

Automatic Brain Tumor Segmentation with Normalized Gaussian Bayesian Classifier and Fluid Vector Flow

Tao Wang

Abstract

An automatic brain tumor segmentation method is presented in this paper. This method has 3 stages. In the first stage, a so-called Normalized Gaussian Mixture Model (NGMM) is proposed and used to model the brain tissues. In the second stage, a Gaussian Bayesian Classifier based on the NGMM and the prior probabilities of different brain tissues is exploited to acquire a so-called Gaussian Bayesian Brain Map (GBBM) from the test 3D brain MR images. GBBM is further processed to highlight the brain tumor and initialize a so-called Fluid Vector Flow (FVF) algorithm. In the last stage, FVF is used to segment the brain tumor. The major contribution of this paper is two-fold. First, we present a NGMM to represent healthy brains. This model can be easily modified for modeling other tasks in various application domains. Second, we extend our 2D FVF algorithm to 3D space and use it for automatic brain tumor segmentation. This method has been validated on a publicly available dataset containing 10 T1 Magnetic Resonance (MR) images of 3 types of brain tumor. The results demonstrate that this technique can automatically generate 3D segmentation images of multiple types of brain tumor solely from T1 MRIs.

I. Introduction

Brain tumor segmentation from Magnetic Resonance Imaging (MRI) is an important task for neurosurgeons, oncologists and radiologists to measure the tumor responses to treatment [1]. It can indicate drug efficacy in clinical trials of new drugs, and also be used for planning of radiation therapy. Manual segmentation is time-consuming and frustrating. Therefore, automatic brain tumor segmentation methods have been highly desirable in recent decades [2]. However, automatic brain tumor segmentation is a very challenging job due to many factors [2]. First, different types of brain tumor have high diversity in sizes, shapes, locations and intensities. Second, similarities between brain tumors and normal tissues are often observed. Last but not the least, most brain tumor databases are not publicly available due to political and privacy reasons. It is very difficult to compare and improve brain tumor segmentation techniques based on a same benchmark.

Many brain tumor segmentation methods have been proposed since the last decade. They can be classified into 2 major categories: training based methods and non-training based methods. Note that the distinction between training based and non-training based methods are sometimes blurry. In the literature, those methods can also be classified with other criteria, such as region-based or voxel-based, etc. See Appendix for a comparison of the related methods.

Training based methods often use some brain tumor images to train a segmentation model and use some other images to test the model. Zhu *et al* [3] formulated the brain tumor segmentation as an optimization process that seeks the boundary points to minimize an energy functional based on snakes model. A modified Hopfield Neural Network (HNN) was constructed and trained to solve this optimization problem. The neural network can ensure convergence of the energy minimization by strictly reducing the energy in each iteration. A Knowledge Based (KB)

