Combining stationary wavelet transform and self-organizing maps for brain MR image segmentation

Ayşe Demirhan*, Inan Güler

Department of Computer Engineering, Faculty of Technology, Gazi University, 06500 Teknikokullar, Ankara, Turkey

Article info

Article history:
Received 26 January 2010
Received in revised form
5 July 2010
Accepted 10 September 2010
Available online 24 September 2010

Keywords:
Brain MR images
Image segmentation
Stationary wavelet transform
Self-organizing maps
Learning vector quantization
Tanimoto similarity index

Abstract

This study presents an image segmentation system that automatically segments and labels T1-weighted brain magnetic resonance (MR) images. The method is based on a combination of unsupervised learning algorithm of the self-organizing maps (SOM) and supervised learning vector quantization (LVQ) methods. Stationary wavelet transform (SWT) is applied to the images to obtain multiresolution information for distinguishing different tissues. Statistical information of the different tissues is extracted by applying spatial filtering to the coefficients of SWT. A multidimensional feature vector is formed by combining SWT coefficients and their statistical features. This feature vector is used as input to the SOM. SOM is used to segment images in a competitive unsupervised approach and an LVQ system is used for fine-tuning. Results are evaluated using Tanimoto similarity index and are compared with manually segmented images. Quantitative comparisons of our system with the other methods on real brain MR images using Tanimoto similarity index demonstrate that our system shows better segmentation performance for the gray matter while it gives average results for white matter.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction

MR imaging is a widespread method that is used to obtain high quality medical images. It is one of the popular, painless and noninvasive brain imaging techniques. Especially for brain imaging MR imaging reveals a unique view by providing high level spatial and contrast resolution. It is used for diagnosis of many diseases and gives high quality informative images of the inside structure of the brain. Multispectral images of tissues that have different contrast values such as T1-weighted, T2-weighted, PD and FLAIR are provided by diverse magnetic resonance parameters (Robb, 2000).

Image segmentation refers to the process of partitioning a digital image into multiple regions. The goal of segmentation is to simplify and change the representation of an image into something that is more meaningful and easier to analyze (Shapiro and Stockman, 2001). In medical image segmentation, different image components are used for analysis of different structures and tissues, spatial distribution of functional activities and pathologic regions. Segmentation of brain MR images consist of peeling the brain from skull and classifying as brain/not-brain or segmenting tissue parts as white matter, gray matter, cerebrospinal fluid or suspicious region (Rogowska, 2000; Egmont-Petersen et al., 2002).

For a great many clinical applications, it is a classical problem to segment brain images into the tissue types such as white matter, gray matter and cerebrospinal fluid. In addition to this, regions of the brain such as tumor, edema and lesion for diseased or injured people should be determined for the purpose of diagnosis and planning surgical operations (Alienze et al., 1998; Reddick et al., 1997; Zizzari et al., 2001; Morra et al., 2003).

Medical image segmentation is usually performed manually. Areas of interest are drawn by expert radiologists and doctors. It is a time consuming and tiring process. Manual segmentation is not objective. Different segmentations done by different experts can be very different. The segmentation that is done by the same expert can be different under different circumstances. The brightness and contrast of the screen can affect the segmentation accuracy and the following analyses. Usage of computers for segmentation can overcome these problems (Güler et al., 2009).

In the medical image segmentation; thresholding, region-based segmentation, edge-based segmentation and classification based segmentation are the most frequently used techniques. Two main limitations are incorporated with thresholding: firstly, only two classes are generated; secondly it cannot be applied to multichannel images. In addition to these, thresholding typically does not take into account the spatial characteristics of an image. This causes it to be sensitive to noise and intensity inhomogeneity, which can frequently occur in MR images. Therefore, variations on classical thresholding have been proposed for medical image segmentation that combines information based on local intensities and connectivity (Pham et al., 2000). Region based segmentation algorithms are region growing algorithm (Kaus et al., 1999; Yu et al., 1992), divide-and-combine
Several studies are performed that use SOM for the purpose (Chen and Pavlidis, 1980) and watershed algorithm (Grau et al., 2000). The main drawback of histogram-based region segmentation is that the histogram provides no spatial information. Region growing algorithm approaches exploit the important fact that pixels which are close to each other have similar gray values (Boudraa and Zaidi, 2005). The requirement of manual interaction to obtain a seed point is the main disadvantage of region growing algorithm. A seed point must be planted for each region to be extracted. Region growing can be sensitive to noise which causes extracted regions to have holes or become disconnected. Divide-and-combine is an algorithm related to region growing, but it does not require a seed point, and has the same drawbacks (Pham et al., 2000). The watershed transformation constitutes one of the most powerful segmentation tools provided by mathematical morphology. But there are two disadvantages in the watershed algorithm. Firstly, classical watershed algorithm on gray images such as tissue images causes oversegmentation. Secondly, there are some regions which are not divided completely particularly in the transition regions of gray matter and white matter, or cerebrospinal fluid and gray matter (Kong et al., 2006). Edge based algorithms work with edge finders. Traditional Sobel and Laplace detectors can be used for this purpose (Gonzalez and Woods, 2002). Edge detection works well on images with good contrast between regions. However, the detection is limited in regions with low contrast. Furthermore, it is difficult to find correlation between the detected edges of the regions of interest (Boudraa and Zaidi, 2005). Classification based algorithms can be constructed according to brightness similarity, contour energy and curvilinear continuousness. Examples of classifiers are the k-nearest neighbor (Boer et al., 2010) and support vector machine (Chaplot et al., 2006). Classification based segmentation needs training. Performance of the classification based segmentation depends on inputs of the classifier and training parameters (Toulson and Boyce, 1992). Knowledge-based expert systems are another methodology that is used for segmentation of images (Matesin et al., 2001; Li et al., 1993). These systems require knowledge engineering and domain experts for building and maintaining the rules and the system. Brain MR images are very complex to be efficiently described and handled using rules. Another widely used method for brain image segmentation is the fuzzy c-means algorithm. This algorithm is not efficient by itself, because it fails to deal with the significant property of images that neighbor pixels are strongly correlated. Ignoring the correlation of neighboring pixels leads to strong noise sensitivity and several other imaging artifacts (Szlágyi et al., 2007).

