Most natural scenes are composed of objects of various kinds which are related to each other through their geometric deployment. Thus, to understand the image, we need to understand the spatial relations between objects as well as their colors and textures. Using knowledge about spatial relations, the system can conduct reasoning about the structure of the scene. It should be mentioned that, in this work, we expect geometric relationships between objects to remain fixed, as the color images processed by the KGSL system were taken at the same location. In general, it would only be required that object characteristics and likely location be known (e.g., vehicles on ground versus on clouds or trees).

After clustering and cluster labeling, the labeled fuzzy cluster centroids of the training images form a decision space. Similarity of clusters results in close proximity of their centroids in the feature space. The cluster labeling rules can be extracted based on the distribution of the labeled cluster centroids, and later used to label other clusters. In a knowledge-guided segmentation using a fuzzy clustering algorithm, we use both domain specific knowledge and the cluster labeling rules as equal tools. The primary source of knowledge for merging clusters of over-segmented images comes from the pattern distribution of cluster centroids from the training images. Fig. 3 shows the distribution of labeled fuzzy cluster centroids projected onto H and S space. Many sidewalk (triangle) and grass (star) clusters

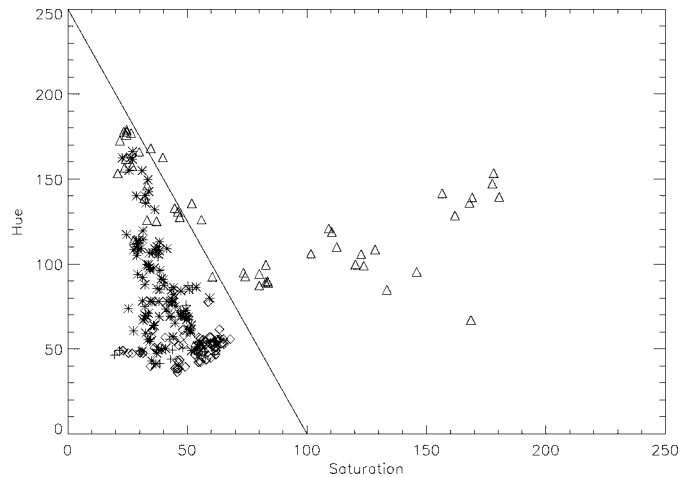


Fig. 3. Distribution of cluster centers. Triangle, star, and diamond represent sidewalks, grass, and bush, respectively.

are easy to identify, but bush (diamond) clusters overlap grass clusters.

Several approaches have been proposed by different researchers to extract heuristic cluster labeling rules from the feature space of cluster centroids [18], [17]. The simplest one is to visually identify the boundaries between different groups of clusters in a two-dimensional (2-D) or 3-D feature space and then label each cluster. A line in Fig. 3 is a boundary between clusters belonging to sidewalks and other objects. The line is represented by $S = 250 - 2.5 \times H$; the clusters on the right side of the line are considered to belong to sidewalks. A cluster labeling rule can thus be developed to label sidewalk clusters

IF $S \geq 250 - 2.5 \times H$ THEN ClusterID = SIDEWALK.

Superior to rules using the absolute locations of each cluster, rules using the relative positions of the clusters were proposed in [17]. Though both methods work well on multifeature images with the number of features less than three, images with more features make visual identification and hence extraction of heuristic cluster labeling rules difficult. The use of a decision tree learning algorithm for the same purpose will be discussed in Section V-B.

B. Control Knowledge

A major purpose of control knowledge is to realize effective image segmentation by using different image processing techniques. The control knowledge is derived from the domain knowledge. Based upon domain knowledge, simple object(s) can be chosen and identified; the entire image can be divided into subimages with reference to the identified object(s). Domain knowledge can help us to decide which subimage to start with and which object(s) will be recognized in each subimage. In the color images, sidewalks are considered as easily detectable objects and are identified first. Once the pixels of sidewalks are removed, the process becomes simpler due to less pixels and less uncertainty for the pixels class. The entire image is then divided into four regions: 1) upper left (UL); 2) upper right (UR); 3) lower left (LL); and 4) lower right (LR) relative to sidewalk (see Fig. 2). For example, grass, bush, and sidewalks are objects the system is seeking in the UL region.

The simplest approach to segment an image is to partition the image into regions by using a single technique with fixed parameters. In practice, obtaining an acceptable segmentation is very difficult unless in a highly controlled environment. When images are collected under different light conditions or some objects have similar features to others (e.g., each of grass, tree, and bush has the same color), local processing of spatial regions or feature regions of the image is desirable.

As different techniques are incorporated into the system, control knowledge becomes important in choosing the right techniques for specific object recognition. In the UL region, there are two objects we need to identify, namely, grass and bush. Identification of grass and bush would be restricted by using only a fuzzy clustering algorithm with color features due to the similar green color of the objects. Here, texture analysis helps. We apply texture analysis to the UL region, knowing that the bush has coarser texture than the surrounding grass. The following rule has been encoded as one control knowledge rule of the example system:

```
IF Region_of_Focus = UL
THEN ComputeTexture(image, Region_of_Focus).
```

Rules have been used to choose appropriate parameters for an algorithm. One problem of using a fuzzy clustering algorithm as an image processing tool is to tell the algorithm how many clusters to partition an image into. With too few clusters, we will undersegment the image, which means there are clusters not belonging to a unique object. By knowing the number of objects in each subimage, the system can, at least, force a fuzzy clustering algorithm to partition an image in such a way that the number of clusters is no less than the number of objects in the subimage

```
IF Region_of_Focus = LR
THEN Min_Clusters = 2 × NumObject[LR].
```

Region_of_Focus is a variable which indicates the current subimage under processing by the system. NumObject[] is an array indexed by UL, UR, LL, or LR and stores the number of expected objects in each subimage. It is possible to have more than one object of each type in a subimage. NumObject[] means the number of different types of object. Objects of the same type might be grouped into the same cluster by a fuzzy clustering algorithm.

