2004 Special Issue

Fully automated biomedical image segmentation by self-organized model adaptation

Axel Wismüller\textsuperscript{a,b,*}, Frank Vietze\textsuperscript{a}, Johannes Behrend\textsuperscript{a}, Anke Meyer-Baese\textsuperscript{b}, Maximilian Reiser\textsuperscript{a}, Helge Ritter\textsuperscript{c}

\textsuperscript{a}Institut für Klinische Radiologie, Ludwig-Maximilians-Universität München, Klinikum Innenstadt, Ziemessensstrasse 1, München 80336, Germany
\textsuperscript{b}Florida State University, Tallahassee, FL, USA
\textsuperscript{c}AG Neuroinformatik, Universität Bielefeld, Bielefeld, Germany

Revised 28 June 2004; accepted 28 June 2004

Abstract

In this paper, we present a fully automated image segmentation method based on an algorithm that provides adaptive plasticity in function approximation problems: the deformable (feature) map (DM) algorithm. The DM approach reduces a class of similar function approximation problems to the explicit supervised one-shot training of a single data set. This is followed by a subsequent, appropriate similarity transformation, which is based on a self-organized deformation of the underlying multidimensional probability distributions. We apply this algorithm to the real-world problem of fully automated voxel-based multispectral image segmentation, employing magnetic resonance data sets of the human brain. In contrast to previous segmentation approaches, the knowledge obtained within the segmentation procedure of a single prototypical reference data set can be re-utilized for the segmentation of new, ‘similar’ data employing a strategy of incremental adaptive learning based on the DM algorithm. Thus, we obtain a fully automatic segmentation method that does neither require manual contour tracing of training regions, visual classification of voxel clusters, nor any other kind of human intervention. Our application demonstrates that flexible learning by a strategy of self-organized incremental model adaptation can contribute to increase the efficiency and practicability of biomedical image processing systems.

© 2004 Elsevier Ltd. All rights reserved.

Keywords: Feature maps; Self-Organization; Neural networks; Image segmentation; Pattern analysis; Knowledge Discovery; Magnetic resonance imaging

1. Introduction

Drift phenomena of underlying data distributions are a frequent challenge to data analysis by machine learning techniques. In order to efficiently extract knowledge from data in the presence of drift phenomena, the question arises how such changes can be taken into account in an adaptive manner, i.e. without a complete re-execution of training algorithms on modified or additional data that have not yet been incorporated in previous learning procedures.

The need to incorporate adaptive learning strategies may originate from different conditions induced by real-world learning problems. For instance, the target of a function approximation problem may represent a dynamical system in a changing environment involving an inevitable temporal shift of parameters. A different situation may be represented by additional data that share some kind of apparent ‘similarity’ with previously utilized training data in the solution of pattern analysis problems by comparing different, but somehow ‘similar’ objects. In biomedical research data sets, this phenomenon can be observed frequently (see e.g. Wismüller & Dersch, 1997). One may think of the inter-individual variability of anatomical features: there are no completely identical biological individuals, but there may be obvious anatomical ‘resemblances’.

In order to cope with such problems, we have proposed an algorithm that provides adaptive plasticity in function approximation problems.
approximation problems: the Deformable feature Map (DM) algorithm as described in Wismüller et al. (2000) and Wismüller, Vietze, Dersch, and Hahn (1998). The DM approach reduces a class of similar function approximation problems to the explicit supervised one-shot training of a single data set. This is followed by a subsequent, appropriate similarity transformation, which is based on a self-organized incremental deformation of the underlying multidimensional probability distributions. A first successful application of the DM algorithm to the problem of image registration has been published previously (Wismüller et al., 2002) (Fig. 1).

In this paper, we present a new real-world application of the deformable feature map to a problem involving adaptive learning based on the inter-individual variability of biomedical data as sketched above. We describe a complete system for automatic multispectral image segmentation, employing magnetic resonance data sets of the human brain.

Segmentation can be defined as the identification of ‘meaningful’ image components. It is a fundamental task in image processing providing the basis for any kind of further high-level image analysis. In medical image processing, a wide range of clinical applications is based on segmentation. For a general overview on medical image segmentation techniques, we refer to the review Rogowska (2000). Interactive segmentation by manual contour tracing, however, requires a considerable amount of human intervention. The expense of human expert resources required for this task is intractable for larger data sets. Therefore, it is desirable to transfer the segmentation task to machines.

A successful approach to this problem is based on multispectral image segmentation. Here, the image object is examined by \( n > 1 \) different image acquisition procedures, e.g. different MRI sequence protocols, or multiple image acquisitions in time. Appropriate preprocessing steps comprise the anatomically correct registration of the data sets and masking a region of interest in which the segmentation should be performed. Finally, each voxel can be characterized as a vector in a \( n \)-dimensional feature space composed of the intensity levels obtained for the different image acquisitions. Segmentation then becomes the problem of classifying these multidimensional feature vectors as belonging to a specific element among a given set of alternative ‘meaningful’ image components.

Continuous efforts based on a systematic exploration of this concept have led to several successful real-world applications in biomedical image processing as presented in previous publications by our group. For the segmentation approach introduced in this paper, some relevant conceptual foundations can be found in previous work on both structural MRI of the human brain (Wismüller et al., 2000) and functional segment in biomedical image time-series, such as functional MRI data analysis for human brain mapping, dynamic contrast-enhanced perfusion MRI for the diagnosis of cerebrovascular disease, or MRI mammography for the analysis of suspicious lesions in patients with breast cancer (Wismüller et al., 2002).

The segmentation algorithms described in these publications were not based on presumptive heuristic rules derived from anatomical meta-knowledge of how a classification decision should be made. In contrast, purely data-driven self-organization of the classifier was employed according to the principle of ‘learning by example’ rather than analyzing data w.r.t a fixed set of given rules. The same holds for the approach presented here. However, in contrast to previous work, the knowledge obtained within the segmentation procedure of a single prototypical reference data set can be re-utilized for the segmentation of new, ‘similar’ data employing a strategy of incremental adaptive learning based on the DM algorithm. Thus, we obtain a fully automatic method for segmentation of multispectral MRI data sets of the human brain that does neither require manual contour tracing of training regions, visual classification of voxel clusters, nor any other kind of human intervention. Our application demonstrates that data analysis by a strategy of self-organized incremental model adaptation can contribute to increase the efficacy and practicability of biomedical image segmentation systems.

Fig. 1. Application of the DM algorithm to a toy example. The two distributions are similar, but not identical. Here, they differ with respect to size and rotation. Note the gradual deformation of the target distribution onto the source distribution with increasing number of iterations. The lines represent a triangulation of the original target distribution. The absence of line crossings during the procedure can serve as an indicator for topology preservation without ‘twisting’.

A. Wismüller et al. / Neural Networks 17 (2004) 1327–1344
i.e. a domain of considerable practical relevance to real-world data processing.

In the remainder of this article, we (i) explain the theory of the DM algorithm as an instrument for self-organized incremental adaptive learning; (ii) discuss its application to medical image segmentation; and (iii) present our segmentation results for multispectral 3D MRI data sets of the human brain with respect to the tissue classes ‘gray matter’, ‘white matter’, and ‘cerebrospinal fluid’, including a quantitative comparative evaluation referring to the performance of a semi-automatic segmentation system described in the literature Wismüller et al. (2000).

2. Theory

2.1. Unsupervised learning

In the following, the DM algorithm is explained in analogy to previous publications. We restrict to aspects that are essential for understanding of the image segmentation application in Section 3. The interested reader may consult (Wismüller et al., 2002) for further details on theoretical considerations, such as convergence properties, various solutions for interpolation and approximation between reference points, and important conceptual differences w.r.t. self-organizing maps.