Neural networks are one of the classification based segmentation methods. They perform classification by a method that learns from data, instead of using a given rule set. They organize themselves in a data driven manner. Neural networks draw attention consistently due to their ability of self-learning, fault tolerance and capability of searching for the optimum. Neural networks are constituted from lots of nonlinear computation elements that work in parallel and they are organized in a design that is similar to biological neural networks. Neural networks change their response according to the environmental conditions, learn from their experience and perform generalizations from old samples. SOM is an unsupervised neural network that uses competitive learning algorithm. It is one of the most popular networks in the neural network field. SOM networks have advantages like they can automatically form similarity diagrams according to the input data (Chaplot et al., 2006). SOM maps high dimensional inputs to one or two dimensional discrete lattice of neurons. It organizes input data into several patterns according to a similarity factor like Euclidean distance. SOM learns both the distribution and the topology of input data. In other words, the network preserves topological relationships in its input and maps neighbor inputs to neighbor neurons (Tian et al., 2007; Balafar et al., 2010). Several studies are performed that use SOM for the segmentation of brain MR images (Egmont-Petersen et al., 2002; Guler et al., 2009; Tian et al., 2007; Glass et al., 2004; Vijayakumar et al., 2007). Following an unsupervised learning process accomplished by a SOM network, usually a vector quantization method, LVQ, is used to calibrate the neurons of the network to find their best location (Kohonen, 2002).

The nonlinear character of MR images makes classical statistical methods particularly inappropriate for use for the segmentation process. The major advantage of neural networks over classical statistical pattern recognition techniques is their relative insensitivity to the selection of the training sets. This property has been shown to be of importance both for single slice and multiple slice classification. Neural networks do not rely on any assumption about the underlying probability density functions, thus possibly improving the results when the data significantly depart from normality (Ozkan et al., 1993). Based on the success of the neural network algorithm in pattern recognition and classification of different tissues in terms of texture, intensity or contrast, SOM algorithm is used in this study for the segmentation task.

Variations in the magnetic fields of the scanner cause image intensity inhomogeneities called bias field or non-uniformity. The result is multiplication of image intensities with a bias that slowly varies spatially. This causes problems for the tasks such as registration and segmentation. To overcome this problem space-invariant filtering techniques like low-pass filtering or neighborhood averaging are applied to the image. However, these filtering methods blur the important features in the image while suppressing the noise. Space-variant filtering techniques aim to address this limitation by using local, feature-dependent strategies. The approaches include recursive low-pass filtering with adaptive coefficients, linear least-squares error filtering, local shape-adaptive template filtering and anisotropic diffusion filtering. Among these techniques anisotropic diffusion filtering technique improves the signal-to-noise ratio, blurs homogeneous regions and preserves the boundaries and interesting structures (Perona and Malik, 1990; Gerig et al., 1992; Bayram et al., 2001). Therefore to eliminate bias field and random noise, anisotropic diffusion filtering is used as a preprocessing step in this study.

Accomplishing a true and reliable segmentation of the brain MR images depends on the selection of the best features that represent different tissue types and that will be used as input to the system. Multiresolutional wavelet analysis provides the subimages of an image localized in different spatial frequencies (Gonzalez and Woods, 2002). It divides the 2D frequency spectrum of an image into one lowpass (approximation) and three highpass (horizontal, vertical and diagonal) subimages. Due to multiresolutional modeling capacity of the wavelets, obtained subimages are in different scales and frequencies. Because these images have different characteristics, it is quite convenient to use them for distinguishing tissue types from each other and tissue analysis (Tsai and Hsiao, 2001).

Wavelet coefficients alone are insufficient for describing tissue properties exactly. Even though they are useful for splitting the textured information into different frequency channels, they lack the local statistical information around a given pixel. Therefore, usually a spatial filtering operation is applied to the wavelet coefficients by sliding a window through the coefficients. Thus local statistics, which are mean absolute deviation (MAD), entropy, energy and standard deviation that describe texture properties are calculated. MAD is the mean of absolute deviations of the data. It represents the uniformity of the textures. Entropy measures the randomness of the texture. Energy indicates if the texture is broader, coarser or finer. Energy and entropy can differentiate homogeneous and non-homogeneous regions. Standard deviation is a measure of average contrast.

In this study, an anisotropic filtering preprocess is applied to brain MR images for the purpose of segmentation. By this way, the
qualities of the images are improved. Then SWT is applied to the images. Diverse features are extracted from the subimages obtained from the transform, by applying the spatial filtering process. These features are combined together with the raw wavelet transform coefficients to obtain a feature vector. This feature vector is applied to the SOM network. The network is trained using unsupervised training methodology. LVQ procedure is utilized for calibrating the map. Results obtained from the system are evaluated using similarity indexes and compared with the manually segmented images.

