Combining Multiple Segmentation Methods for Improving the Segmentation Accuracy

Sultan Aljahdali
College of Computer Sciences
Taif University, Saudi Arabia
aljahdali@tu.edu.sa

E.A. Zanaty
College of Computer Sciences
Taif University, Saudi Arabia
zanaty22@yahoo.com

Abstract: *In this paper, an alternative method based on decision fusion is presented to improve the segmentation accuracy. The proposed method concludes multiple methods instead of a single one. It consists of a set of segmentation methods that are consulted in parallel. The decisions of the various methods are then combined by a fusion module. The individual methods, in this case, are capable of independent and simultaneous operation. Then, we apply and compare the fusion schemes to the area of image segmentation. We seek answers to the questions: Can combining multiple segmentation methods achieve better, final partitioning of an image? If so, how much is this improvement? And which fusion scheme may perform best of all?*

Keywords: *image segmentation, classical and clustering techniques, medical images, decision fusion.*

I. Introduction

Image segmentation is still a subject of on-going investigations and it cannot be conclusively stated that the segmentation problem has been solved because of the application's diversity. In particular, the superiority of one technique over the others cannot be claimed; many experimental results [1,2] showed that the segmentation accuracy depends more on the particular application rather than on the technique chosen to perform the task. Moreover in many applications it is very difficult to design segmentation system that exhibits the required accuracy for the final segmentation product.

A variety of schemes have been proposed for the combination step in the different application domains (e.g., [3,4]). In [5] multi-clustering approach is introduced, where multiple clusterings (using the k-means algorithm) are exploited to determine a co-association matrix of patterns, which is used to define an appropriate similarity measure that is subsequently used to extract arbitrarily shaped clusters.

Olivier et al. [6] proposed a Bayesian method for data fusion, rather than decision fusion, of images, with a Potts Markov Random field model on the hidden variable. Alexander et al. [7] considered combining weak clustering algorithms that use data projections and random data splits. A simple explanatory model is offered for the behavior of combination such weak components. They have been analyzing combination accuracy as a function of parameters, which control the power and resolution of component partitions.

The authors of [8] suggested a method that allows a segmentation of multispectral images through a scalar approach. The suggested method proceeds in three steps. The first step tempts to eliminate redundant observations by maximizing an entropy criterion. Scalar segmentations via an automatic multi-thresholding technique are applied on relevant bands, in the second step. Finally, a fusion of the multi-thresholding results is achieved in the last step to provide the final segmentation.

In this paper, we present an alternative method for improving image segmentation. The proposed method is focused on decision fusion using parallel architectures and based on combining the decisions made by multiple segmentation methods to develop the segmentation accuracy over than a single segmentation technique. The individual segmentation methods are capable of independent and simultaneous operations. However, our work is different from that carried by the authors [4-8] in a number of aspects. Our interest here is the segmentation of medical and non-medical images, while the most other works [7, 8] were devoted to other domains such as remote-sensing image classification. This is because the importance of medical image analysis demands quality segmentation which the single method can not provide. Also, the art of the fusion is improved by fusing different methods applied to huge number of clusters at the same time. For example, in order to apply the different fusion rules on segmentation methods, some of which produce crisp outputs. Thus we need to convert the hard decisions to soft.

Moreover, our research spans a wider mix of various segmentation methods covering the key classical methods and the rather more-recent techniques based on fuzzy clustering [9,10] that have enjoyed noticeable success in that field.

The rest of the paper is organized as follows. Several segmentation methods are reviewed in section II. Section III presents some decision fusion methods. Experimental results are presented in section IV. In section V, we present our conclusions and future work.

II. Image Segmentation Techniques

In this section, we briefly present several existing key segmentation methods, which will be combined in fusion methods.