When the subimage currently under processing is region LR, the system will set the minimum number of clusters to eight for a fuzzy clustering algorithm (there are grass, dead grass, light, and sidewalks in the LR subimage). The idea of over-segmentation guides the number of clusters into which an image is partitioned by a fuzzy clustering algorithm. Partition validity [27] is concerned with checking the quality of clustering results. It has been mainly used to evaluate and compare whole partitions, resulting from different algorithms or from the same clustering algorithm under different parameters. In this paper, the first minimum after the *Min_Clusters* using Xie and Beni's validity measure [27] is chosen as the number of clusters that is used by a fuzzy clustering algorithm to segment the LR subimage.

V. KNOWLEDGE BASE CONSTRUCTION

The problem of recognizing and therefore representing an expert's domain knowledge as rules is critical to build a KGSL system. As a practical matter, it is the most difficult aspect of knowledge-guided system development. Since an important part of domain knowledge in a KGSL system comes from ground-truthed multifeature images, the construction of the knowledge base consists of two major stages, they are image ground-truthing by experts and extracting rules to label clusters.

A. Image Ground Truthing

The image ground-truthing works as follows.

- 1) Ask the expert what object(s) need to be ground-truthed.
- 2) Ask the expert what feature images to use for ground-truth.
- 3) Ask the expert for an explanation of their choices.
- 4) Display an image, let the expert outline the object(s) on the image.
- 5) Display ground-truthed images and ask for modification.
- 6) Repeat Steps 1-5 until finished.

When asked for the objects to ground-truth, the expert is to be presented with a list of objects he/she is interested in. In this way, the system reminds the expert of all objects for which the system is searching. This must be implemented in the system prior to starting the ground-truthing process. Since the system segments multifeature images and each expert favors either one or more feature images for ground-truthing consideration, it would be a good idea to let the expert pick the feature images or work from a composite of feature images. The selected image(s) will be presented to and ground-truthed by the expert. The feature image can be a clustered image, in which, some or all clusters from a "good" clustering are presented to the experts and the experts are asked to identify each cluster.

Ground-truthed images should be presented to the expert for deliberation and further modification. However, there are two problems regarding ground-truthing that need to be addressed: *incompleteness* and *indeterminateness* [28]. *Incompleteness* refers to the fact that the expert will probably leave certain unspecified objects in the image. This problem can be solved by reminding the expert of all objects for which the system is searching. *Indeterminateness* is associated with the occasion when the expert is fairly vague about the exact object outlined. It reflects the fact that experts are not accustomed to outlining the object(s). The most straight forward method is to ignore the portion(s) vaguely outlined in an image and consider the image as partially ground-truthed. The portion(s) which have not been ground-truthed will not be used for training.

B. Extracting Cluster Labeling Rules

The image ground-truthing process by domain experts will allow the extraction of cluster labeling rules. The training images need to be (re-)clustered by the system to generate fuzzy

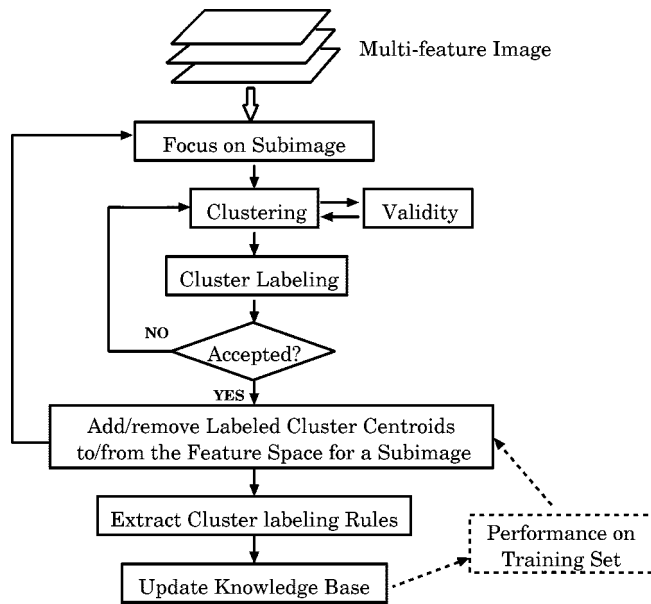


Fig. 4. Extracting cluster labeling rules through (re-)clustering. The performance on training set is computed during the knowledge base refinement.

clusters and to form a feature space of the cluster centroids before rules can be extracted. Centers can be labeled as the majority class of elements when the majority class percentage is above a threshold. The extracted cluster labeling rules are then added into the knowledge base. Major steps to extract cluster labeling rules are illustrated in Fig. 4.

Algorithm I:

- 1) Enter a ground-truthed image into the system.
- 2) Cluster a subimage using a fuzzy clustering algorithm.
- 3) Label a cluster as class of majority where the majority percentage is above a threshold.
- 4) Add labeled clusters to the feature space of a subimage.
- 5) Repeat Steps 2-4 for unlabeled clusters and for a predefined # of iterations.
- 6) Repeat Steps 2-5 for each subimage.
- 7) Repeat Steps 1-6 for each ground-truthed image.
- 8) Extract cluster labeling rules.
- 9) Add the rules to the knowledge base of the KGSL system.

Given the control structure implemented in the system, the fuzzy cluster labeling rules can be extracted using the algorithm shown in Algorithm I. Each ground-truth image is first clustered and the cluster(s) corresponding to simple object(s) (e.g., sidewalks) are labeled. The identified simple objects are used as the reference objects to produce a frame of geometric reference. The whole image is divided into subimages. Each subimage is (re-)clustered using a fuzzy clustering algorithm. The label for a cluster is accepted if its majority pixels (e.g., more than 90%) belong to a unique object.

Feedback from a partition validation metric [27] is used to determine the number of clusters the subimage will be partitioned into. Each cluster of a partitioned image is labeled against the ground-truth, and added to the feature space. After all fuzzy clusters of the training set are labeled and added to the feature space, cluster labeling rules are extracted and added to the system knowledge base. The removal of labeled clusters happens in the knowledge refinement, when the addition of the clusters of a new training image degrades the system performance [29].

After the FCM algorithm is applied to a training image, each pixel in a segmented image will be labeled using its ground-truth image. A pixel p_{xy} in a segmented image will be labeled O_i , if it is mapped to a pixel belonging to an object O_i in the ground-truth image. The coordinates x, y of the pixel are used for the mapping. Generally, a cluster is identified as an expected part of an object, if an appropriate percentage of pixels ($\text{Percent}(O_i)$) in a cluster belong to a unique object in the ground-truth image. The $\text{Percent}(O_i)$ is computed for each object O_i . The label of O_i is assigned to a cluster, if $\text{Percent}(O_i)$ is above a predefined threshold value (e.g., 90%). All unlabeled clusters are collected and reclustered by the system.