The rest of this paper is organized as follows. In Section 2 materials and methods utilized in this study are described. Brain MR images used in the study are introduced. Employed methods, which are anisotropic diffusion filtering, stationary wavelet transform self-organizing maps and learning vector quantization are presented. Feature extraction methods and evaluation metrics are also explained. Section 3 is dedicated to the computational experiments. Details and parameters of the system are given in this section with the results and discussions. Section 4 gives concluding remarks and addresses the future work.

2. Method and model

2.1. Brain MR images

Simulated and manually segmented MR images play an important role for development of medical image analysis algorithms, especially for segmentation algorithms. Image analysis methods must be tested and evaluated in a controlled environment. Simulated and manually segmented images are very useful tools for validation because the ground truth is known. Segmentation accuracy can be evaluated by comparing the results with these images (Aubert-Broche et al., 2006a, 2006b; McConnell Brain Imaging Centre BIC of the Montreal Neurological Institute, 2010; Cocosco et al., 1997).

The Internet Brain Segmentation Repository (IBSR) is a database that researchers from all around the world contribute. It supplies brain MR images as well as the segmentation results that are performed by the trained experts in a manually guided manner (The Internet Brain Segmentation Repository (IBSR), 2010). MR images of IBSR database are registered T1-weighted 3D head scans. Segmented images supplied by the database are constituted by trained experts in a half automatic manually guided way spending hours. Even though manually segmented images cannot be evaluated as 100% ground truth, it is a good way to compare results of the automatic segmentation methods.

20 normal MR brain data sets and their manual segmentations were provided by the Center for Morphometric Analysis at Massachusetts General Hospital and are available at http://www.cma.mgh.harvard.edu/ibsr/. Manual segmentations include gray matter, white matter and other parts. The images have 3.1 mm slice thickness. 8-bit scaled 3D MRI data files (brain regions only) are used. Each one of these files is 256 × 256 sized (The Internet Brain Segmentation Repository (IBSR), 2010).

2.2. Anisotropic diffusion filter

Perona and Malik (1990) developed a powerful multiscale smoothing and edge detection filter named anisotropic diffusion filter for image processing. This method is mathematically formulated as a diffusion process. It encourages intra-region smoothing against smoothing across boundaries. In anisotropic diffusion filtering method, the estimation about local image structure is guided by knowledge of statistics of the noise degradation and the edge strengths (Gerig et al., 1992).

Smoothing process performed by the filter is suppressed at boundaries using a diffusion coefficient by selecting locally adaptive diffusion strengths. This process can be formulated mathematically as

$$\frac{\partial u(x,t)}{\partial t} = \text{div}(c(x,t)\nabla u(x,t))$$  (1)

The diffusion strength is controlled by $c(x, t)$. The $x$ vector represents spatial coordinates. The variable $t$ is used for enumerating iteration steps. The function $u(x, t)$ is taken as image intensity $I(x, t)$. The diffusion function $c(x, t)$ depends on the magnitude of the gradient of the image intensity. The function $c(x, t) = f(\nabla I(x, t))$ is a monotonically decreasing function. This function diffuses within regions and does not affect region boundaries that are at locations of high gradients (Gerig et al., 1992).

There are two different diffusion functions proposed:

$$c_1(x, t) = e^{-\left(\nabla I(x, t)/\kappa \right)^2}$$  (2)
$$c_2(x, t) = \frac{1}{1 + (\nabla I(x, t)/\kappa)^{\alpha + 2}} \quad \alpha > 0$$  (3)

The parameter $\kappa$ is chosen according to the noise level and the strength of the edges. These diffusion functions attenuate diffusion process for the values that are greater than the parameter $\kappa$ while encouraging the diffusion within the regions (Bayram et al., 2001).

2.3. Stationary wavelet transform

Signal is downsamples and convolved with a filter in traditional wavelet transform to obtain the decomposition of the next level. Size of the decomposed signal is the $1/2^n$ of the original signal. $n$ indicates the level of the decomposition. A pixel in the decomposed signal corresponds to $2^n$ pixels in the original signal. Consequently, pixel based segmentation cannot be accomplished using features extracted from the decomposed signal (Jiang and Zhao, 2003; Kim and Kang, 2007). Unser (1995) defined a new approach called SWT, for the characterization of texture properties at multiple scales using the wavelet transform. The proposed analysis uses an overcomplete wavelet decomposition that is translation invariant. It is proven that this representation constitutes a tight frame and it has a fast iterative algorithm. For this reason SWT is used as the wavelet transform method.

For application of the decomposition, the filter used in decomposition is upsampled for each iteration using Eq. (4) and convolved with the signal to obtain the subsignal of the next level using Eq. (5), instead of applying downsampling process to the signal like in traditional transform methods:

$$h_{i+1}(k) = [h_i]_m k [h_i]_m$$  (4)

Here the notation $[\cdot]_m$ denotes the upsampling by a factor of $m$. The effect of one iteration means dilating the filters $h_i$ and $g_i$ by a factor of 2:

$$s_{i+1}(k) = h_{i+1}(k)s_i(k)$$
$$d_{i+1}(k) = g_{i+1}(k)d_i(k), \quad i = 0,1,\ldots,l$$  (5)

Each processing step involves convolution with the basic filters $h$ and $g$ that are expanded by inserting appropriate number of zeros between filter taps. The complexity of this algorithm is
the same for all iterations and it is proportional to the number of samples.