clustering method [4] with multi-spectral analysis (T1, T2, and PD) was proposed and trained to segment Glioblastoma Multiforme (GBM) [5], an almost non-treatable brain tumor with nearly zero five-year recurrence-free survival rate [6]. The KB system took advantage of its coarse-to-fine operation to apply incremental refinement with easily identifiable brain tissues that had already been located and labelled. Vinitiski *et al* [7] proposed a brain tumor segmentation algorithm based on a 4D (T1, T1c, T2, and PD) feature map. The k-nearest neighbour (KNN) algorithm was then modified by discarding a few image points according to some heuristic rules to speed up segmentation. It also demonstrated that utilizing multiple MRI protocols often provide better segmentation. Warfield *et al.* [8] presented an adaptive, template moderated (ATM), spatially varying statistical classification (SVC) method brain tumor segmentation. Kaus *et al.* [9] extended this idea and proposed a classification algorithm to segment brain MRIs into five different tissue classes (background, skin, brain, ventricles, and tumor). The algorithm had been validated in a dataset of 20 patients with low-grade gliomas and meningiomas. Prastawa *et al* [10, 11] proposed a brain tumor segmentation framework based on Expectation Maximization (EM) algorithm and outlier detection. EM algorithm was used to estimate a mixture Gaussian model for the global intensity distribution, while tumors were considered as outliers of the Gaussian model. Zhang *et al* [12] proposed a brain tumor segmentation method based on unsupervised one-class support vector machine (SVM) algorithm, which had the ability of learning the nonlinear brain tumor distribution without any prior knowledge. Morphological filters such as dilation and erosion operations were applied as the post-processing technique to merge tumor regions and remove isolated and small non-tumor parts. Liu *et al* [13] collected and learned information about different aspects of the tumor and its neighbourhood areas from multiple MRI protocols (FLAIR, T1, and T1 with contrast enhancement). A fuzzy logic framework was then used for tumor segmentation. Cobzas *et al* [2] proposed a variational brain tumor segmentation algorithm using a high dimensional feature set trained from MRI data and registered atlases. The paper focused on how to use a conditional model to discriminate between normal tissues and brain tumors. Wang *et al* [14] transformed the 3D brain tumor MRIs into 2D with a “spiral scanning” technique. Dynamic programming was used to delineate an optimal outline of the brain tumor in the transformed 2D image. The optimal outline was transformed back into 3D space to determine the volume of the tumor. Dube *et al* [15] integrated contextual filter responses into the multilevel segmentation by weighted aggregation (SWA) algorithm to segment GBM. The SWA algorithm used voxel intensities in a neighbourhood to compute an affinity between the respective voxels. The affinity is recursively calculated for every voxel pair in the brain MRIs to generate a series of “cuts” (segments) containing voxels with similar intensities. A contextual filter response that computed by texture filter responses based on the gray level co-occurrence matrix (GLCM) method was then integrated to label the cuts as tumor or non-tumor. Corso *et al* [16] extended [15] and integrated a Bayesian formulation into the SWA algorithm to segment GBM. Lee *et al* presented conditional random fields (CRF) [17] and extended to pseudo-conditional random fields (PCRF) [18] for brain tumor segmentation in 2D. CRF and PCRF incorporated the support vector machine method into discriminative random fields, a discriminative alternative to the traditional Markov random fields model to achieve more tractable computation with less restrictive assumptions. Wels *et al* [19] presented a top-down segmentation algorithm was proposed based on a Markov random field (MRF) and graph cuts (GC). The probabilistic boosting trees (PBT) were used for supervised learning of the model.

Non-training based methods, on the other hand, do not have training or learning process. One of the first non-training based methods of 3D brain tumor segmentation was presented by Phillips *et al* [20] in 1995. Fuzzy C-means (FCM) clustering algorithm was used to segment brain tumor from normal brain tissues. FCM is similar to the k-means algorithm for unsupervised clustering but allows labels to be “fuzzy”, which means a pixel can be partly in one class and partly in others. Clustering is based on the concept of separated distributions. Karayiannis *et al* [21] proposed a fuzzy algorithm for learning vector quantization (FALVQ) to segment the meningioma of an individual. Feature vectors were formed by the values of different relaxation parameters. Brain tumor segmentation was formulated as an unsupervised vector quantization process, which does not rely on *a priori* information. Ho *et al* [22] incorporated region competition into level-set algorithm for brain tumor segmentation. The algorithm started with an intensity based fuzzy logic classification of voxels into tumor and background to create a tumor probability map. This map was then used to initialize the level-set snake. The snake model was driven by the image forces balanced with global and local constraints to segment a brain tumor until convergence is achieved. Ray *et al* [23] observed that normal brain structures are symmetric, which is often disturbed by brain tumors. This property was utilized to design a so-called brain tumor locator score (BTLS) for tumor segmentation.

In this paper, an automatic and operator-independent brain tumor segmentation method is presented and validated on a publicly available brain tumor segmentation repository; the Surgical Planning Laboratory (SPL) Brain Tumors Image Database [8, 9, 24]. This database contains T1 MR images of 10 patients. 3D segmentation ground-truth is also available. This database was released in December 2007 and makes possible to compare and improve brain tumor segmentation techniques on the same benchmark.

In our method, brain MR images are pre-processed with software MIPAV [25, 26]. After this pre-processing procedure, there are 3 stages. In the 1st stage, a so-called Normalized Gaussian Mixture Model (NGMM) is proposed and estimated by Expectation-Maximization (EM) based on the ICBM452 brain atlas [27]. NGMM is then used to model the healthy or normal brain tissues. In the 2nd stage, the ICBM Tissue Probabilistic Atlases [28] are utilized to obtain the prior probabilities of different brain tissues. After that, a Gaussian Bayesian Classifier based on the NGMM and the prior probabilities of different brain tissues is exploited to acquire a so-called Gaussian Bayesian Brain Map (GBBM) from the test 3D brain MR images. GBBM is further processed to highlight the brain tumor and initialize a so-called Fluid Vector Flow (FVF) algorithm. In the last stage, FVF is used to segment the brain tumor.