A. Histogram thresholding

Histogram thresholding is one of the oldest techniques for image segmentation. It assumes that images are composed of regions with different gray level ranges, the histogram of an image can be separated into a number of peaks (modes), each corresponding to one region, and there exists a threshold value corresponding to valley between the two adjacent peaks.

A thresholding procedure attempts to determine an intensity value, called the *threshold*, which separates the desired classes. The segmentation is then achieved by grouping all pixels with intensity between two such thresholds into one class. This method relies on the identification of a good threshold. Failing to find such a threshold may lead to poor segmentation. In addition, thresholding typically does not take into account the spatial characteristics of an image. This causes it to be sensitive to noise and intensity inhomogeneities. A rather recent strategy has been proposed by Bonnet [11] to overcome the difficulty of identifying a good threshold.

B. Region growing

The region growing method is a well-developed technique for image segmentation [12]. It postulates that neighboring pixels within the same region have similar intensity values. The general idea of the region growing method is to group pixels with the same or similar intensities to one region according to a given homogeneity criterion. More precisely, the region growing method starts with a set of pre-specified seed pixels and grows from these seeds by merging neighboring pixels that satisfy a homogeneity criterion. The procedure is iterative; this process is repeated until no more pixels are merged.

The region growing method can utilize the relationships of neighboring pixels and is not sensitive to the ill-balanced data problem. However, the problem with region growing is its inherent

dependence on the selection of seeds (often done manually), and how to set the homogeneity criterion.

C. K-means clustering method

K-means clustering, also known as hard c-means clustering, is one of the simplest unsupervised classification algorithms. The procedure follows a simple way to classify the dataset through a certain number of clusters. The algorithm partitions a set of N vector $X=\{x_j, j=1,...,N\}$ into C classes c_i, $i=1, \ldots, C$, and finds a cluster centre for each class v_i denotes the centroid of cluster c_i such that an objective function of dissimilarity, for example a distance measure, is minimized. The objective function that should be minimized, when the Euclidean distance is selected as a dissimilarity measure, can be described as:

$$P = \sum_{i=1}^{C} \left(\sum_{k, x_k \in c_i} \|x_k - v_i\|^2 \right), \quad (1)$$

where $\sum_{k, x_k \in c_i} \|x_k - v_i\|^2$ is the objective function within group i, and $\|x_k - v_i\|^2$ is a chosen distance measure between a data point x_k and the cluster centre v_i.

The partitioned groups are typically defined by a $(C \times N)$ binary membership matrix $U=(u_{ij})$, where the element u_{ij} is 1 if the j-th data point x_j belongs to group i, and 0 otherwise. That means:

$$u_{ij} = \begin{cases} 1 & if \|x_j - v_i\|^2 \le \|x_j - v_k\|^2, \text{ for each } k \ne i \\ 0 & otherwise \end{cases} \quad (2)$$

$$v_i = \frac{\sum_{x_j \in v_i}^{N} x_j}{R_i} \quad (3)$$

where R_i is number of data point in class c_i.

The K-means clustering algorithm consists of the following steps:

Step 1: Initialize the cluster centers v_i, $i=1,...,C$. This is typically achieved by randomly selecting C points from among all of the data.

Step 2: Determine the membership matrix U according to Equation (2).

Step 3: Compute the objective function according to Equation (1). Stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold.

Step 4: Update the cluster centers v_i, i=1,..., C using Equation (3).

Step 5: Go to step 2.

The algorithm is rather sensitive to the initial randomly selected cluster centres. In addition, the number of clusters has to be supplied beforehand. However some methods (e.g., [13]) have been tried to estimate this number automatically.