The above cluster labeling approach is ideal. In practice, some clusters cannot be labeled due to: 1) incomplete ground-truthing, or 2) being a heterogeneous (inhomogeneous) cluster. The first case happens because images are partially ground-truthed. It can be avoided by masking out the portions of the image that are not ground-truthed from being processed by the KGSL system. In other words, unground-truthed image pixels should not be processed by the system and used to generate cluster labeling rules.

A heterogeneous cluster is the result of imperfect segmentation. It is possible that through a predefined number of clustering iterations, some clusters remain unlabeled. Hence, the system should keep track of heterogeneous clusters, and if the number of heterogeneous clusters is significant, the system should remind the developer to consider other fuzzy clustering algorithms or even different types of image processing techniques. A simple approach to deal with heterogeneous clusters is to exclude them from training.

Numerous papers have appeared comparing the performance of a variety of learning algorithms on real and constructed datasets [30], [31], [5], [32], [33]. One observation is that their capabilities appear to be task-dependent. We are interested in using decision trees, in particular C4.5 [34], to generate “understandable” rules from a set of labeled cluster centroids of training images. A discussion of the use of neural network to learn from the labeled fuzzy clusters can be found in [18]. If you are willing to sacrifice interpretability, you may substitute your favorite classifier here.

The labeled fuzzy clusters for each subimage form a training set for decision tree learning. We used decision trees, like C4.5, to generate a collection of rules (using C4.5 rules). The heuristic rules can be applied to label fuzzy clusters produced from a new image. It may happen that several of the rules are applicable, so C4.5 provides a measure of correctness from the training set for each rule extracted. If the set of applicable rules predict different classes, we will accept the decision generated by the rule having the highest confidence. This problem may vanish after more training images are available and processed.

C. Unlabeled Clusters

One problem associated with cluster labeling using rules derived by C4.5 rules is that some clusters may not be identified as belonging to any of the expected objects. This is usually caused by inadequate generalization of decision trees. To solve this problem, the k nearest neighbor (kNN) algorithm is applied to the testing images to label unidentified clusters, after a set of rules created by C4.5 rules from the decision trees is applied in processing an image.

Suppose that a labeled cluster from a training set is represented by a vector \mathbf{V} . The Euclidean distance between an unlabeled cluster vector \mathbf{X} and the vector \mathbf{V} can be calculated by

$$d(\mathbf{X}, \mathbf{V}) = \|\mathbf{X} - \mathbf{V}\| = \sqrt{[(\mathbf{X} - \mathbf{V})^T(\mathbf{X} - \mathbf{V})]}. \quad (1)$$

The distances $d(\mathbf{X}, \mathbf{V})$ are ranked and the k minima ($k = 7$) are selected. To decide which object the cluster \mathbf{X} belongs to, we count the frequency of each class labeled for the k selected minima and the majority is chosen as the label for cluster \mathbf{X} .

VI. KNOWLEDGE BASE REFINEMENT

It is desirable that the initial knowledge base of a knowledge-guided system can be progressively refined into a higher performance system. This feature is particularly important in some applications like remote sensing and medical image processing. Data collection for remote sensing study may be done over several years or even a decade, due to the fact that most geophysical phenomena occur over long periods of time and cover vast areas. In medical image processing, new pathological cases become available over years. With the capability of refinement, the knowledge base can grow incrementally as new ground-truth images are processed and cluster labeling rules are extracted. This feature is also important to maintain the expert's and management's interest in the system by showing them improved performance via the system.

The KGSL approach uses separate training and testing data sets. In a single typical train- and-test run, the ground-truth images are divided into two disjoint subsets: training and testing sets. After training the system with the training set, we test the performance of the system using the independent unseen testing set of images. Later, when new ground-truth images become available, we can refine the knowledge base (mostly the cluster labeling rules) through (re-)clustering. Knowledge base refinement follows the same procedures of knowledge base construction, as shown in Fig. 4. In addition, the performance of the system over the training set is evaluated and used to either accept or reject a new training image.

During knowledge refinement, the initial knowledge base is progressively refined into a higher performance knowledge base. The refinement stage is characterized by the addition of new objects, enhancement of the pattern distribution of cluster centroids, or even removal of some inaccurate training data in an attempt to improve the system's ability to reach correct conclusions. The refinements are mostly applied to the feature space of a training set through the addition or deletion of labeled clusters.

A. Performance Measures

The system performance is computed over the training set which is considered to be controlled and mostly noise-free. For

objects that the system is capable of recognizing, the goal of refinement can be viewed as reducing misclassified cases. To elaborate on this point, we define pixel-level true positive (TP) for an object O_i in (2). The true negative (TN), false positive (FP), and false negative (FN) are defined similarly [28]

$$TP(O_i) = \frac{\# \text{Pixels correctly classified as } O_i}{\text{Total } \# \text{pixels belonging to } O_i}. \quad (2)$$

The pixel-levels TP, TN, FP, and FN are computed for each object in the training set. For example, if a pixel is correctly labeled as belonging to an object O_i against the ground-truth image, the pixel will be counted toward “#Pixels correctly classified as O_i .” The “Total #pixels belonging to O_i ” is the number of pixels ground-truthed to object O_i for all training images.

The refinement of the knowledge base is viewed as improving system performance as images or objects are added. For this purpose, let us define the difference in $TP(O_i)$ before and after the refinement as $\Delta TP(O_i)$ [see (3)], and similarly for $TN(O_i)$, $FP(O_i)$, and $FN(O_i)$. The change of the performance of a system before and after refinement is defined in (4)

$$\Delta TP(O_i) = TP_{\text{refined}}(O_i) - TP(O_i) \quad (3)$$

$$\Delta P = \sum_i w(O_i) [\Delta TP(O_i) + \Delta TN(O_i) - \Delta FP(O_i) - \Delta FN(O_i)] \quad (4)$$

where the $w(O_i)$ is the “cost” associated with an object O_i and it is a nonnegative number in $[0.0, 1.0]$. The refinement will be accepted only when the condition $\Delta P > 0$ is satisfied.