The basic variant of the DM algorithm will be introduced in this section. It refers to incremental unsupervised learning based on topology-preserving adaptive vector quantization. The extension to supervised learning will be explained in Section 2.2.

According to Fig. 2, given are two similar, but not identical data distributions in the $n$-dimensional feature spaces $X$ and $Y$. Here, the total number of data vectors may differ between $X$ and $Y$, i.e. ‘similarity’ refers to probability densities. Let $x^\mu \in X (\mu \in \{1, \ldots, q\})$ denote the data points of the so-called source space $X \subset \mathbb{R}^n$, and $y^\nu \in Y (\nu \in \{1, \ldots, p\})$ the data points of the so-called target space $Y \subset \mathbb{R}^n$. Given this situation, the objective of the DM algorithm can be addressed as follows: how can $X$ and $Y$ be matched onto each other in a somewhat optimal manner, including local nonlinear deformations. In other words, how can we define a mapping $S : X \rightarrow Y$ that satisfies the following constraints:

(i) optimal correspondence of probability densities $f$ and $f'$ before and after the match, i.e. minimization of $\int_X \| f'(S(x)) - f(x) \| d^n x$, where $\| \cdot \|$ denotes an appropriate norm in $\mathbb{R}^n$, e.g. the Euclidean norm. Here, it should be noted that other distance measures between the probability densities may be employed as well, such as the Kullback–Leibler divergence (Kullback & Leibler, 1951);

(ii) minimization of the total deformation $\int_X \| (S(x)) - x \| d^n x$.

In the following, we present an algorithm that may provide suboptimal solutions of this minimization problem.

The target distribution in $Y$ can be represented by a set $C_Y$ of prototypical codebook vectors $r_j$, i.e. $C_Y = \{ r_j \in \mathbb{R}^n | j \in \{1, \ldots, N\}\}$ as a result of a suitable VQ procedure, e.g. Kohonen’s self-organizing map (SOM) algorithm (Kohonen, 1989) or minimal free energy VQ (Dersch, 1995; Rose, Gurewitz, & Fox, 1992), etc.

The basic idea of the DM algorithm is the self-organized adjustment of the original codebook vector positions $r_j \in C_Y$ of the target space $Y$ by re-training the codebook vectors with the data points of the source space $X$. This procedure results in a new corresponding codebook $C_X = \{ w_j \in \mathbb{R}^n | j \in \{1, \ldots, N\}\}$ representing the source distribution in $X$. In detail, for each codebook vector position $r_j$, we construct an associated weight vector $\tilde{w}_j \in X$ using a preliminary coordinate transformation $\tilde{w}_j = S_0^{-1}(r_j)$. In general, if $X$ and $Y$ can be considered as ‘similar’, choosing $S_0$ as the identical mapping can serve as an appropriate initial guess.

Subsequently, the codebook vector positions $w_j$ are adapted in an iterative procedure. After randomly choosing a data vector $x \in X$, the codebook vectors $w_j$ are updated according to

$$w_j(t+1) = w_j(t) + \varepsilon(t) h_j(x(t), \sigma(t))(x(t) - w_j(t)),$$

starting from $w_j(t = 0) = \tilde{w}_j$, employing the cooperativity function

$$h_j(x(t), \sigma(t)) = \exp\left(-\frac{(r_j - r_{\max}(x(t)))^2}{2\sigma^2(t)}\right),$$

and a suitable (e.g. exponential) annealing scheme of the learning parameter $\varepsilon(t)$ and the cooperativity length $\sigma(t)$ for every training step $t$. The codebook vector $r_{\max}(x(t))$ represents the ‘winner neuron’ with respect to the minimal distance to the presented data vector $x(t)$ in the feature space $X$. It should be emphasized that the cooperativity function $h_j(x(t), \sigma(t))$ is based on the metric of the target space, $Y$,
whereas the update of the codebook vectors according to (1) occurs in the source space \( X \). The positions of the \( r_j \) in the target space \( Y \) form the so-called ‘model cortex’ and remain unchanged. After the \( w_j \in X \) have converged, their values represent a new mapping \( w_j = S^{-1}(r_j) \).

Thus, the iterative training according to the update rule (1) results in a set of pairs \((w_j, r_j)\) of corresponding vectors, representing reference points for the definition of a mapping \( S: X \rightarrow Y, \ x \mapsto y \). Between these reference points, \( S \) has to be determined by some appropriate interpolation or approximation procedure. For this purpose, various alternative solutions have been proposed in Wismüller et al. (2002).

In order to take mapping properties of different VQ algorithms into account, i.e. to adapt the initial VQ codebook of the target space to the distortion behavior of the DM algorithm, we have proposed a heuristic computational scheme:

(i) The data vectors \( y \) of the target space \( Y \) are presented as the input to a vector quantizer as before. The resulting set of codebook vectors \( r_j^y \) with \( C_y^y = \{ r_j^y \in \mathbb{R}^n | j \in \{1, \ldots, N\} \} \) representing the data distribution in \( Y \) forms the model cortex for a single run of the DM algorithm on the target distribution (step (ii)) and for a single run on the source distribution (step (iii)).

(ii) To adapt the codebook \( C_y^y \) to the properties of the DM algorithm, a single run of the DM algorithm is performed on the data distribution in the target space \( Y \) as described above using the learning rule (1) and the cooperativity function (2). For this purpose, \( C_y^y \) is used for defining both the initial codebook and the model cortex. This results in a codebook \( C_y = \{ r_j \in \mathbb{R}^n | j \in \{1, \ldots, N\} \} \).

(iii) In a last step, adaptive vector quantization is performed by a single run of the DM algorithm on the data distribution in the source space \( X \). The codebook vectors \( w_j \) are initialized using the codebook vectors \( r_j \in C_y \), i.e. \( w_j = S^{-1}(r_j) \), where \( C_y \) is used as the model cortex like in step (ii). The DM algorithm causes the codebook vector positions to adapt slightly to the data distribution of the source space, and a corresponding codebook \( C_x = \{ w_j \in \mathbb{R}^n | j \in \{1, \ldots, N\} \} \) is obtained.

The results for fully automatic segmentation of the brain imaging data as described in Section 4 are obtained by the heuristics (i)–(iii). In analogy to the original version of the DM algorithm sketched above, we obtain a set of pairs \((w_j, r_j)\) of corresponding vectors which can be used as reference points for the definition of a mapping \( S: X \rightarrow Y, \ x \mapsto y \), followed by an appropriate interpolation or approximation procedure.

For the DM approach, there should be no points in \( X \) that do not correspond to specific points in \( Y \) and vice versa. For the segmentation method presented in this paper, this means that there should be no severe systematic differences between reference and test data, such as those involved by major pathological changes that distort the gray level probability distributions of the test data in comparison to the reference data. In such situations, manual re-editing of the datasets may be required, see Section 5.

For real-world applications, it has been shown to be advantageous to integrate the DM algorithm into the framework of a GRBF neural network in order to perform adaptive supervised learning, as pointed out in Section 2.2. In this context, an explicit determination of the mapping \( S \) can be avoided. Vice versa, it is the extension to supervised learning that provides a method for approximating the set of reference points.