D. Fuzzy C-means clustering method

Fuzzy c-means clustering (FCM), also known as fuzzy ISODATA, is a data clustering algorithm in which each data point belongs to a cluster to determine a degree specified by its membership grade. Bezdek [14] has proposed this algorithm as an alternative to earlier K-means clustering. FCM partitions a collection of N vector x_i, $i=1,...,N$ into C fuzzy groups, and finds a cluster centre in each group such that an objective function of a dissimilarity measure is minimized. The major difference between FCM and K-means is that FCM employs fuzzy partitioning such that a given data point can belong to several groups with the degree of belongingness specified by membership grades between 0 and 1. In FCM, the membership matrix U is allowed to have not only 0 and 1 but also the elements with any values between 0 and 1, this matrix satisfies the constraints:

$$\sum_{i=1}^{C} u_{ij} = 1, \forall j = 1,..., N \quad (4)$$

The objective function of FCM can be formulated as follows:

$$P(u, c_1, ..., c_C) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m \|x_j - v_i\|^2, \quad (5)$$

where u_{ij} is between 0 and 1; v_i is the cluster centre of fuzzy group i, and the parameter m is a weighting exponent on each fuzzy membership (in our implementation, we take it 2). Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, updating of membership u_{ij} and the cluster centres v_j by:

$$v_i = \frac{\sum_{j=1}^{N} u_{ij}^m x_j}{\sum_{j=1}^{N} u_{ij}^m}. \quad (6)$$

$$u_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{x_j - v_i}{x_j - v_k} \right)^{2/(m-1)}}. \quad (7)$$

The FCM clustering algorithm is composed of the following steps:

Step 1: Initialize the membership matrix U with random values between 0 and 1 such that the constraints in Equation (4) are satisfied.

Step 2: Calculate fuzzy cluster centers v_i, i=1,..., C using Equation (6).

Step 3: Compute the cost function (objective function) according to Equation (5). Stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold.

Step 4: Compute a new membership matrix U using Equation (7).

Step 5: Go to step 2.

D. Kernelized fuzzy c-means method

The kernel methods [15] are one of the most researched subjects within machine learning community in the recent few years and have widely been applied to pattern recognition and function approximation. The main motives of using the kernel methods consist of: (1) inducing a class of robust non-Euclidean distance measures for the original data space to derive new objective functions and thus clustering the non-Euclidean structures in data; (2) enhancing robustness of the original clustering algorithms to noise and outliers, and (3) still retaining computational simplicity.

The algorithm is realized by modifying the objective function in the conventional fuzzy c-means (FCM) algorithm using a kernel-induced distance instead of Euclidean distance in the FCM, and thus the corresponding algorithm is derived and called as the kernelized fuzzy c-means (KFCM) algorithm, which to be more robust than FCM.

In this work, the kernel function $K(x, V)$ is taken as the Gaussian radial basic function (GRBF):

$$K(x, v) = \exp\left(\frac{-\|x - v\|^2}{\sigma^2} \right), \quad (8)$$

where σ is an adjustable parameter. The objective function is given by:

$$j_m = 2 \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m (1 - K(x_j, v_i)) \cdot \quad (9)$$

The fuzzy membership matrix u can be obtained from:

$$u_{ij} = \frac{\left(1 - K(x_j, v_i)\right)^{-1/(m-1)}}{\sum_{k=1}^{C} \left(1 - K(x_j, v_k)\right)^{-1/(m-1)}}. \quad (10)$$

The cluster center v_i can be obtained from:

$$v_i = \frac{\sum_{j=1}^{N} u_{ij}^{\ m} K(x_j, v_i) x_j}{\sum_{j=1}^{N} u_{ij}^{\ m} K(x_j, v_i)}. \quad (11)$$

The proposed KFCM algorithm is almost identical to the FCM, except in Step 2, Eq.(11) is used instead of Eq.(6) to update the centers, and in step 4 Eq.(10) is used instead of Eq.(7) to update the memberships.