In real-world problems, there may be misclassification costs associated with the objects. If the costs are not identical between different objects, performance measures of the system need to be designed explicitly to prefer improved recognition of higher weighted objects. Attempts have been made to incorporate misclassification costs when doing more fully automated machine learning [35]–[37].

B. The Refinement

A KGSL system keeps a record of its pixel-level performance for the training images. Using such information, the system can tell it has weaknesses in recognizing some objects and when possible, it can remind the domain experts who use the system to provide training images to allow it to improve. To add an image into the training set and update the knowledge base, the system goes through the steps listed in Algorithm II.

Algorithm II:

- 1) Repeat Steps 1–6 of Algorithm I for a new training image.
- 2) Extract cluster labeling rules.
- 3) Compute the performance for training images (original and new) using the new ruleset.
- 4) If performance improved, accept the ruleset.
- 5) Otherwise,
 - remove new clusters from the feature space, and reinstate the old ruleset.

In practice, addition of a new training image into the training set causes the addition of labeled clusters of the image into the

feature space of the original training images. Hence, the distribution of labeled clusters will be temporarily altered. After the system extracts the cluster labeling rules and updates the knowledge base, clusters from the training images (original and new) must be relabeled using the updated knowledge base. The system performance is computed for the training set (original and new). Once it is proven that the addition will improve the system performance, the new ruleset will be stored and the performance will be recorded for the updated knowledge base.

It is possible that a training image contains a new object on which the system has not been trained before. The KGSL system keeps a list of objects it can recognize and compares the name of a new object with the list. The extraction of the cluster labeling rules for a new object follows the same procedures as in knowledge base construction. As a difference between the knowledge base construction and knowledge base refinement, while the cluster labeling rules are extracted after an image is processed by the system during knowledge refinement, cluster labeling rules are extracted only after all training images are processed in the initial knowledge base construction. This is because, during knowledge refinement, cluster centers exist (with labels) for the original training images.

The refinement of the cluster labeling rules can be either *cluster-based* or *image-based*. By *cluster-based* refinement, after each cluster is labeled against the ground-truth, it is added to the feature space immediately and the cluster labeling rules are extracted right away. By *image-based* refinement, only after the entire image has been labeled or the system has gone through a predefined number of clustering iterations, are all labeled clusters of an image added to the feature space. A pure *cluster-based* refinement takes considerably more computation time than *image-based* refinement. However in *image-based* refinement, when a training image is rejected, all clusters of the image are rejected, meaning some valuable information contained in the image may be lost. Knowledge base refinement of the KGSL approach could be either *stepwise-forward* or *stepwise-backward* refinements or their combination. In *stepwise-forward* refinement, we start with an old training set, and add, at each stage, some training data to the training set. The *stepwise-backward* refinement is an elimination procedure. We start with a full training set and remove, at each step, the worst training data. We propose an *image-based stepwise-forward* refinement for the KGSL approach, in which, all labeled clusters of a training image are added to the feature space for rule extraction. In such an approach, since we use *stepwise-forward* refinement, the knowledge base grows and the performance will consistently improve.

VII. SEGMENTATION OF COLOR IMAGES

To build and test a specific KGSL system, a total of 93 color photographs were taken outdoors. The implementation of a KGSL system started with 45 training images. The rest of images were left for testing the system. A KGSL system began the segmentation using the mrFCM [38] clustering algorithm. The sidewalks on the image were identified first. The entire image was divided into four subimages (UL, UR, LL, and LR) relative to the sidewalks. Rules were set up to control the

TABLE I
PIXEL-LEVEL (PERCENTAGE) CONFUSION
MATRIX FOR 48 TESTING IMAGES PROCESSED BY THE SYSTEM (“DG” AND
“SW” DENOTE DEAD GRASS AND THE SIDEWALK, RESPECTIVELY)

As \Rightarrow	Bush	DG	Grass	Light	SW	Tree
Bush	84		15		1	
DG		76	15	8	1	
Grass	2	3	82	1	2	10
Light		3	5	80	12	
SW		2	9	1	88	
Tree			34			66

processing to seek from different objects on each subimage. When there were objects like trees and grass that needed to be distinguished, texture features [1] were also calculated. Each subimage was focused on and segmented using mrFCM. Knowing the number of objects it was searching for on each subimage (coded in rules) and by using a validity measure [27], the KGSL system will segment a subimage into an appropriate number of clusters. A cluster was labeled as the majority class of pixels if more than 90% of pixels in a cluster belong to a unique object. All unlabeled clusters for a subimage were collected and reclustered by the system. The labeled fuzzy cluster centroids of training images form a feature space. The cluster labeling rules were extracted using C4.5 rules and were added into the knowledge base of the system.

After all training images had been processed to build the knowledge base, the unseen 48 images were processed and used to evaluate the performance of the system. All object segmentation generated by the KGSL system were compared with “ground-truth” objects previously outlined by the author. The pixel-level confusion matrix was computed for all objects in the training set, testing set, and both. Table I lists the results of the application of the KGSL system to the testing set. The system correctly classifies 84%, 82%, 80%, and 88% of pixels into bush, grass, light, and sidewalks, respectively. The KGSL system tends to misclassify 15% of dead grass pixels and 34% of tree pixels into normal grass. The misclassification of tree pixels is partly attributed to leaves of the tree that fell on the grass. Even though the texture features of objects were calculated and compared, grass mixed with tree leaves is still difficult to distinguish from a tree.

The misclassifications are also attributed to shade from the tree or the bush. When the pictures were taken, the tree blocked the sunlight and cast shadows on the ground. Though the histogram thresholding method was used to separate shadow pixels from unshadowed pixels before clustering, the problem is not completely solved. As shadows were also cast on the grass by the bush, misclassification between bush and grass exists. Shading also contributes to the misclassification of sidewalks into grass for the same reason. Generally, the system performs better on images without shading, for example when the pictures were taken under a cloudy sky. Fig. 5 shows a typical image before and after segmentation.