2.2. Adaptive supervised learning: DM algorithm and GRBF network

A classical problem of machine learning is function approximation. Various algorithms have been proposed to solve this problem. In particular, nonlinear function approximation by neural networks has made valuable contributions to this field leading to innumerable successful applications in real-world domains throughout the sciences and engineering. Important examples are multilayer perceptrons trained by the error-back-propagation algorithm (Rumelhart, 1986) or (generalized) radial-basis-functions networks (GRBF networks, see e.g. Dersch, 1995; Girosi & Poggio, 1990; Moody & Darken, 1989). These algorithms are based on the supervised training of a sample data set by adapting the neural network parameters in order to represent an appropriate model of the target function. The GRBF network approach decouples the function approximation problem into two different computational steps: an initial unsupervised vector quantization (VQ) step is followed by a supervised training of the output weights. The architecture of such a network is depicted in Fig. 3.

As pointed out in the introduction, drift phenomena may change the target function over time, or new data may be incorporated into previously conducted learning scenarios.

![Fig. 3. Architecture of a three-layer (generalized) radial-basis-functions (RBF) network.](image-url)
Both situations can impair the performance of neural network function approximators considerably.

In this situation, adaptive plasticity of neural network weights guided by incremental learning can be a successful strategy in order to avoid a complete re-training of the function approximation network. Within the framework of GRBF function approximation, it is usually the supervised training of the output weights which is kept flexible in order to meet the needs of learning a changing target function, whereas the parameters obtained in the initial VQ procedure are preserved. For example, this approach is frequently chosen in the so-called mixture-of-experts solution of time-series prediction by competing RBF networks (see e.g. Kohlmorgen, 1995). This is motivated by the observation that the VQ step is computationally more expensive than the adaptive training of the output weights. However, there may be situations where repetitive supervised training is restricted to a single data set, followed by a series of appropriate subsequent computational steps. From a theoretical point of view, this approach reduces the function approximation problem for \( F' \). Given an arbitrary point \( x \in X \), this can be performed by the following computational steps:

1. Calculate \( S(x) \in Y \) as described above.
2. Calculate the activations \( a_j \) of the codebook vectors \( \mathbf{r}_j \) using the metric of \( Y \):
   \[
   a_j(S(x)) = \exp\left(-\frac{\|S(x) - \mathbf{r}_j\|^2}{2\sigma^2}\right).
   \]
3. Calculate the output activations of the GRBF network using the output weights \( s_{ij} \) which have been determined by supervised learning of a single data set in \( Y \):
   \[
   F_i(S(x)) = \sum_{j=1}^{N} s_{ij} a_j(S(x)).
   \]

As an alternative to the steps 1 and 2, the activities \( a_j \) can approximately be calculated using the metric of \( X \):

\[
\bar{a}_j(x) = \exp\left(-\frac{\|x - \mathbf{w}_j\|^2}{2\sigma^2}\right).
\]

Then, Eq. (6) is to be replaced by

\[
F_i(x) = \sum_{j=1}^{N} s_{ij} \bar{a}_j(x).
\]

This alternative approach provides the advantage that the interpolation between the reference points \( (\mathbf{w}_j, \mathbf{r}_j) \) can be omitted, as the mapping \( S \) is not explicitly required. However, the performance of the function approximator may decrease, as the original training of the output weights \( s_j \), was based on the metric of the target space \( Y \).

This approach is adopted for fully automatic segmentation of multispectral MRI data sets of the human brain as described later in Section 3.3.

3. Application to automated image segmentation

The general concept of multispectral voxel-based brain segmentation can be explained as follows: \( n \) different 3D data sets for each brain are obtained employing different MRI acquisition parameters. For the segmentation procedure described in this paper, we used \( n = 4 \) MRI acquisition sequences (T1 weighted, T2 weighted, proton density weighted, and inversion recovery sequences, see Section 3.1). Segmentation aims at classifying each voxel of the multispectral data set as belonging to a specific tissue type, thus obtaining information about structure and volume of the tissue classes. For various concepts and applications of multispectral medical image segmentation, we refer to the literature, e.g. Clarke et al. (1995), Cline, Lorensen, Kikinis, and Jolesz (1990), Fletcher, Marsotti, Hornak...
A classical problem with numerous clinical applications is the segmentation of brain imaging data with respect to the tissue classes gray matter, white matter, and cerebrospinal fluid (CSF). Several other structures such as meninges or venous blood may be introduced as additional segmentation classes. However, these additional classes comprise only a small part of the total brain volume. Furthermore, for most of the clinical applications, the focus of interest is reduced to gray and white matter structures. Therefore, we assigned these minor additional classes to CSF.

Although such a threefold classification of brain tissue may be sufficient for numerous clinical applications, it should be emphasized that the concept presented in this paper can be extended to an arbitrary number of tissue classes. In particular, one may think of introducing additional classes for the identification of pathological tissue, e.g., multiple sclerosis plaques or malignant brain tumor structures.

In order to adopt the supervised variant of the DM algorithm as described in Section 2.2 for fully automatic image segmentation of multispectral image data, the output weights of the GRBF network have to be trained in a supervised manner. For this purpose, a training data set has to be obtained manually based on interactive contour tracing performed by a human expert. The training data can then be utilized to calculate the output weights of the GRBF network in a supervised training procedure using a gradient descent method. Details about this so-called semi-automated segmentation procedure have been published previously (Wismüller et al., 2000).

The essential conceptual progress of the segmentation approach presented in this paper can be characterized as follows: In contrast to previous work on semi-automated segmentation, we do not have to generate individual training data for GRBF network segmentation of each data set separately; but the knowledge obtained from segmentation of a single reference data set can be re-utilized for new data. Thus, an incremental adaptive learning procedure can exploit information based on the similarity of data sets, as sketched in the introduction of this paper.

Guided by this idea, we propose a fully automated segmentation concept in Section 3.3. Human interaction is required only once for the segmentation of a single reference data set using the semi-automated method. Once this has been completed, a fully automated segmentation of brain MRI data sets from different individuals can be obtained without any further human intervention.

In the following, we describe the image data and comment on several preprocessing steps. For details, we refer to Wismüller et al. (2000).

### 3.1. Image data and preprocessing

The image data sets were obtained on a 1.5 T whole body MRI scanner (Siemens, Magnetom Vision). Ten healthy male volunteers (aged between 24 and 32 years) were examined employing a standardized MRI sequence protocol: T1 weighted MP-RAGE, T2 weighted and proton density (PD) weighted spin echo, and Inversion-Recovery (IR) sequences. For explanation of the respective MRI pulse sequence physics, we refer to the excellent collection (Riederer & Wood, 1997). The MRI acquisition parameters of each sequence are listed in Table A.1.

Fig. 4 presents an example of corresponding images of the four different MRI acquisition techniques. The example shows one of 63 coronal1 cross-sections orthogonal to the anterior commissure–posterior commissure (AC–PC)2 line. A respective pilot plan is depicted in Fig. 5.

Data preprocessing of the multispectral MRI data comprises (i) anatomically correct alignment of the data sets acquired within the different image acquisition procedures; (ii) exclusion of extra-cerebral3 structures by a mask, see Fig. 6; and (iii) rescaling of the gray level distribution of each data set by standard normalization. A detailed description of these preprocessing steps can be found in Wismüller et al. (2000).

### 3.2. Semi-automated segmentation by a GRBF network

After performing the preprocessing steps explained in Section 3.1, we obtain multispectral data \( \mathbf{G} \in \mathbb{R}^{(m,w,f,n)} \) of \( n \) correctly aligned, normalized data sets consisting of \( l \) images with size \( m_x \times m_y \) each (in this paper: \( n=4, l=63, m_x=m_y=256 \)), where extracerebral voxels have been excluded by a presegmentation mask. This can be interpreted as follows.