E. Spatial constrained SKFCM method

SKFCM is applied directy to image segmentation like FCM, it would be helpful to consider some spatial constraints on the objective function. This penalty term contains spatial neighborhood information, which acts as a regularizer and biases the solution toward piecewise-homogeneous labeling. Such regualization is helpful in segmenting images corrupted by noise. The objective function is as follows:

$$j_m = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^{m}\left(1 - K(x_j, v_i)\right) + \frac{\alpha}{N_R}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^{m}\sum_{r\in N_j}(1-u_{ir})^m \quad (12)$$

where N_j stands for the set of neighbors that exist in a window around x_j (do not include x_j itself) and N_R is the cardinality of N_j. The parameter α controls the effect of the penalty term and lies between zero and one inclusive.

An iterative algorithm for minimizing Eq.(12) is derived by evaluating the centroids and membership functions that satisfy a zero gradient condition like the KFCM. A necessary condition on u_{ij} for Eq.(12) to be at a local minimum or or saddle point is:

$$u_{ij} = \frac{\left((1-K(x_j, v_i)) + \left(\alpha\sum_{r\in N_j}(1-u_{ir})^m / N_R\right)\right)^{-1/(m-1)}}{\sum_{k=1}^{C}\left((1-K(x_j, v_k)) + \left(\alpha\sum_{r\in N_j}(1-u_{kr})^m / N_R\right)\right)^{-1/(m-1)}} \quad (13)$$

The proposed SKFCM algorithm is almost identical to the KFCM, except in Step 4, Eq.(13) is used instead of Eq.(7) to update the memberships.

Two major problems are known with the FCM, KFCM, SKFCM methods: (1) How to determine the number of clusters. (2) The computational cost is quit high for large data sets. Some improvements over these methods are proposed in [13].

III. Decision Fusion

Once the set of segmentation has been created, an effective way of combining their outputs must be found. Fusion of multiple methods can be performed either at data level or at the decision level. We focus in this paper on decision fusion using parallel architectures. The problem can then be stated as follows. The goal is to assign each image point x to one class label ω_i out of C class labels $\Omega=\{R_1, \ldots, R_C\}$. To accomplish this task, a set of L segmentation methods may be consulted. The output of each method can be one of two types:

- Class label (hard output): $Z_i \in \Omega$.
- Soft output: A C-element vector $Z_i = [z_{i,1}, \ldots, z_{i,C}]^T$ which represents the supports to the C classes. A special case of this vector is a probability distribution over Ω estimating the posterior probabilities $P(R_i / x)$, i=1,…,C (shortened as $P(R_i,x)$ in the sequel).

There are many decision fusion methods for each type of outputs. In this paper, we will use several of them, namely, the popular voting methods [16] for hard outputs and the minimum, maximum, median, and product rules for soft outputs.

A. Voting scheme

Voting strategies [16] can be applied to a multiple classifier system. An input sample is assigned the class for which there is a consensus of the L segmentation methods agreed on its identity. Otherwise, the sample is rejected. Numerous voting techniques have been presented in the literature, the most popular of which is the majority vote. In majority voting method, the probability among the classifiers are equal and does not require any parameters to be trained. The differentiation between the classes is the number of votes they have received. Assuming a lack of extra information, a tie can only be broken by a random draw. However, in the complete agreement mode, when all the individual classifiers agree on the classification of a sample, the sample is classified to that class, otherwise, the sample is rejected.

B. Fusion techniques for soft labels

The largest group of fusion methods combines soft decisions. Most popular among them are the minimum, maximum, mean, median and product fusion rules, which are defined as follows [16].

a) Median rule: The rule assigns x to R_i class where $P_{med}(R_i,x)$ is maximum:

$$P_{med}(R_i, x) = \underset{k=1}{\overset{L}{med}}\ u_{i,j}(x) \quad (13)$$

b) Mean rule: The rule assigns x to R_i class where $P_{mean}(R_i,x)$ is maximum:

$$P_{mean}(R_i, x) = \underset{k=1}{\overset{L}{mean}}\ u_{i,j}(x) \quad (14)$$

c) Maximum rule: The rule assigns x to R_i class where $P_{\max}(R_i,x)$ is maximum:

$$P_{\max}(R_i, x) = \max_{k=1}^{L} u_{i,j}(x) \quad (15)$$

d) Minimum rule: The rule assigns x to R_i class where $P_{\min}(R_i,x)$ is maximum:

$$P_{\min}(R_i, x) = \min_{k=1}^{L} u_{i,j}(x) \quad (16)$$

e) Product rule: The rule assigns x to R_i class where $P_{prod}(R_i,x)$ is maximum:

$$P_{prod}(R_i, x) = \prod_{k=1}^{L} u_{i,j}(x) \quad (17)$$

C. Converting hard decisions to soft

To apply the different fusion rules on segmentation methods, some of which produce crisp outputs (e.g., region growing). One thus needs to convert in this case hard decisions to soft, which is discussed next.

A membership function is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. Gaussian membership function is a popular method for specifying fuzzy sets. Gaussian models for $p(x/R_i)$ are popular because of their mathematical tractability given by:

$$p(x/R_i) = \frac{1}{\sqrt{2\Pi}\sigma} e^{\frac{-(x-\mu)^2}{2\sigma^2}}; \quad (18)$$

where $\mu = \frac{1}{R_i}\sum_{x\in R_i} x$, $\sigma^2 = \frac{1}{|R_i|-1}\sum_{x\in R_i}(x-\mu)^2$

The a priori estimate of the probability of a certain class is converted to a posterior, or measurement conditioned through Bayes theorem, probability of a state of nature via:

$$P(R_i / x) = \frac{p(x/R_i)P(R_i)}{P(x)}, \quad (19)$$

since $\sum_i P(R_i/x) = 1$, then from eq.(19) we get $\sum_i p(x/R_i) = \frac{p(x)}{P(R_i)}$

where $P(R_i)$ are called a priori or prior probabilities (equal probability), and $p(x/R_i)$ the class-conditional probability density, which describes the chance of finding a feature vector from class R_i at the position x.

IV. Experimental results

The experiments were performed with several data sets. The first experiment consists of two simple synthetic images (synthetic1 and synthetic2), one corrupted by 9% salt and pepper noise, and another corrupted by gaussian noise of standard deviation 50 respectively, and the image size is 142×145 pixels, as shown in Fig.1(a), and Fig.1(b), respectively. The second set includes simulated volumetric MR data consisting of ten classes. The advantages for using digital phantoms rather than real image data for validating segmentation methods include prior knowledge of the true tissue types and control over image parameters such as modality, slice thickness, noise and intensity inhomogeneities. We used a high-resolution T1-weighted MR phantom with slice thickness of 1mm, 3% noise and no intensity inhomogeneities, obtained from the classical simulated brain database of McGill University [17]. Two slice drawn from the simulated MR data is shown in Figs. 1(d) and 1(e).

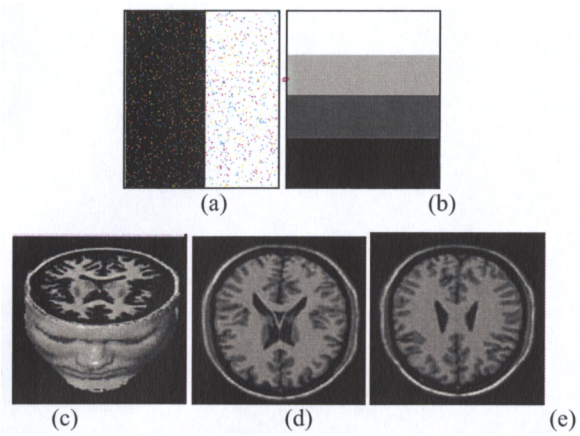

Fig.1: Test images: (a) Synthetic 1, (b) Synthetic 2, (c) 3D simulated data, (d) and (e) two original slice from the 3D simulated data (slice91 and slice100).