A. Comparison With Simple Clustering

One of the advantages of the KGSL system is that the system uses the concept of *focus-of-attention*, which has been realized

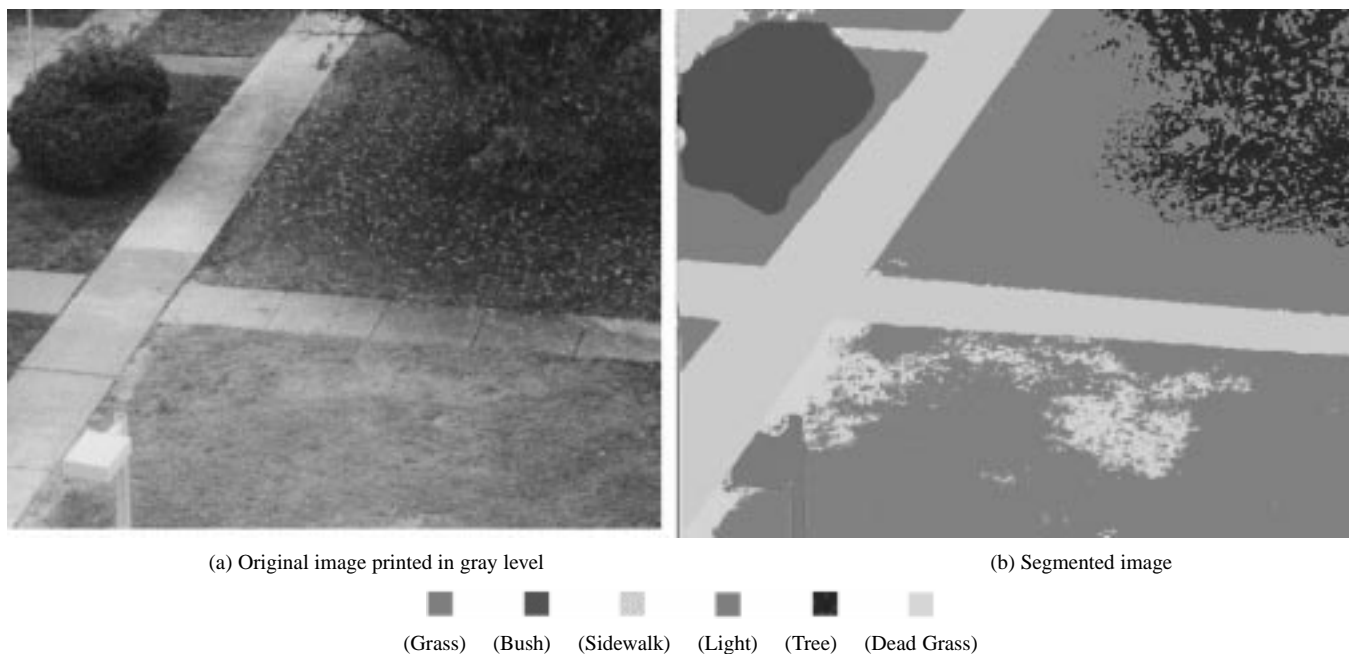


Fig. 5. Images before and after the segmentation using the KGSL system.

TABLE II
PIXEL-LEVEL (PERCENTAGE) CONFUSION MATRIX FOR TESTING SET BY A SIMPLE SEGMENTATION WITH 14 CLUSTERS (“DG” AND “SW” DENOTE DEAD GRASS AND THE SIDEWALK, RESPECTIVELY)

As \Rightarrow	Bush	DG	Grass	Light	SW	Tree
Bush	23	1	46		1	29
DG	1	60	24	1	13	1
Grass	5	9	70	2	7	7
Light		27	12	20	40	1
SW		6	2	5	85	2
Tree	9	2	46		2	41

through division of the image into subimages and through (re-)clustering. The simplest approach to segment an image is to apply a single technique with fixed parameters to the whole image. Table II shows how well a simple segmentation using a fuzzy clustering algorithm performed on the same set of images processed by the KGSL system. All images are converted into HSI and divided into training and testing sets, as in the KGSL system. Each image is clustered into 14 clusters using the mrFCM clustering algorithm.

A comparison is also made against decision trees, created by C4.5 [34]. The C4.5 algorithm is trained with the same training set used by the KGSL system, and then, the pruned decision trees created from training are applied to testing images. All C4.5 parameters are at their default values. Results are available in Table III.

Based on the comparisons, the question as to whether or not the results from the KGSL system are significantly better than others can be answered by computing the “ P -values.” The achieved level of significance is often denoted by P and is some times referred to as the “ P -value” [39]. It can be interpreted as a measure of the plausibility of the hypothesis that *result from KGSL system is better than a single-stage decision tree (or fuzzy clustering)*. Results are available in Table IV.

TABLE III
PIXEL-LEVEL (PERCENTAGE) CONFUSION MATRIX FOR TESTING SET USING C4.5 (“DG” AND “SW” DENOTE DEAD GRASS AND THE SIDEWALK, RESPECTIVELY)

As \Rightarrow	Bush	DG	Grass	Light	SW	Tree
Bush	63	1	7		5	24
DG	1	68	20		10	1
Grass	5	5	73		3	14
Light	1	5	7	52	34	1
SW	1	6	1	6	85	1
Tree	16	1	20		3	60

TABLE IV
SIGNIFICANCE TEST FOR COMPARISONS WITH C4.5 OR A SINGLE-STAGE FUZZY CLUSTERING APPROACHES (“DG” AND “SW” DENOTE DEAD GRASS AND THE SIDEWALK, RESPECTIVELY)

	KGSL		C4.5			FCM		
	μ_1	s_1	μ_2	s_2	P	μ_3	s_3	P
Bush	84	7.2	63	5.3	.99	23	10.1	1.0
Grass	82	10.9	73	8.4	.75	70	11.1	.78
Light	80	9.1	52	4.2	.99	20	13.3	1.0
SW	88	7.9	85	6.5	.61	85	6.0	.62
Tree	66	10.5	60	11.0	.65	41	9.3	.96
DG	76	9.1	68	6.7	.78	60	9.8	.88

Based on Table IV, the KGSL system significantly outperforms the single-stage decision tree on recognition of the bush and light, with P -value of at least 99%. It also significantly outperforms the single-stage fuzzy clustering approach on recognition of the bush, light, and tree. Generally, the decision trees perform better than the fuzzy clustering algorithm due to the fact that a segmentation by decision trees is done on each pixel of the image. When a fuzzy clustering algorithm is used, pixels are first grouped into clusters, and then, each cluster is labeled by cluster labeling rules. If a cluster is not a homogeneous cluster, such as a cluster consisting pixels from different objects, the minority class(es) of pixels in a cluster will be mislabeled.