Each voxel of the multispectral 3D data set represents a \( n \)-dimensional feature vector \( \mathbf{x} \) that is determined by the tissue class of this specific voxel:

\[
\mathbf{x} = \begin{pmatrix}
\mathbf{g}_1 \\
\mathbf{g}_2 \\
\vdots \\
\mathbf{g}_n
\end{pmatrix} = \begin{pmatrix}
G_{r,1} \\
G_{r,2} \\
\vdots \\
G_{r,n}
\end{pmatrix},
\]

(9)

Now the data set \( X = \{ \mathbf{x} \} \) is presented as the input to an initial unsupervised clustering step. For this purpose, we use the minimal free energy VQ algorithm as described in Wismüller et al. (2002, 2000). Thus, a set \( \mathbf{C} \) of codebook vectors \( \mathbf{w}_j \) with \( \mathbf{C} = \{ \mathbf{w}_j \in \mathbb{R}^n | j \in [1, \ldots, N] \} \) is computed that represent the data set \( X \). Here, the number \( N \) of codebook vectors is much smaller than the number of feature vectors \( \mathbf{x} \). The resulting codebook vectors \( \mathbf{w}_j \) represent the parameters required for computing the activation of the hidden layer

---

1 i.e. Oriented in a plane orthogonal to a line that can be drawn from the face to the back of the skull.
2 These anatomical structures define a horizontal reference line for the definition of a coordinate system according to the atlas by Talairach and Tournoux Talairach and Tournoux (1988) used in neurosurgery.
3 Located outside the brain.
neurons in a GRBF neural network according to Fig. 3, see below.

Once the VQ procedure has been completed, the output weights $s_j$ (see Fig. 3) of the GRBF network have to be trained in a second, supervised learning step in order to use the GRBF network as a neural classifier. For this purpose, a global gradient descent training procedure explained in Wismüller et al. (2000) is employed. In this context, the training data set $T = \{(x^n, y^n) | n \in \{1, \ldots, p\}\}$ required for supervised learning is provided by interactive labeling of a small representative subset of the data by a human observer. This is performed by manual contour tracing of regions that can clearly be attributed to one of the tissue classes without ambiguity by the human expert. Fig. 7 shows an example for regions manually labeled as ‘gray matter’, ‘white matter’, and ‘CSF’. If a training vector $x^n$ can be assigned to the tissue class $\lambda$, the corresponding component $y^n_\lambda$ of the target vector $y^n$ is set to one, whereas all the other components are set to zero.

After completion of the supervised training, each feature vector $x \in X$ can be assigned to one of the three tissue classes ‘gray matter’, ‘white matter’, and ‘CSF’ in the working phase of the GRBF network. If a feature vector $x$ is presented to the input layer (see Fig. 3), the neurons of the hidden layer will be activated according to:

$$a_j(x) = \frac{\exp\left(-\frac{\|x - w_j\|^2}{2\rho_j^2}\right)}{\sum_{i=1}^{N} \exp\left(-\frac{\|x - w_i\|^2}{2\rho_i^2}\right)}.$$  

Fig. 5. Pilot plan for the acquisition of coronal cross-sections of the human brain orthogonal to the anterior commissure–posterior commissure (AC–PC) line.
Now the hidden layer activations $a_j(x)$ are propagated linearly to the $m$ cells of the output layer via the output weights $s_j$:

$$y_i(x) = \sum_{j=1}^{N} s_j a_j(x), \quad i \in \{1, \ldots, m\}.$$  \hspace{1cm} (11)

Here, each output neuron $i$ corresponds to one of the $m$ tissue classes $\lambda$. A voxel represented by its feature vector $x$ is assigned to the class $\lambda(x) = i$ with maximal activation $y_i(x)$ of the corresponding output neuron $i$:

$$\lambda(x) = i, \quad \text{with} \quad y_i = \max_j y_j.$$  \hspace{1cm} (12)

The assignment of all the voxels to a tissue class according to Eq. (12) finally yields the segmentation of the data set.

3.3. Segmentation by the DM algorithm

Although the semi-automated segmentation method as described in Section 3.2 reduces the time consumption by human experts considerably in contrast to manual interactive contour tracing, still a significant amount of human intervention is required in order to obtain a training data for each brain imaging data set individually.

As human intervention is expensive and, therefore, should be minimized, it is desirable to perform segmentation in a fully automated way, i.e. without interaction on the part of a human expert reader. Here, we introduce a method that enables fully automated segmentation of arbitrary data sets solely based on a given segmentation of a single so-called ‘reference’ data set. The key idea is to employ the supervised variant of the DM algorithm as explained in Section 2.2 in order to exploit intrinsic similarities between reference and test data. Thus, human intervention, i.e. manual segmentation is needed only once, namely for the generation of a single reference training data set.

Both the reference data set and any other data set to be segmented are preprocessed according to 3.1, i.e. $n$-dimensional feature vectors are obtained from $n$ anatomically aligned, normalized single MR sequence data sets containing brain voxels after exclusion of extracerebral structures.

The segmentation approach based on the DM algorithm exploits the pairwise resemblance of different brain MRI data sets with regard to their multidimensional data distribution in the gray level feature space. Figs. 8 and 9 show this by the example of two four-dimensional multispectral data sets representing the gray level distributions of multispectral brain MR data gained from two different individuals. The images on the left show projections of the four-dimensional feature vector distribution on each plane spanned by any combination of two out of four different axes representing the gray levels obtained from specific image acquisition methods, i.e. T1-, T2-weighted, proton density, and inversion recovery MR sequences. The images on the right represent gray level projections obtained from brain MR data of a different individual. The similarity of corresponding projections is depicted clearly.

![Fig. 6. Presegmentation by masking of extracerebral structures.](image)

![Fig. 7. Manual labeling of tissue classes for supervised learning. The labeled regions (medium gray 'gray matter', light gray 'white matter', dark gray 'CSF') provide the training data set for supervised learning of the output weights $s_j$ of the GRBF neural network.](image)
3.4. Segmentation of the reference data set

As a first step, a specific multispectral 3D brain MRI data set has to be chosen as the reference data set $Y$. The segmentation of this data set forms the basis for subsequent fully automated segmentation of arbitrary data sets gained from different individuals. Therefore, the segmentation of the reference data set should be as accurate as possible, as segmentation errors would lead to corresponding errors in the segmentation of the test data.

The feature vectors $y$ of the reference data set are presented to a minimal free energy vector quantizer according to Wismüller et al. (2000). The resulting codebook $C_Y = \{r_j^{\prime} \in \mathbb{R}^n | j \in \{1, \ldots, N\}\} \subseteq \mathbb{R}^n$ is used as an initialization (the so-called model cortex) for the subsequent application of the DM algorithm (see step (i) of the computational scheme explained at the end of Section 2.1). By iterating Eq. (1), a new codebook $C_Y = \{r_j^{\prime} \in \mathbb{R}^n | j \in \{1, \ldots, N\}\} \subseteq \mathbb{R}^n$ is obtained, which represents the data set $Y$ as well. This first run of the DM algorithm is required in order to fit the original codebook to the properties of the DM algorithm w.r.t. the distortion behavior of different VQ algorithms (see step (ii) of the computational scheme at the end of Section 2.1).

Based on the codebook $C_Y$, the output weights $s_j$ of a GRBF network (Fig. 3) are trained according to Section 3.2.
For this purpose, the training data obtained by manual contour tracing should be of excellent quality w.r.t size and segmentation accuracy in order to serve as the basis for the subsequent fully automatic segmentation of arbitrary test data. In this context, a significant investment of human expertise can be tolerated for obtaining the training data, as these resources have to be employed only once for the reference data set.

In the working phase of the GRBF network, each feature vector $\mathbf{y}$ is mapped onto one of the three tissue classes GM, WM and CSF using the codebook $C_Y$ and the output weights $s_j$. Thus, we obtain a segmentation of the reference data set.