The quality of the segmentation algorithm is of vital importance to the segmentation process. The comparison score S for each algorithm is proposed in [15], which defined as:

$$S = \left| \frac{A \cap A_{ref}}{A \cup A_{ref}} \right| \quad (20)$$

where A represents the set of pixels belonging to a class as found by a particular method and A_{ref} represents the set of pixels belonging to the very same class in the reference segmented image (ground truth).

Image segmentation results

Here, the aforementioned segmentation methods have been implemented. For the techniques of region growing and histogram thresholding, the seeds and thresholds were selected manually with great care. The Gaussian RBF kernel is used for KFCM and SKFCM. We set the parameters m=2, σ=150, α=0.7 and N_R=26 when using 3D MR phantom image, because the add noise is relatively big, otherwise we use α=0.1, and N_R=8 (a 3 × 3 window centered a round each pixel). These values will be used in the rest of this work if no specific value is explicitly stated.

Experiment on synthetic1

We applied these algorithms to a synthetic test image; the synthetic image contains a two classes pattern corrupted by 9% salt and pepper noise. The performance of each segmentation method on this dataset is reported in the upper part of the first column of Table I. The table shows that the highest segmentation accuracy is obtained using region growing. However their accuracy depends largely on how well the initial seeds are manually selected. After that SKFCM gives better results than the other methods, and the other methods are obtained similar accuracy.

Experiment on synthetic2

The performance of each segmentation method on the 4-class synthetic image synthetic2 is reported in the upper part of the second column of Table I. Obviously, histogram thresholding acquires the best segmentation performance, and the least segmentation accuracy is obtained by applying the region growing method. Note FCM, KFCM, SKFCM and k-means give similar accuracy.

We tested the efficiency of the accuracy for a synthetic2 image with various degrees of standard deviation of gaussian noise. Fig.(2) depicts the relationship between accuracy results when FCM, KFCM, SKFCM, histogram and region growing are applied to synthetic2 image and various degree of standard deviation of gaussian noise. Note, region growing the most segmentation method affected by noise.

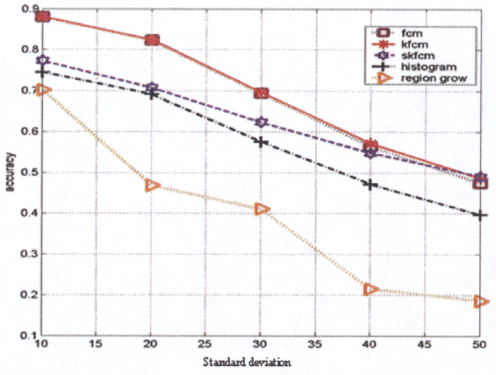

Fig.2: The relation between accuracy and standard deviation, when FCM, KFCM, SKFCM, histogram and region grow are applied on synthetic2 image.

Experiment on the simulated 3D data.

The upper part of the 3rd column of Table I shows the corresponding accuracy scores of the individual segmentation methods after applying them on the simulated data. Obviously, k-means and histogram acquires the best segmentation performance, and the other methods gave similar accuracy.The outcomes from the individual image segmentation techniques have been fed to the fusion algorithms.

The lower part of the first and second columns of Table I show the performance of each fusion method on synthetic1 and synthetic2, respectively. The table shows that the highest segmentation accuracy is obtained using the mean fusion rule. It gives an improvement about 2.6% and 3.7% over the accuracy of the best individual segmentation method and an improvement about 6.7% and 4.4% over the average segmentation's accuracy for synthetic1 and synthetic2, respectively. The average fusion accuracy improvement is showed to be about 6.5% and 4.2% higher than the average segmentation accuracy for synthetic1 and synthetic2, respectively.