The KGSL system outperforms the simple segmentation for **all** objects. If we take a look at the first row of Tables II and III, there is a significant portion of bush misclassified. The misclassification of bush to tree is due largely to the facts that bush and tree have the same color and similar texture features. This type of misclassification is avoided by in the KGSL system, because the system divides the whole image into subimages and realizes the bush and the tree are in the different regions. Different from single segmentation method using a fuzzy clustering, the KGSL system utilizes the (re-)clustering approach: pixels of labeled clusters are collected and reclustered. The optimal number of clusters a subimage partitioned into is determined by the knowledge base. This effort reduces the number of heterogeneous fuzzy clusters generated by mrFCM.

Even with a simple segmentation approach, some objects in the image can be successfully identified, such as sidewalks. With 85% of sidewalks correctly classified by any method, it is reasonable to believe the sidewalks are simple objects and should be extracted first. We define a simple object as an object that can be segmented successfully using an one-step segmentation method. In system implementation, a set of training images is segmented once using a fuzzy clustering algorithm; clusters are labeled against the ground-truth images; a pixel-level confusion matrix is computed. The object which is most correctly identified is chosen as a simple object.

B. Other Applications

This paper describes an approach or a framework to process images taken at the same location. To apply this approach, we expect the geometric relationships between objects to remain fixed. Applications of the approach may be found in processing MRI or satellite images. Indeed, knowledge-guided segmentation systems have been developed for different domain areas. It has been applied to work on MR images to identify enhanced tumor tissue in the human brain by Clark and Hall *et al.* [17] and by Fletcher-Heath and Hall *et al.* [40] without enhancement. MR brain images are initially segmented by an unsupervised algorithm (FCM), then an expert system locates a tissue or a cluster and analyzes it by matching it with a model or searching the tissue or cluster in a subimage for an expected feature. The knowledge base contains information on cluster distribution in feature space and tissue models. The idea of *focus-of-attention* was achieved by concentrating on pixels that correspond to the intra-cranial region. Pixels belonging to extra-cranial tissues, such as air, skin, and fat are not of interest and are excluded from further processing.

The idea of *focus-of-attention* can also be found in [18]. Zhang and Hall *et al.* [18] applied a knowledge-guided approach to satellite images. Coastal zone color scanner (CZCS) images were processed by the system and the pixels of different types of ocean water were identified (see Fig. 6). Their system first identified the pixels belonging to the land and excluded them from later processing. To focus on the coastal regions, a bathymetry map was used to exclude pixels belonging to the open ocean. The rest of pixels were (re-)clustered. All three of the systems [17], [18], [40], fit nicely into the framework or approach illustrated here.

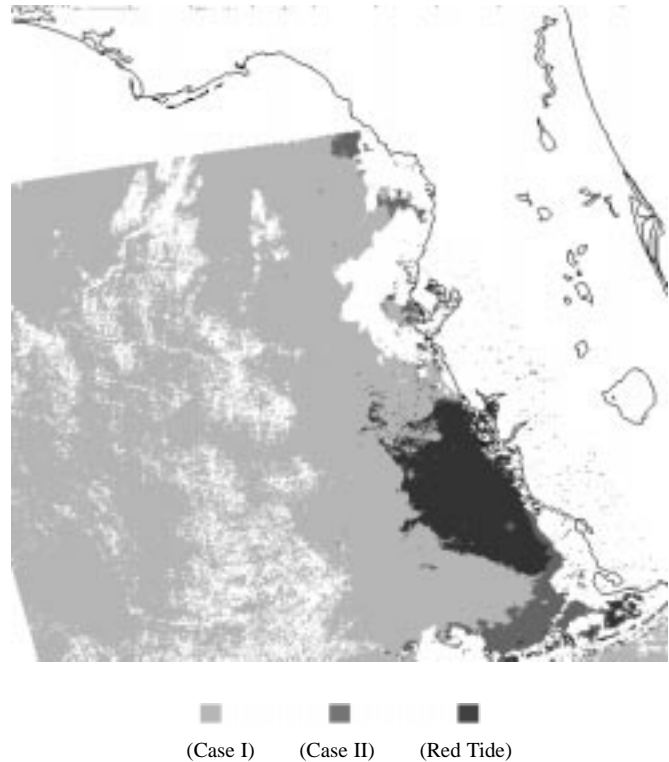


Fig. 6. CZCS images with red tides after processing by the knowledge-based system.

VIII. SUMMARY AND DISCUSSION

A KGSL approach or framework which automatically segments and identifies objects in multifeature images was described. The promise of the approach was illustrated by processing a set of color images of a fixed outdoor location under varying light conditions. Segmentation results of color images are compared with two single-stage classifiers which rely on fuzzy clustering or decision trees. The system built with the KGSL approach outperforms those classifiers. As the KGSL system makes use of multiple classifiers and integrates different image processing algorithms under the guidance of a knowledge base, it makes sense that it will, if properly applied, outperform single-stage classifiers.

The approach uses fuzzy clustering algorithms as a major processing technique, and if necessary, it incorporates different types of image processing algorithms. To begin, it activates the initial segmentation using a fuzzy clustering algorithm and identifies the simple object(s) in the image. Based on the extracted object(s), a geometric frame of reference is set; the whole image may then be divided into subimages. The (re-)clustering and identification of objects is done to the subimages rather than to the entire image.

Since a major portion of knowledge comes from images ground-truthed by domain experts, the KGSL approach provides knowledge extraction functions to automatically extract the cluster labeling rules. After an image is processed by the system, each cluster of a segmented image which obtains a minimum homogeneity percentage is labeled as the label of its majority of pixels. The labeled cluster centroids form a decision space from which cluster labeling rules can be extracted,

using a rule learning algorithm (or other machine learning algorithm). The extracted rules are added to the knowledge base. When a new training image is available, the image may be segmented by the system, and labeled clusters can be added into the training set. The addition of labeled clusters activates the updating of the knowledge base.