For the fully automated segmentation of arbitrary test data, i.e. brain MR imaging data sets gained from different individuals, the codebooks $C'_Y$ and $C_Y$ and the output weights $s_j$ are required, as explained in the following section.

3.5. Fully automated segmentation of arbitrary data sets

Here, we describe the steps required for fully automated segmentation of an arbitrary multispectral 3D brain MR imaging data set $X$ based on the given segmentation of the reference data $Y$, which has been obtained in Section 3.4.

Fig. 9. Data distributions of two four-dimensional multispectral MR image data sets ($T_1$-weighted, $T_2$-weighted, proton density, and inversion recovery) from different brains. The figures on the left belong to the brain of a specific individual, those on the right show the respective distribution gained from the MR imaging data of the brain of a different individual. Each figure depicts the projection on the $T_1$–$T_2$-, $T_1$-proton density-, $T_1$-inversion recovery-, $T_2$-proton density, $T_2$-inversion recovery, and proton density-inversion recovery plane, respectively.
As described above, X is preprocessed as described in Section 3.1. For the application of the DM algorithm to the segmentation of the data set X, the codebook vectors \( w_j \) of X are initialized with the codebook vectors \( r_j \):

\[
\mathbf{w}_j = \mathbf{r}_j.
\]

The adaptive vector quantization is then performed by iterating equation (1), where \( \varepsilon(t) \) and \( \sigma(t) \) are annealed exponentially during the iteration process and \( C'_Y \) serves as the model tissue class. This procedure results in a codebook \( C_X = \{w_j \in \mathbb{R}^e | j \in [1, ..., N]\} \) aligned to data set \( X \).

In the subsequent working phase of the GRBF network, the feature vectors \( x \in X \) are classified w.r.t. the tissue classes GM, WM and CSF, as explained in Section 3.2. However, the decisive technical feature in this context is that we use these output weights which have been obtained by the GRBF network training of the reference data set \( Y \) together with the adaptively trained codebook \( C_X \). Therefore, a re-training of the output weights for the data set \( X \) is not required. Thus, the manual generation of a new training data set can be omitted.

It should be noted that all the steps can be executed in a fully automatic manner, i.e. without user interaction. An additional advantage of the described method is that we do not have to obtain a complete vector quantization for each data set. Instead, only an adaptive alignment of a given codebook to a new data set has to be performed, which significantly reduces the computational expense of the segmentation process.

Computations were performed on a Sun Ultrasparc workstation. The software code was written in C for the minimal free energy VQ and the training of the output weights. All other programs, such as the DM algorithm, the GRBF classification and data visualization routines, were written in Interactive Data Language (IDL™).

4. Results

After preprocessing according to Section 3.1, the data sets were segmented by applying two different strategies:

- Semi-automated tissue classification by a GRBF network according to Section 3.2;
- Fully automated tissue classification by the supervised variant of the DM algorithm using a GRBF network with preserved output weights according to Sections 2.2, 3.3 and 3.5.

The problem of evaluating the segmentation results is that the ‘true’ segmentation, i.e. the ‘gold standard’, is unknown. Even complete interactive segmentation by experienced human experts would be influenced by subjective bias leading to inter- and intra-individual variability of manual segmentation results. Furthermore, a manual segmentation of a complete brain data set is generally not available due to the demand for immense resources of human expert time and effort. To issue a vote on segmentation quality despite this lack of a gold standard, a semi-quantitative voting scheme was introduced. The segmentation quality of each slice of a 3D data set was carefully revised and evaluated by an experienced neuroradiologist on a voting scale ranging from 1 (= very good) to 5 (= poor). The votes were averaged over all slices and served as an evaluation criterion for segmentation quality.

We first discuss results for the semi-automated tissue classification by a GRBF network which was performed on all 10 data sets. The parameters of the GRBF classifier are listed in Appendix A.2.1. The training data comprise approx. 1% of the whole data set each. Detailed numbers are listed in Appendix A.2.2. The generation of a training data set based on manual interactive contour tracing by a human expert required 41 ± 11 min each. Computation time on a Sun Ultrasparc workstation was 2.2 ± 0.3 h, where the major part was used for computing the vector quantization. It should be noted that the computer programs were not optimized w.r.t. time consumption.

The voting results for the semi-automated GRBF segmentation quality of the 10 analyzed data sets are listed in Table A.7. Obviously, this method yields good results. Typical segmentation examples are shown in Figs. 10b–12b. A detailed list of the segmentation results for each data set is presented in Appendix A.2.3.

When evaluating these excellent results, one should take into consideration that these could only be obtained by (i) high computational expense and (ii) a significant amount of human expert resources required for generating an individual training data set.

Now the fully automated segmentation method described in Section 3.3 will be discussed. Here, human intervention is required only once for segmentation of a single reference data set. The segmentation of the other data sets can then be performed without any user interaction.

For this method, data set 1 was chosen as the reference data set. It was segmented according to Section 3.4. Fully automated segmentation of the remaining nine data sets w.r.t. the structure classes GM, WM, and CSF was performed according to Section 3.5. Computation time was 10 ± 2 min for each data set (as compared to 2.2 ± 0.3 h of computation time plus 41 ± 11 min for interactive contour tracing of the training data in the semi-automated method, see above). The parameters for the fully automated method are listed in Appendix A.3.1.

As can be concluded from the results of the quality evaluation in Table A.7, this method is slightly outperformed by the semi-automated technique with regard to human expert voting, but yields good results nevertheless. Segmentation examples are shown in Figs. 10c–12c. Detailed segmentation results can be found in Appendix A.3.2.

The results were compared with the results obtained by the method explained in Section 3.2 as a (suboptimal) ‘gold standard’ using contingency tables. The average inter-method agreement for all voxels was 94.7%.
Fig. 10. Comparison of the segmentation results for a frontal coronal cross-section of the brain (data set 3): (a) T1-weighted image; (b) semi-automated segmentation by a GRBF neural network; (c) fully automated segmentation using the DM algorithm; and (d) fully automated segmentation without using the DM algorithm.

Fig. 11. Comparison of the segmentation results for a central coronal cross-section of the brain (data set 3): (a) T1-weighted image; (b) semi-automated segmentation by a GRBF neural network; (c) fully automated segmentation using the DM algorithm; and (d) fully automated segmentation without using the DM algorithm.
It should be noted that the deviations are not only caused by voxels located within the brain, but also by voxels that can be attributed to meninges or other extracerebral structures which have not been removed properly in the presegmentation process. Furthermore, even the ‘gold standard’ may be subject to tissue class assignment errors based on the individual assessment of a specific human expert when performing manual contour tracing in order to obtain the training data set. Therefore, a complete agreement of segmentation results cannot be expected, anyway.

5. Discussion

In this paper, we have introduced a new segmentation method based on adaptive learning using the DM algorithm. The knowledge obtained within the segmentation procedure of a single prototypical reference data set can be re-utilized for the segmentation of new, ‘similar’ data employing a strategy of incremental self-organized model adaptation.

The essential advantage of the segmentation method is that it does not require any human intervention, but nonetheless leads to good results. Compared to the semi-automated GRBF tissue classification with manual generation of the training data set, it provides the additional advantage that it does not depend on the radiologist’s individual assessment, where imprecise labeling of the training data set may lead to impaired segmentation results. In this context, it should be noted that a training data set is generated in order to obtain a reference data set which serves as the basis for the fully automated segmentation of an arbitrary number of new data sets. Finally, a significant reduction of computational expense compared to the semi-automated method should be emphasized. This results from the fact that for each segmentation process only a moderate codebook adaptation has to be performed instead of a complete VQ in the semi-automated method.