The subsignals obtained from the SWT have the same length with the original signal and the results are translation invariant. Nonetheless they contain information of the middle frequency region which is very useful for segmenting images.

2.4. Feature extraction

Raw wavelet coefficients alone are inadequate to represent textural features precisely. They are useful for splitting the textured information into different frequency channels. Despite that, they do not involve local statistical information around pixels. There is a need for nonlinearity for distinguishing the texture pairs that have similar brightness values or second order statistics (Acharyya et al., 2003).

Spatial filtering process is applied to the images for computing local features. This process consists of four steps: (1) a center point \((x, y)\) is determined, (2) an operation that involves only the pixels around the predefined neighborhood of the center point is performed, (3) result of the operation is taken as the response of the process at the center point and (4) the process is repeated for every point in the image. Moving the center point in the image creates a new neighborhood for each pixel in the input image. This process is called neighborhood processing or spatial filtering (Gonzales et al., 2004).

Spatial filtering process is applied to the wavelet coefficients and a fixed \(w \times w\) sized window is slid on the coefficients. Local statistics of the window is calculated for every position. Obtained values are associated with the center pixel of the window as a feature value.

There exists a wide variety of textural features. In this study, mean absolute deviation (MAD), entropy, energy and standard deviation are used as textural features. MAD is the mean of absolute deviations of the data in a data set and it is an energy measure that represents the uniformity of the textures. MAD of an \(N\) sized data is computed as

\[
\text{MAD} = \frac{1}{N} \sum_{i=1}^{N} |x_i - \bar{x}|
\]

where \(\bar{x}\) is the mean of the data set and \(x_i\) represents a data sample. Entropy is a statistical measure of randomness and it is used as a feature to characterize the textures of images. It is defined as

\[
\text{Entropy} = -\sum_{i} x_i^2 \log(x_i^2)
\]

Energy is also a statistical feature used for representing textural features. The amounts of variations within a sliding window are represented by the texture energy features. Energy is computed as follows:

\[
\text{Energy} = \frac{1}{N} \sum_{i} |x_i|^2
\]

Eq. (9) formulates the standard deviation which is a measure of average contrast:

\[
\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_i - \bar{x})^2}
\]

2.5. Self-organizing maps

Many algorithms and methods have been developed for image segmentation. Segmentation process that uses a neural network or a set of neural networks relies on processing small areas of an image. Then the decision-making mechanism marks the areas of an image according to the category recognized by the neural network. A type of network designed especially for this is the SOM (Güler et al., 2009).

SOM algorithm maps an input image sized \(N_x \times N_y\) to \(M\) number of regions according to their characteristic features. SOM is trained to learn or set the mapping. It reduces the dimensions of data to a map, helps to understand high-dimensional data and groups similar data together. SOM clusters the data by having several units compete for the current object. The data is entered into the system and neurons are trained by providing information about inputs. The weight vector of the unit closest to the current object becomes the winning unit. A simple SOM consist of two layers. First layer includes input nodes and second layer includes output nodes. Output nodes are in a two-dimensional grid view (Fig. 1). Every input is connected to every output with adjustable weights (Kohonen et al., 2000).

SOM classifier determines input type focusing on \(m\)-dimensional feature vector. In this study, gray-level image (\(I\)) is transformed to a feature vector \(x = [x_0, x_1, ..., x_m] \). Every \(x_i\) is a different feature and \(m\) is the dimension of the feature vector. At the beginning, values of weight vector \(m_i = [\mu_{i1}, \mu_{i2}, ..., \mu_{im}]\) can be random or linear. Weights are adjusted while the network learns. It is a nonlinear projection of the \(p(x)\) (probability density function) of the feature vector \(x\) onto the two-dimensional map. Vector \(x\) is compared with all the weight vectors, \(m_i\), the smallest Euclidian distance \(|x - m_i|\) is defined as the best-matching unit (BMU) or winner node, signified by the subscript \(c\):

\[
|x - m_c| = \min_i(|x - m_i|)
\]

Adjustment of the weight vector for the winning output neuron and its neighbors are calculated as

\[
m_i(t + 1) = m_i(t) + h_{ci}(t)(x(t) - m_i(t))
\]

where \(t = 0, 1, 2, ...\) is an integer of the discrete time coordinate. The \(h_{ci}(t)\) function acts as the neighborhood function. A widely applied neighborhood kernel can be written in terms of the Gaussian function:

\[
h_{ci}(t) = \mathcal{N}(x) \exp\left(-\frac{|r_c - r_i|^2}{2\sigma^2(t)}\right)
\]

where \(\mathcal{N}(x)\) is scalar-valued learning rate factor and the parameter \(\sigma(t)\) defines the width of the kernel which corresponds to the radius of \(N_c(t)\) that refers to a neighborhood set of array points around node \(c\). \(r_c\) and \(r_i\) are the location vectors of nodes \(c\) and \(i\), respectively. Both \(\mathcal{N}(x)\) and \(\sigma(t)\) are monotonically decreasing functions of time (Kohonen, 2002).

Fig. 1. Mapping of feature vector to the output.
A feature vector is used as input to the SOM. First of all, the number of map units is determined using a heuristic formula of $5 \times \text{data length}^{0.544321}$ (Alhoniemi et al., 2010). Then the SOM is initialized using random initialization. After initialization, the SOM is trained in an unsupervised and sequential manner. Sequential training is repetitive like batch training but instead of sending all data vectors to the map for weight adjustment, one data vector at a time is sent to the network.