Though the images segmented by the system described in this paper are expected to be taken at a fixed location, there are a variety of image understanding and analysis problems which can be solved using the KGSL approach. For example, it can be used to help a physician to measure tumor tissue in a human body and it can be applied to process remotely sensed images by satellites to monitor the weather or environmental changes. In either application, we may expect small variations between images taken at different time or for different patients. Geographic features, such as buildings, roads, and islands will generally remain on satellite image of same location. The human skull, ventricle, and the corpus callosum are on MR brain images of different patients. Even the relative locations of the objects do not change. The system described in this paper is useful for segmenting those images to detect any abnormality.

The construction of a knowledge base involves some human effort. However, building an image recognition system from scratch without using high-level knowledge is going to be significantly more time-consuming and less structured for success. Part of problem is to search for the ways to fit different algorithms together and come up with a control structure [8]. The paper discusses ways to organize the process of recognition utilizing some *priori* knowledge about the scene. It also describes an approach that can be used to extract knowledge, such as cluster labeling rules, from a set of ground-truthed images. Hence, this system can be applied to many different kinds of domains.

The features of knowledge refinement are especially useful for medical image processing and remote sensing applications. In these applications, the acquisition of ground-truth images will be over a long period of time. Whenever new ground-truth images with interesting features are available, the KGSL approach can be applied to segment the images, automatically extract cluster labeling rules and add them to the knowledge base to improve its performance. The promise of the system was demonstrated by processing a set of color images and the application of the approach to ocean satellite images.

REFERENCES

- [1] R. Jain, R. Kasturi, and B. Schunck, *Machine Vision*. New York: McGraw-Hill, 1995.
- [2] J. C. Bezdek, L. O. Hall, and L. P. Clarke, "Review of MR image segmentation techniques using pattern recognition," *Med. Phys.*, vol. 20, no. 4, pp. 1033–1048, 1993.
- [3] M. Vaidyanathan, L. P. Clarke, R. P. Velthuizen, S. Phuphanich, A. M. Bensaid, L. O. Hall, J. C. Bezdek, H. Greenberg, A. Trotti, and M. I. Silbiger, "Comparison of supervised MRI segmentation methods for tumor volume determination during therapy," *Magn. Reson. Imag.*, vol. 13, no. 5, pp. 719–728, 1995.
- [4] L. P. Clarke, R. P. Velthuizen, L. O. Hall, J. C. Bezdek, A. M. Bensaid, and M. I. Silbiger, "Comparison of supervised pattern recognition techniques and unsupervised methods for MRI segmentation," *Proc. SPIE*, vol. 1652, pp. 668–677, 1992.
- [5] A. Baraldi and P. Blonda, "A survey of fuzzy clustering algorithms for pattern recognition—Part I," *IEEE Trans. Syst., Man, Cybern. B*, vol. 29, pp. 778–785, Dec. 1999.
- [6] J. C. Bezdek and S. K. Pal, *Fuzzy Models for Pattern Recognition*. Piscataway, NJ: IEEE Press, 1992.
- [7] L. Mascariilla, "Fuzzy rules extraction and redundancy elimination: An application to remote sensing image analysis," *Int. J. Intell. Syst.*, vol. 12, pp. 793–817, 1997.
- [8] M. Heath, S. Sarkar, T. Sanocki, and K. Bowyer, "A robust visual method for assessing the relative performance of edge-detection algorithms," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 1338–1359, Dec. 1997.
- [9] V. S. Nalwa and T. O. Binford, "On detecting edges," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 699–714, 1986.
- [10] S. Sarkar and K. L. Boyer, "Optimal infinite impulse response zero crossing based edge detectors," *CVGIP, Image Understand.*, vol. 54, pp. 1–13, 1992.
- [11] S. P. Raya, "Low-level segmentation of 3-D magnetic resonance brain images: A rule-based system," *IEEE Trans. Med. Imag.*, vol. 9, no. 3, pp. 327–337, 1990.
- [12] M. Sonka, S. K. Tadikonda, and S. M. Collins, "Knowledge-based interpretation of MR brain images," *IEEE Trans. Med. Imag.*, vol. 15, pp. 443–452, Apr. 1996.
- [13] M. Kobashi and L. G. Shapiro, "Knowledge-based organ identification from CT images," *Pattern Recognit.*, vol. 28, no. 4, pp. 475–491, 1995.
- [14] N. Ezquerro and R. Mullick, "Knowledge-guided segmentation of 3-D imagery," *CVGIP, Graph., Models, Image Process.*, vol. 58, no. 6, pp. 512–523, 1996.
- [15] H. Suzuki and J. Torikawa, "Automatic segmentation of head MRI images by knowledge-guided thresholding," *Comput. Med. Imag. Graph.*, vol. 15, no. 4, pp. 233–240, 1991.
- [16] D. M. McKeown, W. A. Harvey, and J. McDermott, "Rule-based interpretation of aerial imagery," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-7, pp. 570–585, May 1985.
- [17] M. C. Clark, L. O. Hall, D. B. Goldgof, R. Velthuizen, F. R. Murtagh, and M. S. Silbiger, "Automatic tumor segmentation using knowledge-based clustering," *IEEE Trans. Med. Imag.*, vol. 17, pp. 187–201, Feb. 1998.
- [18] M. Zhang, L. O. Hall, D. B. Goldgof, and F. Muller-Karger, "Knowledge-guided classification of coastal zone color images off the West Florida shelf," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 14, no. 8, pp. 987–1007, 2000.
- [19] A. M. Bensaid, L. O. Hall, J. C. Bezdek, and L. P. Clarke, "Partially supervised clustering for image segmentation," *Pattern Recognit.*, vol. 29, no. 5, pp. 859–871, 1996.
- [20] N. M. Dempster, A. P. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Statist. Soc. B*, vol. 39, pp. 185–197, 1977.
- [21] I. R. Krishnapuram, H. Frigui, and O. Nasraoui, "41 fuzzy and possibilistic shel clustering algorithms and their applications to boundary detection and surface approximation," *IEEE Trans. Fuzzy Syst.*, vol. 3, no. 1, pp. 29–43, 1995.
- [22] J. Giarratano and G. Riley, *Expert Systems: Principles and Programming*, 2nd ed. Boston, MA: PWS-Kent, 1994.
- [23] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Reading, MA: Addison-Wesley, 2001.
- [24] C. Jiang and M. O. Ward, "Shadow segmentation and classification in a constrained environment," *CVGIP, Graph., Models, Image Process.*, vol. 59, pp. 213–225, 1994.
- [25] M. Nagao, T. Matsuyama, and Y. Ikeda, "Region extraction and shape analysis in aerial photographs," *Comput. Vis., Graph., Image Process.*, vol. 10, pp. 195–223, 1979.
- [26] F. Hayes-Roth, D. A. Waterman, and D. B. Lenat, *Building Expert Systems*. Reading, MA: Addison-Wesley, 1983.
- [27] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-11, pp. 841–847, July 1980.
- [28] L. Eshelman, D. Ehret, J. McDermott, and M. Tan, "MOLE: A tenacious knowledge-acquisition tool," in *Readings in Knowledge Acquisition and Learning*. San Mateo, CA: Morgan Kaufmann, 1993, pp. 253–259.
- [29] D. H. Fisher, "Knowledge acquisition via incremental conceptual clustering," *Mach. Learn.*, vol. 2, pp. 133–172, 1987.
- [30] S. P. Curram and J. Mingers, "Neural networks, decision tree induction and discriminant analysis: An empirical comparison," *J. Oper. Res. Soc.*, vol. 45, no. 4, pp. 440–450, 1994.
- [31] J. W. Shavlik, R. J. Mooney, and G. G. Towell, "Symbolic and neural learning algorithms: An empirical comparison," *Machine Learning*, vol. 6, no. 2, pp. 111–144, 1991.
- [32] Y. R. Barai and S. V. Barai, "Evaluating machine learning models for engineering problems," *Artif. Intell. Eng.*, vol. 13, no. 3, pp. 257–272, 1999.