It should be noted that the segmentation approach presented in this paper ignores spatial information that is used in most other methods, such as in human contour tracing and in various software packages, e.g. the rather widely used FreeSurfer™ software Dale, Fischl, and Sereno (1999). However, in contrast to these approaches, our method incorporates the full gray level information obtained by different MRI sequences simultaneously in the sense of multispectral image analysis, which may partly compensate the exclusion of spatial information in the segmentation.

Fig. 12. Comparison of the segmentation results for an occipital coronal cross-section of the brain (data set 3): (a) T1-weighted image; (b) semi-automated segmentation by a GRBF neural network; (c) fully automated segmentation using the DM algorithm; and (d) fully automated segmentation without using the DM algorithm.
process. Although other methods may be used with more than one MRI sequence as well, they frequently discard major parts of this information, e.g. by simply averaging the voxel-specific gray levels obtained in the different MRI sequences.

Although we have not specifically tested the level of similarity between different brains that can be tolerated by the DM algorithm, we conjecture that there will be limited performance when comparing brains with abnormal findings or apparent anatomical variants. In these cases, manual re-editing will be required, as for other brain segmentation systems as well. However, as the segmentation method described in this paper does not take into account morphological features, but only voxel-specific gray level spectra, smaller anatomical differences may be tolerated, as long as they do not disturb the frequency distribution of voxels representing the different tissue classes considerably.

In the case of pathological changes, one might introduce additional tissue classes, e.g. for white matter lesions. However, if the properties of the pathological lesions are not known in advance, one might think of extending the segmentation approach by introducing an additional ‘residual class’ that may be rejected by the classifier. However, although this is an interesting perspective for future research work, we conjecture that such a system might be subject to errors in difficult cases involving major pathology, nevertheless.

Therefore, we do not recommend the uncritical use of the segmentation system without visual control. Nevertheless, there is a wide range of medical problems for potential future applications of the segmentation system where major pathological differences between reference and test data are missing, such as the formation of neuroanatomical atlases, e.g. for inter-individual comparisons w.r.t. the localization of brain activity in functional MRI studies, or the determination of global brain atrophy measures, such as brain volume measurements in neurodegenerative or demyelinating diseases.

One may think of applying the method explained in Section 3.3 without using the DM algorithm. In this context, it could be speculated that it may be possible to apply the codebook \( C_y \) obtained from the reference data set directly to the unsegmented test data set, i.e. use the original codebook for segmentation without codebook adaptation by the DM algorithm. When comparing such a segmentation strategy with the corresponding ‘gold standard’ (see Table 1), we can observe a statistically significant reduction of the inter-method agreement in comparison to the DM approach (Wilcoxon matched pairs signed rank test (Sachs & Reynarowych, 1984, \( \alpha < 0.01 \)). An important reason for this behavior can be derived from measurements of the hard clustering error \( H \) for each vector quantization method. It is calculated by

\[
H = \frac{1}{n} \sum_{i=1}^{n} \min_j \| x_i - w_j(x_i) \|^2,
\]

where \( w_j(x_i) \) denotes the codebook vector located closest to data point \( x_i \). While \( H \) is lowest for direct vector quantization of the data set, it increases after direct translation of the codebook from the reference data set, and can be improved significantly

\( (\alpha < 0.01, \text{see Table 2}) \) after applying the DM algorithm. Taking into account the significance of improvement w.r.t. both the segmentation quality and the quantization error, the additional computational expense for using the DM algorithm can be justified well.

Figs. 10–12 illustrate typical segmentation results in this context. As can be seen from the regions indicated by bounding boxes, the DM algorithm (Figs. 10c–12c) visibly increases the segmentation quality when compared to the results obtained by direct application of the original codebook without using the DM algorithm (Figs. 10d–12d), if the semi-automated GRBF neural network segmentation serves as the ‘gold standard’ (Figs. 10b–12b).

In summary, from a detailed comparative evaluation of the two methods the following conclusions can be drawn. The semi-automated method slightly outperforms the fully automated one w.r.t. segmentation quality, however, it requires a considerable amount of user interaction, i.e. human expert resources. On the other hand, it suffers from significant computational expense. The fully automated method according to Section 3.3 yields almost equivalent results w.r.t. segmentation quality, but does not require any user interaction. In addition, computation time is reduced significantly. Thus, our application demonstrates that data analysis by a strategy of self-organized incremental model adaptation can contribute to increase the efficiency and practicability of biomedical image segmentation systems.
Appendix A. Tables

A.1. Image data acquisition

Table A.1
Acquisition parameters for the four MRI sequences of the 3D data sets

<table>
<thead>
<tr>
<th>T1</th>
<th>T2</th>
<th>PD</th>
<th>IR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Magn. Field strength (T)</td>
<td>1.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Number of images</td>
<td>1.26</td>
<td>63</td>
<td></td>
</tr>
<tr>
<td>Matrix size</td>
<td>512×512</td>
<td>256×256</td>
<td></td>
</tr>
<tr>
<td>Pixel size (mm²)</td>
<td>0.449×0.449</td>
<td>0.898×0.898</td>
<td></td>
</tr>
<tr>
<td>Slice thickness (mm)</td>
<td>1.5</td>
<td>3.0</td>
<td></td>
</tr>
<tr>
<td>TR (ms)</td>
<td>11.6</td>
<td>3710</td>
<td>3710</td>
</tr>
<tr>
<td>TE (ms)</td>
<td>4.9</td>
<td>90</td>
<td>22</td>
</tr>
<tr>
<td>TI (ms)</td>
<td>–</td>
<td>–</td>
<td>223</td>
</tr>
</tbody>
</table>

A.2. Semi-automated segmentation by a GRBF network

A.2.1. Parameters

Table A.2
Parameters employed for vector quantization of the gray level feature space by minimal free energy vector quantization and the GRBF neural network classifier

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>VQ</td>
<td></td>
</tr>
<tr>
<td># Codebook vectors N</td>
<td>50</td>
</tr>
<tr>
<td># Iteration steps</td>
<td>100</td>
</tr>
<tr>
<td>Radius (r_{t=0})</td>
<td>2.0</td>
</tr>
<tr>
<td>Radius (r_{t_{\text{max}}})</td>
<td>0.1</td>
</tr>
<tr>
<td>GRBF</td>
<td></td>
</tr>
<tr>
<td># Classes (m)</td>
<td>3</td>
</tr>
<tr>
<td>Radius (\rho)</td>
<td>0.5</td>
</tr>
</tbody>
</table>