2.6. Learning vector quantization

LVQ is a supervised vector quantization method. Since LVQ is a statistical classification method, its purpose is to define class regions in the input data space. LVQ optimizes the weight vectors of sending all data vectors to the map for weight adjustment, one by one. Sequential training is repetitive like batch training but instead of sending all data vectors to the map for weight adjustment, one data vector at a time is sent to the network. There are LVQ1, LVQ2 and LVQ3 algorithms developed for the application of the LVQ method (Kohonen, 2002). Let $x(t)$ be an input sample and $m_i(t)$ represent sequential values of the weight vectors $m_i$ in the discrete time domain. The nearest $m_i$ to $x$ is denoted by $m_i$. LVQ1 algorithm is defined by the following equations.

If $x$ and $m_i$ belong to the same class ($x$ is classified correctly):

$$m_i(t+1) = m_i(t) + \alpha(t) [x(t) - m_i(t)]$$  \hspace{1cm} (13)

If $x$ and $m_i$ belong to the different classes ($x$ is classified incorrectly):

$$m_i(t+1) = m_i(t) - \alpha(t) [x(t) - m_i(t)]$$  \hspace{1cm} (14)

If $i \neq c$:

$$m_i(t+1) = m_i(t)$$  \hspace{1cm} (15)

Here $\alpha(t)$ is the learning rate at time $t$. $0 < \alpha(t) < 1$, and $\alpha(t)$ usually decreases monotonically in time (Kohonen, 2002).

LVQ2 algorithm is identical with the LVQ1. In LVQ2 two weight vectors nearest to $x$, $m_i$ and $m_j$ are updated simultaneously. Weight vector $m_i$ should belong to the correct class and $m_j$ should belong to a wrong class. In addition to this, $x$ must fall into a region of values called window. Assume that $d_{1i}$ and $d_{2j}$ are the Euclidian distances of $x$ from $m_i$ and $m_j$, respectively. Under these circumstances $x$ falls in a window of width $w$ as indicated in the following equation (Kohonen, 2002):

$$\min (d_{1i}, d_{2j}) > s, \text{ where } s = \frac{1-w}{1+w}$$  \hspace{1cm} (16)

LVQ2 algorithm is described as follows:

$$m_i(t+1) = m_i(t) - \alpha(t) [x(t) - m_i(t)]$$  \hspace{1cm} (17)

$$m_j(t+1) = m_j(t) + \alpha(t) [x(t) - m_j(t)]$$  \hspace{1cm} (18)

LVQ2 algorithm shifts the decision borders towards the Bayesian limits but does not take the location of $m_i$ into consideration in the long run. Hence, it is necessary to make corrections to make sure that $m_i$ is continued to be brought near to the class distributions. Combining with the previous ideas, if $x$, $m_i$, and $m_j$ belong to the same class for $k \in (i,j)$ the LVQ3 algorithm becomes

$$m_k(t+1) = m_k(t) + \varepsilon \alpha(t) [x(t) - m_k(t)]$$  \hspace{1cm} (19)

The optimal value of $\varepsilon$ depends on the size of the window, being smaller for narrower windows (Kohonen, 2002).

2.7. Evaluation metrics

Characterizing the performance of the brain MR image segmentation algorithms is difficult due to the complexity of neuroanatomical structures, quality of imaging and the need for correct segmentation. Usually Tanimoto (also known as Jaccard) and Dice similarity indexes are used for the purpose of evaluating the performance of the segmentation algorithms. These indexes are the region based coefficients that are measures of spatial overlap (Chang et al., 2009).

Given two sets, $A$ (ground truth) and $B$ (segmentation result), each with $n$ binary label attributes that can take values of 0 or 1. The Tanimoto coefficient is used to measure the overlap that $A$ and $B$ share with their attributes. The following information can be obtained for the combinations of the attributes of $A$ and $B$: true positive (TP) represents the total number of attributes where both $A$ and $B$ have a value of 1, false positive (FP) represents the total number of attributes where the attribute of $A$ is 0 and the attribute of $B$ is 1, false negative (FN) represents the total number of attributes where the attribute of $A$ is 1 and the attribute of $B$ is 0 and true negative (TN) represents the total number of attributes where the attributes of $A$ and $B$ are 0.

Each attribute must fall into one of these four categories:

$$TP + FP + FN + TN = n$$  \hspace{1cm} (20)

Tanimoto similarity coefficient is calculated as the size of the intersection divided by the size of union of the set $A$ and $B$ as follows:

$$T = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} = \frac{TP}{FN + FP + TP}$$  \hspace{1cm} (21)

Dice similarity coefficient that is derived from a reliability criterion, known as Kappa statistics, is calculated as 2 times the size of union divided by sum of the sizes of the sets (Duda et al., 2000; Berkhin, 2006):

$$D = \frac{2|A \cap B|}{|A| + |B|} = \frac{2 \times TP}{(2 \times TP) + FP + FN}$$  \hspace{1cm} (22)

3. Results and discussion

Images that are used for the training and testing processes are obtained from the IBSR database. This database contains T1-weighted MR images of 20 normal people and the segmentations of these images as gray matter, white matter and other, which is performed by experts. Each MR image shows a brain slice of 3.1 mm thickness and the size of the voxels are $1.17 \times 1.17 \times 3.1$ mm$^3$. The size of the images is 256 $\times$ 256 pixels. IBSR data sets have various difficulty levels and they include low contrast scans. Images are mostly inhomogeneous. Table 1 shows the IBSR data sets used in this study for training and test processes and the number of slices that the data sets include.