- [33] N. Cercone, A. A. Chan, and C. Chan, "Rule-induction and case-based reasoning: Hybrid architectures appear advantageous," *IEEE Trans. Knowl. Data Eng.*, vol. 11, no. 1, pp. 166–174, 1999.
- [34] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, CA: Morgan Kaufmann, 1993.
- [35] B. A. Draper, C. E. Brodley, and P. E. Utgoff, "Goal-directed classification using linear machine decision trees," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, Sept. 1994.
- [36] P. D. Turney, "Cost-sensitive classification: Empirical evaluation of a hybrid genetic decision tree induction algorithm," *J. Artif. Intell. Res.*, vol. 2, pp. 369–409, 1995.
- [37] P. Domingos, "MetaCost: A general method for making classifiers cost-sensitive," in *Proc. 5th Int. Conf. Knowledge Discovery and Data Mining*, San Diego, CA, 1999, pp. 155–164.
- [38] T. W. Cheng, D. B. Goldgof, and L. O. Hall, "Fast fuzzy clustering," *FUZZ-IEEE*, pp. 2289–2295, 1995.
- [39] T. W. Anderson and S. L. Sclove, *An Introduction to the Statistical Analysis of Data*. Boston, MA: Houghton Mifflin, 1978.
- [40] L. M. Fletcher-Heath, L. O. Hall, D. B. Goldgof, and F. R. Murtagh, "Automatic segmentation of nonenhancing brain tumors in magnetic resonance images," *Artif. Intell. Med.*, vol. 21, pp. 43–63, 2001.



Mingrui Zhang received the B.S. degree in radio-electronics from the Peking University, Beijing, China, in 1988, and the M.S. degree in marine science and the M.S. and Ph.D. degrees in computer science and engineering from University of South Florida, Tampa, in 1994, 1996, and 1999, respectively.

Currently, he is an Assistant Professor at Winona State University, Winona, MN. His research interests include expert system, multispectral image processing, fuzzy clustering, and their applications

to ocean remote sensing.



Lawrence O. Hall (S'85–M'85–SM'98) received the B.S. degree in applied mathematics from the Florida Institute of Technology, Melbourne, in 1980, and the Ph.D. degree in computer science from Florida State University, Tallahassee, in 1986.

Currently, he is Professor of Computer Science and Engineering at the University of South Florida, Tampa. His research interests lie in distributed machine learning, pattern recognition, and integrating artificial intelligence into image processing. The exploitation of imprecision with the use of fuzzy

logic in pattern recognition, artificial intelligence, and learning is a research theme. He has authored over 140 publications in journals, conferences, and books. Recent publications have appeared in *Artificial Intelligence in Medicine*, *Fuzzy Sets and Systems*, *JAIR*, *IEEE TRANSACTIONS ON FUZZY SYSTEMS*, *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS*, the *International Fuzzy Systems Association/NAFIPS Conference* (2001), and *CVPR* (2001). He also co-edited the *2001 Joint North American Fuzzy Information Processing Society (NAFIPS), IFSA Conference Proceedings*.

Dr. Hall received the IEEE SMC Society Outstanding Contribution Award in 2000. He was a former President of NAFIPS and the Vice President for membership of the SMC society. He is currently Editor of *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS*. He is also Associate Editor for *IEEE TRANSACTIONS ON FUZZY SYSTEMS*, *International Journal of Intelligent Data Analysis*, and *The Handbook of Fuzzy Logic*.



Dmitry B. Goldgof (S'84–M'87–SM'93) received the Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign in 1989.

He is currently a Professor in the Department of Computer Science and Engineering, the University of South Florida, Tampa, and a Member of H. Lee Moffitt Cancer Center. His research interests include motion analysis of rigid and nonrigid objects, computer vision, image processing and its biomedical applications, and pattern recognition. He has graduated nine Ph.D. students and 19 M.S. students, and has published

over 45 journal and 90 conference publications, 15 books chapters, and one book.

Dr. Goldgof is a winner of the 19th Annual Pattern Recognition Society Award. He is currently the North American Editor for *Image and Vision Computing Journal* and the Electronic Associate Editor for *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS* (electronic section). He is a Member of the International Association of Pattern Recognition Education Committee.