A.2.2. Training data

Table A.3
Statistics of the training data sets used for the GRBF classifier

<table>
<thead>
<tr>
<th>Tissue</th>
<th>Data set 1</th>
<th>Data set 2</th>
<th>Data set 3</th>
<th>Data set 4</th>
<th>Data set 5</th>
<th>Data set 6</th>
<th>Data set 7</th>
<th>Data set 8</th>
<th>Data set 9</th>
<th>Data set 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gray matter</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td># Voxel</td>
<td>4108</td>
<td>3620</td>
<td>3292</td>
<td>3437</td>
<td>2937</td>
<td>3700</td>
<td>2943</td>
<td>4035</td>
<td>5051</td>
<td>3647</td>
</tr>
<tr>
<td>Volume (cm³)</td>
<td>9.95</td>
<td>8.77</td>
<td>7.97</td>
<td>8.32</td>
<td>7.11</td>
<td>8.96</td>
<td>7.13</td>
<td>9.77</td>
<td>12.23</td>
<td>8.83</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>43.05</td>
<td>43.29</td>
<td>43.35</td>
<td>42.78</td>
<td>45.03</td>
<td>43.38</td>
<td>44.46</td>
<td>45.47</td>
<td>47.07</td>
<td>46.87</td>
</tr>
<tr>
<td>White matter</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td># Voxel</td>
<td>4427</td>
<td>3993</td>
<td>3367</td>
<td>3960</td>
<td>2929</td>
<td>3629</td>
<td>2853</td>
<td>3931</td>
<td>4374</td>
<td>3107</td>
</tr>
<tr>
<td>Volume (cm³)</td>
<td>10.72</td>
<td>9.67</td>
<td>8.15</td>
<td>9.59</td>
<td>7.09</td>
<td>8.79</td>
<td>6.91</td>
<td>9.52</td>
<td>10.59</td>
<td>7.52</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>46.39</td>
<td>47.75</td>
<td>44.34</td>
<td>49.28</td>
<td>44.91</td>
<td>42.55</td>
<td>43.10</td>
<td>44.30</td>
<td>40.76</td>
<td>39.93</td>
</tr>
<tr>
<td>CSF</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td># Voxel</td>
<td>1008</td>
<td>749</td>
<td>935</td>
<td>638</td>
<td>656</td>
<td>1200</td>
<td>824</td>
<td>908</td>
<td>1305</td>
<td>1027</td>
</tr>
<tr>
<td>Volume (cm³)</td>
<td>2.44</td>
<td>1.81</td>
<td>2.26</td>
<td>1.55</td>
<td>1.59</td>
<td>2.91</td>
<td>2.00</td>
<td>2.20</td>
<td>3.16</td>
<td>2.49</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>10.56</td>
<td>8.96</td>
<td>12.31</td>
<td>7.94</td>
<td>10.06</td>
<td>14.07</td>
<td>12.45</td>
<td>10.23</td>
<td>12.16</td>
<td>13.20</td>
</tr>
</tbody>
</table>

(continued on next page)
Table A.3 (continued)

<table>
<thead>
<tr>
<th>Tissue</th>
<th>Data set 1</th>
<th>Data set 2</th>
<th>Data set 3</th>
<th>Data set 4</th>
<th>Data set 5</th>
<th>Data set 6</th>
<th>Data set 7</th>
<th>Data set 8</th>
<th>Data set 9</th>
<th>Data set 10</th>
</tr>
</thead>
<tbody>
<tr>
<td># Voxel</td>
<td>9543</td>
<td>8362</td>
<td>7594</td>
<td>8035</td>
<td>6522</td>
<td>8529</td>
<td>6620</td>
<td>8874</td>
<td>10,730</td>
<td>7781</td>
</tr>
<tr>
<td>Volume (cm³)</td>
<td>23.11</td>
<td>20.25</td>
<td>18.39</td>
<td>19.46</td>
<td>15.79</td>
<td>20.65</td>
<td>16.03</td>
<td>21.49</td>
<td>25.98</td>
<td>18.84</td>
</tr>
<tr>
<td>Total ratio (%)</td>
<td>1.54</td>
<td>1.24</td>
<td>1.21</td>
<td>1.22</td>
<td>1.12</td>
<td>1.26</td>
<td>1.06</td>
<td>1.26</td>
<td>1.84</td>
<td>1.26</td>
</tr>
<tr>
<td>Recognition rate (%)</td>
<td>99.6</td>
<td>99.3</td>
<td>98.1</td>
<td>97.3</td>
<td>99.3</td>
<td>97.6</td>
<td>97.1</td>
<td>98.8</td>
<td>98.7</td>
<td>97.1</td>
</tr>
</tbody>
</table>

The table shows the number of voxels labeled as ‘gray matter’, ‘white matter’, and ‘CSF’, the resulting volumes and the corresponding percentage of the whole training data set. Furthermore, the ratio of correct GRBF classification results for the training data set is listed.

A.2.3. Results

Table A.4

Results for classification by a GRBF neural network according to Section 3.2

<table>
<thead>
<tr>
<th>Tissue</th>
<th>Data set 1</th>
<th>Data set 2</th>
<th>Data set 3</th>
<th>Data set 4</th>
<th>Data set 5</th>
<th>Data set 6</th>
<th>Data set 7</th>
<th>Data set 8</th>
<th>Data set 9</th>
<th>Data set 10</th>
</tr>
</thead>
<tbody>
<tr>
<td># Voxel</td>
<td>347,682</td>
<td>374,001</td>
<td>335,073</td>
<td>369,664</td>
<td>329,562</td>
<td>382,609</td>
<td>334,744</td>
<td>382,923</td>
<td>320,004</td>
<td>352,624</td>
</tr>
<tr>
<td>Volume (cm³)</td>
<td>841.94</td>
<td>905.67</td>
<td>855.62</td>
<td>895.17</td>
<td>798.06</td>
<td>926.51</td>
<td>810.61</td>
<td>927.27</td>
<td>774.91</td>
<td>853.90</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>56.02</td>
<td>55.49</td>
<td>54.92</td>
<td>56.09</td>
<td>56.47</td>
<td>56.68</td>
<td>53.75</td>
<td>54.34</td>
<td>54.77</td>
<td>57.12</td>
</tr>
<tr>
<td># Voxel</td>
<td>184,734</td>
<td>208,760</td>
<td>184,965</td>
<td>221,441</td>
<td>177,587</td>
<td>191,185</td>
<td>194,447</td>
<td>235,978</td>
<td>170,184</td>
<td>180,491</td>
</tr>
<tr>
<td>Volume (cm³)</td>
<td>447.35</td>
<td>505.53</td>
<td>447.91</td>
<td>536.23</td>
<td>430.04</td>
<td>462.97</td>
<td>470.87</td>
<td>571.44</td>
<td>412.11</td>
<td>437.07</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>29.77</td>
<td>30.97</td>
<td>29.44</td>
<td>33.60</td>
<td>30.43</td>
<td>28.32</td>
<td>31.22</td>
<td>33.49</td>
<td>29.13</td>
<td>29.24</td>
</tr>
<tr>
<td># Voxel</td>
<td>88,191</td>
<td>91,274</td>
<td>98,310</td>
<td>67,926</td>
<td>43,04</td>
<td>462.97</td>
<td>470.87</td>
<td>571.44</td>
<td>412.11</td>
<td>437.07</td>
</tr>
<tr>
<td>Volume (cm³)</td>
<td>213.56</td>
<td>221.03</td>
<td>238.06</td>
<td>164.49</td>
<td>185.24</td>
<td>245.14</td>
<td>226.58</td>
<td>207.65</td>
<td>227.77</td>
<td>204.02</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>14.21</td>
<td>13.54</td>
<td>15.65</td>
<td>10.31</td>
<td>13.11</td>
<td>15.00</td>
<td>15.02</td>
<td>12.17</td>
<td>16.10</td>
<td>13.65</td>
</tr>
<tr>
<td># Voxel</td>
<td>620,607</td>
<td>674,035</td>
<td>628,348</td>
<td>659,031</td>
<td>583,645</td>
<td>675,025</td>
<td>622,757</td>
<td>704,652</td>
<td>584,245</td>
<td>617,367</td>
</tr>
<tr>
<td>Volume (cm³)</td>
<td>1502.84</td>
<td>1632.22</td>
<td>1521.59</td>
<td>1595.89</td>
<td>1413.34</td>
<td>1634.62</td>
<td>1508.05</td>
<td>1706.36</td>
<td>1414.79</td>
<td>1495.00</td>
</tr>
</tbody>
</table>

The table shows the number of voxels labeled as ‘gray matter’, ‘white matter’, and ‘CSF’, the resulting absolute volumes, the corresponding percentage of the total volume, and the total values for each data set.