The lower part of the 3rd column of Table I shows the performance of each fusion scheme on the MRI dataset. The table shows that the highest segmentation accuracy is obtained using the maximum fusion rule; it is obvious that the mean fusion rule exhibited similar value on this set. The least performance is obtained applying the minimum and product fusion rules. The maximum fusion rule gives an improvement about 1.7 % over the best segmentation accuracy and an improvement about 2.8 % over the average segmentation accuracy, where the average fusion accuracy is computed here by averaging all the fusion methods except the product and minimum rules, which gave the worse results. The average fusion accuracy improvement is showed to be about 2.4% higher than the average segmentation accuracy, again not including the product and minimum rules. The result also shows

that we get better accuracies using the fusion soft output as it contains more information than the hard outputs.

Weak segmentation

In this experiment, the same set of segmentation methods is used without going through the tuning phase to reach the best possible accuracy. Instead the segmentation methods are used deliberately at worse performance accuracy. For example, this can be done by selecting the initial seeds and number of clusters carelessly. The accuracies of intentionally made weak segmentation methods on two datasets are listed in the upper part of the last two columns of Table I for the synthetic2, and simulated 3D data, respectively.

Applying the same fusion schemes on the test set for this set of segmentation methods, the results are reported in the upper part of the last two columns of Table I for the synthetic2 and simulated 3D data, respectively. This table shows that the highest segmentation accuracy is obtained using mean scheme and the least is obtained by applying the minimum or product fusion rule.

This Table shows that the highest segmentation accuracy is obtained using mean scheme and the least is obtained by applying the minimum or product fusion rule. For the synthetic2 image, the mean fusion rule gives an improvement about 5.43% over the best segmentation method accuracy and an improvement about 16.23% over the average segmentation accuracy. The average fusion accuracy improvement is showed to be about 2.76% higher than the average segmentation accuracy.

For the simulated 3D data, the mean fusion rule gave an improvement about 6.68% over the best segmentation accuracy and an improvement about 10.6% over the average segmentation accuracy. The average fusion accuracy improvement is shown to be about 5.76% higher than the average segmentation accuracy.

Table I : Segmentation accuracy of individual methods and performance of implemented fusion techniques on synthetic1, synthetic2, and MRI volume dataset.

	Method	Accuracy				
		Well-tuned Methods			Weak Methods	
		Synthetic 1	Synthet 2	MRI volume	Synthetic 2	MRI volume
Individual methods	FCM	0.91615	0.832537	0.52531	0.44609	0.34495
	KFCM	0.91597	0.835839	0.53341	0.47152	0.37227
	SKFCM	0.95286	0.835316	0.54708	0.20227	0.34703
	Region	0.97320	0.693539	0.54842	0.43607	0.35372
	Histogram	0.91704	0.842303	0.55275	0.44721	0.40460
	K-means	0.91776	0.826830	0.55394	0.53383	0.36929
Fusion techniques	Voting	0.99389	0.878961	0.56309	0.52684	0.46103
	Max	0.99854	0.870248	0.57182	0.35748	0.39077
	Min	0.99438	0.870338	0.13433	0.35896	0.40123
	Mean	0.99980	0.880148	0.57075	0.58813	0.47150
	Median	0.99980	0.880056	0.56633	0.57429	0.42412
	Product	0.99873	0.880148	0.13452	0.35909	0.39167

V. CONCLUSION

An alternative methodology for improving image segmentation has been presented. Rather than tuning a method for the best possible performance. It resorts to combining several methods via decision fusion to improve the segmentation accuracy. Our experiments have shown that decision fusion can indeed improve the segmentation accuracy over the best/average individual segmentation methods accuracy. Moreover, the mean fusion rule has exhibited the best overall performance in all our experiments, outperforming all other schemes. The mean rule has given an improvement about 10.6% over the segmentation average accuracy for a set of intentionally- made weak methods for segmentation. Our results have also stressed that the gain from decision fusion for such a weak set, without any efforts for performance optimization or tuning, can even be more evident than for a set of very-accurate members.