Images from 10 people are used for testing, while images from the other 10 people are used for testing the system. IBSR images contain 60 slices in average as can be seen in Table 1. Therefore the middle slice, which is the 30th one, is selected among all the slices that belong to a person for constructing the training and test data. The database contains the skull stripped forms of the images that include only intracranial brain structures and the manual segmentations of them. In segmentation process, these skull stripped and manually segmented images are used. An example of the training and test images are shown in Figs. 2 and 3, respectively.
Anisotropic diffusion filter, which smoothes the inner parts of the regions and emphasizes the edges, is applied to the training and test images before the segmentation process. Eq. (3) is used to determine the diffusion strength of the filter. Filtering parameters $\kappa$ and the number of iterations are taken as 5 and 10, respectively. Fig. 4 shows the results of the anisotropic diffusion filtering of the images in Fig. 2(b) and 3(b).

SWT is used to extract the features that distinguish different tissue types of the brain from the brain MR images. Unlike other wavelet transform methods, subsampling process is not applied during SWT. This means that the subimages obtained as a result of the transform are the same size as the original image. Thus the descriptive statistical features of the pixels can be extracted from the coefficients obtained from the transform using the spatial filtering method.

### Table 1

<table>
<thead>
<tr>
<th>IBSR data sets used for training</th>
<th>Number of slices</th>
<th>IBSR data sets used for test</th>
<th>Number of slices</th>
</tr>
</thead>
<tbody>
<tr>
<td>1_24</td>
<td>65</td>
<td>2_4</td>
<td>65</td>
</tr>
<tr>
<td>4_8</td>
<td>61</td>
<td>5_8</td>
<td>60</td>
</tr>
<tr>
<td>6_10</td>
<td>63</td>
<td>7_8</td>
<td>60</td>
</tr>
<tr>
<td>8_4</td>
<td>63</td>
<td>11_3</td>
<td>63</td>
</tr>
<tr>
<td>12_3</td>
<td>63</td>
<td>13_3</td>
<td>63</td>
</tr>
<tr>
<td>15_3</td>
<td>60</td>
<td>16_3</td>
<td>60</td>
</tr>
<tr>
<td>17_3</td>
<td>63</td>
<td>100_23</td>
<td>63</td>
</tr>
<tr>
<td>110_3</td>
<td>64</td>
<td>111_2</td>
<td>63</td>
</tr>
<tr>
<td>112_2</td>
<td>63</td>
<td>191_3</td>
<td>63</td>
</tr>
<tr>
<td>202_3</td>
<td>63</td>
<td>205_3</td>
<td>63</td>
</tr>
</tbody>
</table>

---

**Fig. 2.** The 30th slice of the 1_24 training dataset: (a) original, (b) skull stripped and (c) manually segmented.

**Fig. 3.** The 30th slice of the 2_4 test dataset: (a) original, (b) skull stripped and (c) manually segmented.

**Fig. 4.** Results of the anisotropic diffusion filtering of the images: (a) Fig. 2(b) and (b) Fig. 3(b).
One level SWT is applied to the images using the Daubechies 2 wavelet from the Daubechies wavelet family. As a result of the transform, one approximate and 3 detailed (horizontal, vertical and diagonal) subimages are obtained. Mean of the horizontal, vertical and diagonal subimages is calculated and one image indicating the details of the image is obtained. Eventually one approximate and one detailed image are obtained using the SWT.

Using only the raw coefficients obtained from the SWT is not enough for segmenting the images. In addition to these coefficients, features that represent the local statistical information of the tissue regions are extracted from the images by sliding a $3 \times 3$ sized window along the image. These features are the mean absolute deviation, entropy, energy and standard deviation calculated as in Eqs. (6)–(9).

Feature vector is constructed by combining the coefficients obtained from the SWT and the features calculated from these coefficients.

Names of the tissues are used as labels in the constructed feature vector for each of the images in the training and test data sets. Label information is obtained from the IBSR database, because it supplies the manually segmented images as well as the original ones. Feature vector is normalized in the [0 1] range before using as input to the SOM network.

SOM network is trained in an unsupervised manner (not taking into consideration the label information) using the parameters indicated in Table 2. Here $n$ is the number of input data.

The next process to be performed is the calibration of the map to place different input patterns on the map. For this purpose manually analyzed data sets are used as input and labeling is performed by determining the best matching neuron for the input. Number of hits for a specific tissue type that each neuron gets is determined in the labeling stage. The label of a neuron is assigned as the tissue type which the neuron gets maximum hit. If a network unit gets no hit or gets the same number of hits for the different tissue types then labeling is done by considering the neighboring units.

SOM and LVQ methods are combined in many practical applications. After SOM network is constructed, trained and labeled, supervised learning is performed using input data to ensure the best settlement of the neurons for the problem. In this instance, fine-tuning is applied to the weight vectors of the different classes or clusters using LVQ. Supervised training process is started with LVQ1 and continued with LVQ3 to increase the recognition accuracy of the system. Running length and $\alpha(t)$ learning rate of the LVQ algorithms are chosen 1000 and 0.5, respectively. For LVQ3 algorithm window width parameter was 0.2 and relative learning parameter $e$ was 0.3.

Performance of the system is tested using test images. Achieved results are evaluated using overlap metrics also known as Tanimoto and Dice similarity indexes. The value of these metrics is 1.0 if the results are completely overlapped and 0.0 if the results do not share any pixels classified as same.