A.3. Fully automated segmentation using the DM algorithm

A.3.1. Parameters

Table A.5

Parameters employed for the DM algorithm and the GRBF neural network classifier

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td># Codebook vectors</td>
<td>50</td>
</tr>
<tr>
<td># Iteration steps $t_{\text{max}}$</td>
<td>50,000</td>
</tr>
<tr>
<td>Learning rate $\sigma(t=0)$</td>
<td>0.01</td>
</tr>
<tr>
<td>Learning rate $\sigma(t=t_{\text{max}})$</td>
<td>$10^{-5}$</td>
</tr>
<tr>
<td>Radius $\sigma(t=0)$</td>
<td>0.1</td>
</tr>
<tr>
<td>Radius $\sigma(t=t_{\text{max}})$</td>
<td>$10^{-5}$</td>
</tr>
<tr>
<td># Classes $m$</td>
<td>3</td>
</tr>
<tr>
<td>Radius $\rho$</td>
<td>0.5</td>
</tr>
</tbody>
</table>

A.3.2. Results
Table A.6
Results for segmentation by the DM algorithm according to Section 3.3 and subsequent classification by a GRBF neural network according to Section 3.2

<table>
<thead>
<tr>
<th>Tissue</th>
<th>Data set 2</th>
<th>Data set 3</th>
<th>Data set 4</th>
<th>Data set 5</th>
<th>Data set 6</th>
<th>Data set 7</th>
<th>Data set 8</th>
<th>Data set 9</th>
<th>Data set 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gray matter # Voxel</td>
<td>Gray matter Volume (cm³)</td>
<td>390,427</td>
<td>366,756</td>
<td>374,032</td>
<td>349,585</td>
<td>429,403</td>
<td>361,695</td>
<td>416,472</td>
<td>352,489</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>57.92</td>
<td>58.37</td>
<td>56.75</td>
<td>59.90</td>
<td>63.61</td>
<td>58.08</td>
<td>59.10</td>
<td>60.33</td>
<td>56.97</td>
</tr>
<tr>
<td>White matter # Voxel</td>
<td>White matter Volume (cm³)</td>
<td>197,553</td>
<td>186,498</td>
<td>205,193</td>
<td>166,789</td>
<td>185,732</td>
<td>186,051</td>
<td>225,739</td>
<td>161,294</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>29.31</td>
<td>29.68</td>
<td>31.14</td>
<td>28.58</td>
<td>27.51</td>
<td>29.88</td>
<td>32.04</td>
<td>27.61</td>
<td>30.47</td>
</tr>
<tr>
<td>CSF # Voxel</td>
<td>CSF Volume (cm³)</td>
<td>86,055</td>
<td>75,094</td>
<td>79,806</td>
<td>67,271</td>
<td>59,890</td>
<td>75,011</td>
<td>62,441</td>
<td>70,462</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>12.77</td>
<td>11.95</td>
<td>12.11</td>
<td>11.53</td>
<td>8.87</td>
<td>12.04</td>
<td>8.86</td>
<td>12.06</td>
<td>12.56</td>
</tr>
<tr>
<td>Total # Voxel</td>
<td>Total Volume (cm³)</td>
<td>674,035</td>
<td>628,348</td>
<td>659,031</td>
<td>583,645</td>
<td>675,025</td>
<td>622,757</td>
<td>704,652</td>
<td>584,245</td>
</tr>
</tbody>
</table>

The table shows the number of voxels labeled as ‘gray matter’, ‘white matter’, and ‘CSF’, the resulting absolute volumes, the corresponding percentage of the total volume, and the total values for each data set.

Table A.7
Semi-quantitative evaluation of segmentation quality employing the semi-automated segmentation method using GRBF classification and the fully automated segmentation method using the DM algorithm, respectively

<table>
<thead>
<tr>
<th>Data set 1</th>
<th>Data set 2</th>
<th>Data set 3</th>
<th>Data set 4</th>
<th>Data set 5</th>
<th>Data set 6</th>
<th>Data set 7</th>
<th>Data set 8</th>
<th>Data set 9</th>
<th>Data set 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRBF</td>
<td>1.5</td>
<td>1.6</td>
<td>1.9</td>
<td>1.8</td>
<td>1.6</td>
<td>2.0</td>
<td>2.0</td>
<td>1.7</td>
<td>1.5</td>
</tr>
<tr>
<td>DM</td>
<td>–</td>
<td>1.7</td>
<td>2.1</td>
<td>1.9</td>
<td>1.6</td>
<td>2.0</td>
<td>2.2</td>
<td>1.8</td>
<td>1.6</td>
</tr>
</tbody>
</table>

A.4. Results of the fully automated segmentation without using the DM algorithm

Table A.8
Results for segmentation according to Section 3.3 without application of the DM algorithm, i.e. by direct translation of the codebook from the reference data set. The table shows the number of voxels for data set 3 classified as ‘gray matter’, ‘white matter’, and ‘CSF’, the resulting absolute volumes, the corresponding percentage of the total volume, and the total values for each data set

<table>
<thead>
<tr>
<th>Tissue</th>
<th>Data set 2</th>
<th>Data set 3</th>
<th>Data set 4</th>
<th>Data set 5</th>
<th>Data set 6</th>
<th>Data set 7</th>
<th>Data set 8</th>
<th>Data set 9</th>
<th>Data set 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gray matter # Voxel</td>
<td>Gray matter Volume (cm³)</td>
<td>398,464</td>
<td>396,138</td>
<td>394,971</td>
<td>351,593</td>
<td>441,980</td>
<td>367,073</td>
<td>416,941</td>
<td>353,103</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>59.12</td>
<td>63.04</td>
<td>59.93</td>
<td>60.24</td>
<td>65.48</td>
<td>58.94</td>
<td>59.17</td>
<td>60.44</td>
<td>61.36</td>
</tr>
<tr>
<td>White matter # Voxel</td>
<td>White matter Volume (cm³)</td>
<td>188,816</td>
<td>169,316</td>
<td>193,059</td>
<td>165,289</td>
<td>168,399</td>
<td>176,277</td>
<td>223,446</td>
<td>161,846</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>28.01</td>
<td>26.95</td>
<td>29.29</td>
<td>28.52</td>
<td>24.95</td>
<td>28.31</td>
<td>31.72</td>
<td>27.70</td>
<td>27.58</td>
</tr>
<tr>
<td>CSF # Voxel</td>
<td>CSF Volume (cm³)</td>
<td>86,775</td>
<td>62,894</td>
<td>71,001</td>
<td>66,763</td>
<td>64,646</td>
<td>79,267</td>
<td>64,265</td>
<td>69,296</td>
</tr>
<tr>
<td>Ratio (%)</td>
<td>12.87</td>
<td>10.01</td>
<td>10.77</td>
<td>11.44</td>
<td>9.58</td>
<td>12.73</td>
<td>9.12</td>
<td>11.86</td>
<td>11.06</td>
</tr>
<tr>
<td>Total # Voxel</td>
<td>Total Volume (cm³)</td>
<td>674,035</td>
<td>628,348</td>
<td>659,031</td>
<td>583,645</td>
<td>675,025</td>
<td>622,757</td>
<td>704,652</td>
<td>584,245</td>
</tr>
</tbody>
</table>

The quality of each image of a 3D data set was evaluated by an experienced radiologist on a 5-grade semi-quantitative score (1 = very good, 5 = poor). The table shows the average scores of all the images belonging to a 3D data set.

The table shows the number of voxels labeled as ‘gray matter’, ‘white matter’, and ‘CSF’, the resulting absolute volumes, the corresponding percentage of the total volume, and the total values for each data set.
References