For our future work we will investigate more fusion methods and other approaches for image segmentation. Our ongoing work includes a further, thorough objective and subjective evaluation of the improvement gained from decision fusion over the individual segmentation techniques, via the help of specialized medical doctors.

REFERENCES

[1] A. A. Farag, R. M. Mohamed, and H. Mahdi, "Experiments in Image Classification and Data Fusion," Proceedings of 5th International Conference on Information Fusion, Annapolis, MD, vol. 1, pp. 299-308, 2002.

[2] L.I. Kuncheva, "Switching between selection and fusion in combining classifiers: An experiment," IEEE transactions on Systems Man and Cybernetics, Part B-cybernetics, vol. 32, pp. 146-156, 2002.

[3] L.I. Kuncheva, J. Bezdek, and M. Sutton, "On combining multiple classifiers by fuzzy templates," In: Proceedings of the 1998 Annual Meeting of the North American Fuzzy Information Processing Society, NAFIPS'98, Pensacola FL, pp. 193-197, 1998.

[4] D. Frossyniotis, M. Pertselakis, and A. Stafylopatis, "A multi clustering fusion algorithm," In: Proceedings of the Second Hellenic Conference on Artificial Intelligence, April 11–12 LNAI 2308. Springer-Verlag, Thessaloniki, Greece, pp. 225–236, 2002.

[5] A. Fred, "Finding consistent clusters in data partitions," In: Proceedings of the Second International Workshop on Multiple Classifier Systems (MCS 2001) Lecture Notes in Computer Science, 2096. Springer, Cambridge, UK, pp. 309–318, 2001.

[6] F. Olivier, and M. Ali, "A hidden Markov Model for image fusion and their joint segmentation in medical image computing," arXiv:physics/0403150, vo1.31, Mar 2004

[7] T. Alexander, K.J. Anil, and P. William, "Combining Multiple Weak Clusterings," Proc. 3d IEEE Intl. Conference on Data Mining, pp.331-338, 2003.

[8] C. Kermad, and K. Chehdi, "Multi-Bands Image Segmentation: A Scalar Approach", 2000.

[9] H. Arefi, M. Hahn, F. Samadzadegan, "Comparison of clustering techniques applied to laser data, Geo-imagery Bridging continents", xx-th ISPRS congress, Istanbul, Turkey, 2004.

[10] E.A.Zanaty, "Image segmentation in medical data", Science Echoes Journal ,7, 14-22, 2006.

[11] N. Bonnet, and J. Cutrona, "A 'no-threshold' histogram-based image segmentation method," Pattern Recognition Letters, vol.35, pp.2319–2322, 2002.

[12] M. Matei, and G. Bernard, "Segmentation using a region growing thresholding," Proc. Of the Electronic Imaging Conference of the International Society for Optical Imaging (SPIE/IS&T 2004), SanJos (California, USA), 2005.

[13] E.A.Zanaty, M.T.El-Melegy, M.R.Girgis and Walaa M.Abd-Elhafiez," On cluster validity indexes in fuzzy and hard clustering algorithms for image segmentation" IEEE international conference on computer vision, Vol. 6, VI 5-8, 2007.

[14] J.C. Bezdek, "Some non-standard clustering algorithms," In: Legendre, P. & Legendre, L. *Developments in Numerical* Ecology. NATO ASI Series, *G14*. Springer- Verlag, 1987.

[15] D.Q. Zhang, and S.C. Chen, "A novel kernelized fuzzy c-means algorithm with application in medical image segmentation," Artif. Intell. Med, vol.32, pp.37–50, 2004.

[16] L. Lam, and C.Y. Suen, "Application of Majority Voting to Pattern Recognition: an Analysis of Its Behavior and Performance," IEEE Transactions on Systems, Man, and Cybernetics, vol. 27, No. 5, 1997.

[17] BrainWeb: Simulated Brain Database. McConnell Brain Imaging Centre. Montreal Neurological Institute, McGill .