Performance of the system, trained using one image selected from each of the 10 test data sets, is evaluated using one image selected from each of the 10 test data sets. The results of the system are compared with manually segmented images for the gray matter and white matter and evaluated using Tanimoto and Dice similarity indexes. Results are given in Tables 3 and 4.

Segmentation performances of the different algorithms that use datasets from the IBSR database announced at the official site of the IBSR are given in Table 5. Numbers in the table indicate the Tanimoto similarity indexes that show the performance of the methods. They are the results for all 20 brain scans.

Results of the methods in Table 5 and our method for the used test data sets are given in Table 6. Comparing the results of our system with the results of these methods using Tanimoto similarity index reveals that our system shows better segmentation performance for the gray matter while it gives average results for white matter. Comparison results of the methods for the test data sets for gray matter are given in Fig. 5 as a graph.

Average segmentation performance of the different methods for the test data sets is given as a graph in Fig. 6.

While comparing the results it should be noted that all the methods including ours use different types of filters to remove the

### Table 2

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training algorithm</td>
<td>Sequential</td>
</tr>
<tr>
<td>Number of neurons</td>
<td>$5 \times 10^{5.54321}$</td>
</tr>
<tr>
<td>Initialization</td>
<td>Random</td>
</tr>
<tr>
<td>Lattice</td>
<td>Hexagonal</td>
</tr>
<tr>
<td>Learning rate [rough fine-tuning]</td>
<td>[0.5 - 0.05]</td>
</tr>
<tr>
<td>Neighborhood function</td>
<td>Gauss</td>
</tr>
<tr>
<td>Learning rate function</td>
<td>Exponential</td>
</tr>
</tbody>
</table>

### Table 3

Tanimoto similarity index values obtained from the system.

<table>
<thead>
<tr>
<th>Data set</th>
<th>White matter</th>
<th>Gray matter</th>
</tr>
</thead>
<tbody>
<tr>
<td>2_4</td>
<td>0.2989</td>
<td>0.4534</td>
</tr>
<tr>
<td>5_8</td>
<td>0.5639</td>
<td>0.6911</td>
</tr>
<tr>
<td>7_8</td>
<td>0.3538</td>
<td>0.5356</td>
</tr>
<tr>
<td>11_3</td>
<td>0.6295</td>
<td>0.6979</td>
</tr>
<tr>
<td>13_3</td>
<td>0.6035</td>
<td>0.7119</td>
</tr>
<tr>
<td>16_3</td>
<td>0.5544</td>
<td>0.6905</td>
</tr>
<tr>
<td>100_23</td>
<td>0.5863</td>
<td>0.7210</td>
</tr>
<tr>
<td>111_2</td>
<td>0.6544</td>
<td>0.6775</td>
</tr>
<tr>
<td>191_3</td>
<td>0.6031</td>
<td>0.6633</td>
</tr>
<tr>
<td>205_3</td>
<td>0.6334</td>
<td>0.7054</td>
</tr>
<tr>
<td>Average</td>
<td>0.5481</td>
<td>0.6548</td>
</tr>
</tbody>
</table>

### Table 4

Dice similarity index values obtained from the system.

<table>
<thead>
<tr>
<th>Data set</th>
<th>White matter</th>
<th>Gray matter</th>
</tr>
</thead>
<tbody>
<tr>
<td>2_4</td>
<td>0.4602</td>
<td>0.6239</td>
</tr>
<tr>
<td>5_8</td>
<td>0.7211</td>
<td>0.8173</td>
</tr>
<tr>
<td>7_8</td>
<td>0.5227</td>
<td>0.6976</td>
</tr>
<tr>
<td>11_3</td>
<td>0.7276</td>
<td>0.8221</td>
</tr>
<tr>
<td>13_3</td>
<td>0.7527</td>
<td>0.8317</td>
</tr>
<tr>
<td>16_3</td>
<td>0.7133</td>
<td>0.8169</td>
</tr>
<tr>
<td>100_23</td>
<td>0.7394</td>
<td>0.8379</td>
</tr>
<tr>
<td>111_2</td>
<td>0.7911</td>
<td>0.8078</td>
</tr>
<tr>
<td>191_3</td>
<td>0.7524</td>
<td>0.7976</td>
</tr>
<tr>
<td>205_3</td>
<td>0.7756</td>
<td>0.8273</td>
</tr>
<tr>
<td>Average</td>
<td>0.7001</td>
<td>0.7880</td>
</tr>
</tbody>
</table>

### Table 5

Average overlaps between manually guided segmentations and various methods for all 20 brain scans.

<table>
<thead>
<tr>
<th>White matter</th>
<th>Gray matter</th>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.567</td>
<td>0.564</td>
<td>adaptive MAP</td>
</tr>
<tr>
<td>0.562</td>
<td>0.558</td>
<td>biased MAP</td>
</tr>
<tr>
<td>0.567</td>
<td>0.473</td>
<td>fuzzy c-means</td>
</tr>
<tr>
<td>0.554</td>
<td>0.550</td>
<td>Maximum aposteriori probability (MAP)</td>
</tr>
<tr>
<td>0.551</td>
<td>0.535</td>
<td>Maximum-likelihood</td>
</tr>
<tr>
<td>0.571</td>
<td>0.477</td>
<td>tree-structure k-means</td>
</tr>
<tr>
<td>0.832</td>
<td>0.876</td>
<td>Manual (4 brains averaged over 2 experts)</td>
</tr>
</tbody>
</table>
bias field from the images before segmentation process. In our method we dealt with the bias field using an anisotropic diffusion filtering as a preprocessing step. The other methods use a low-pass moving average filter with a threshold to neglect spurious bias values during the averaging process before segmentation. Each method uses a preprocessing step which is considered to be the most appropriate one for that particular method. Comparing the effects of preprocessing steps on the results of segmentation methods is beyond the scope of this study. In performance comparisons, each method is evaluated together with its particular filters and preprocessing steps.

Figs. 7 and 8 show the test images, manual segmentations and the result images obtained from the system for 111_2 and 205_3 data sets, respectively.

4. Conclusion

This paper proposes an automatic brain MR image segmentation method. The proposed method contains preprocessing, feature extraction, segmentation and evaluation stages. Images obtained from the IBSR database are used for the training and

<table>
<thead>
<tr>
<th>Data set</th>
<th>MAP</th>
<th>Adaptive MAP</th>
<th>Biased MAP</th>
<th>Fuzzy c-means</th>
<th>Maximum-likelihood</th>
<th>k-means</th>
<th>Our method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gray</td>
<td>White</td>
<td>Gray</td>
<td>White</td>
<td>Gray</td>
<td>White</td>
<td>Gray</td>
</tr>
<tr>
<td>2,4</td>
<td>0.31</td>
<td>0.19</td>
<td>0.36</td>
<td>0.28</td>
<td>0.30</td>
<td>0.18</td>
<td>0.31</td>
</tr>
<tr>
<td>5,8</td>
<td>0.15</td>
<td>0.07</td>
<td>0.27</td>
<td>0.13</td>
<td>0.15</td>
<td>0.08</td>
<td>0.20</td>
</tr>
<tr>
<td>7,8</td>
<td>0.60</td>
<td>0.64</td>
<td>0.62</td>
<td>0.64</td>
<td>0.61</td>
<td>0.65</td>
<td>0.53</td>
</tr>
<tr>
<td>11,3</td>
<td>0.68</td>
<td>0.72</td>
<td>0.66</td>
<td>0.71</td>
<td>0.69</td>
<td>0.74</td>
<td>0.58</td>
</tr>
<tr>
<td>13,3</td>
<td>0.66</td>
<td>0.70</td>
<td>0.65</td>
<td>0.68</td>
<td>0.66</td>
<td>0.67</td>
<td>0.52</td>
</tr>
<tr>
<td>16,3</td>
<td>0.58</td>
<td>0.56</td>
<td>0.58</td>
<td>0.57</td>
<td>0.59</td>
<td>0.57</td>
<td>0.46</td>
</tr>
<tr>
<td>100,23</td>
<td>0.67</td>
<td>0.66</td>
<td>0.66</td>
<td>0.66</td>
<td>0.68</td>
<td>0.67</td>
<td>0.55</td>
</tr>
<tr>
<td>111,2</td>
<td>0.56</td>
<td>0.64</td>
<td>0.57</td>
<td>0.65</td>
<td>0.61</td>
<td>0.68</td>
<td>0.48</td>
</tr>
<tr>
<td>191,3</td>
<td>0.65</td>
<td>0.72</td>
<td>0.66</td>
<td>0.69</td>
<td>0.66</td>
<td>0.72</td>
<td>0.57</td>
</tr>
<tr>
<td>205,3</td>
<td>0.67</td>
<td>0.71</td>
<td>0.67</td>
<td>0.70</td>
<td>0.66</td>
<td>0.71</td>
<td>0.59</td>
</tr>
<tr>
<td>Average</td>
<td>0.55</td>
<td>0.56</td>
<td>0.57</td>
<td>0.57</td>
<td>0.56</td>
<td>0.57</td>
<td>0.48</td>
</tr>
</tbody>
</table>

**Fig. 5.** Comparison results of the methods for the test data sets for gray matter.

**Fig. 6.** Average segmentation performance of the different methods for the test data sets.
testing processes. An anisotropic filtering preprocess is performed before segmentation to improve the quality of the brain MR images. Then SWT is applied to the images to obtain subimages that contain multiresolution information for distinguishing different tissues. Statistical features are extracted from the subimages using spatial filtering process. A multidimensional feature vector is formed by combining SWT coefficients and their statistical features. This feature vector is used as input to the SOM. SOM is trained using unsupervised training methodology. Then LVQ1 and LVQ3 algorithms are utilized for fine-tuning the weights of the SOM. As a result, brain MR images are segmented into gray matter, white matter and background regions. The results of the system are compared with manually segmented images for the gray matter and white matter and evaluated using Tanimoto and Dice similarity indexes. According to the comparison results the proposed method is able to provide robust clinical image segmentation. Segmentation regions such as tumor, edema and lesion for diseased or injured people should be determined for the purpose of diagnosis and planning surgical operations. Further studies are ongoing for improving the segmentation accuracy and extending the segmentation to include diseased or injured brain regions.

Acknowledgements

This study has been supported by Gazi University Scientific and Research Project Fund (Project no.: 07/2009–04).

References


McConnell Brain Imaging Centre (BIC) of the Montreal Neurological Institute, McGill University, BrainWeb: Simulated Brain Database, accessed from (http://www.bic.mni.mcgill.ca/brainweb/), 2010.

The Internet Brain Segmentation Repository (IBSR), Center for Morphometric Analysis Neuroscience Center, Harvard Medical School, accessed from (http://www.cma.mgh.harvard.edu/ibsr/), 2010.

The Self-organizing Maps, third ed. Springer, Germany.