There are 2 major contributions in this paper. First, we present a NGMM to represent healthy or normal brain. This model can be easily modified for modeling other tasks in various application domains. Second, we extend our 2D FVF algorithm [29] to 3D space and use it for automatic brain tumor segmentation. One drawback of our previous 2D FVF algorithm was that an initial contour must be given to start the vector flow evolution. In this paper, we take advantage of the GBBM to provide an initial position of a brain tumor to the 3D FVF algorithm to make this process fully automatic.

The rest of this paper is organized as follows. In Section II, the materials including the test dataset and brain atlases are briefly described. Section III introduces the proposed brain tumor

segmentation method. Experimental results are reported in Section IV, before the work is concluded in Section V.

II. Materials

II.1. Test Dataset

SPL Brain Tumors Image Dataset [8, 9, 24] is a freely available brain tumor segmentation repository. It contains brain MR images (SPGR T1 POST GAD) of 10 patients (3 meningiomas, 3 low grade gliomas and 4 astrocytomas). Segmentation ground-truth, which was defined as the area of those brain tumor voxels in which at least three of four expert raters agreed regarding their identification [9], is also available. The image format is no-header, unsigned short 16-bit (byte order: MSB LSB). The resolution is 256x256x124, with pixel size of 0.9375 x 0.9375 mm, slice thickness of 1.5 mm, slice gap of 0.0 mm, and the acquisition order of LR.

II.2. Brain atlases

Two groups of brain atlases, the Talairach–Tournoux (TT) brain atlas [30, 31] and the ICBM atlases [27, 28] are often used to collect healthy or normal brain information.

The print TT atlas was constructed from a single brain specimen sectioned and photographed sagittally. Coronal and axial sections were subsequently interpolated manually. The Cerefy database [32] contains an extended and enhanced electronic version of the TT brain atlas. It can read/write DICOM 3 format. It is free Java software including the atlas. However, the atlas itself is not freely accessible.

In this paper, we use two ICBM atlases, the ICBM452 atlas [27] and the ICBM Tissue Probabilistic Atlases [28], which are freely accessible in the public domain.

The ICBM452 atlas [27] is a freely available brain atlas. It is an average of intensities and spatial positioning of T1-weighted MR images of normal adult brains. This atlas is not based on any single subject but is constructed from the average position, orientation and scale from a number of individual brains. The ICBM452 atlas is used to estimate a Normalized Gaussian Mixture Model (NGMM) by the Expectation-Maximization (EM) algorithm in this paper.

The ICBM Tissue Probabilistic Atlases [28] classified the ICBM452 atlas into gray matter (GM), white matter (WM), and cerebrospinal fluid (CSF). The GM, WM, and CSF maps were separated into their separate components. Each component was then averaged in atlas space across the subjects to create the probability fields for each tissue type. These fields represent the likelihood of finding GM, WM, or CSF at a specified position for a subject that has been registered to the atlas space. The ICBM Tissue Probabilistic Atlases are used to obtain the prior probabilities of GM, WM, and CSF in this paper.

III. Proposed Method

III.1. Pre-processing

MR images must be pre-processed for further processing and analysis. The Neuroimaging Informatics Tools and Resources Clearinghouse (NITRC) [33] provides the links to a variety of brain MR image processing software. We choose to use MIPAV [25, 26] after carefully investigation and comparison. MIPAV (Medical Image Processing, Analysis, and Visualization) is a Java application mainly for processing and analysis of brain MR images. It can run on Java-enabled systems such as Windows, UNIX, or Macintosh OS X. We also notice some other powerful tools such as 3D Slicer [34], FSL [35], STAPLE [36], BioImage Suite [37], ITK-SNAP [38], and software suites provided by Asclepius [39] at INRIA, LONI [40], etc. They focus on different aspects of 3D neuro-imaging and demonstrate excellence in different applications. We choose MIPAV because it suits our application, the dataset, and computational resources.

The pre-processing stage has two major steps: skull stripping and registration. Other pre-processing steps, such as noise reduction and inter-scan intensity standardization are described in details in [2] and [19].

SPL Brain Tumors Image Dataset [8, 9] contains the MR scans of patients' heads and structures nearby. To get the Volume Of Interest (VOI), *i.e.*, the brain, we must strip the skull and other non-brain structures and tissues. We use MIPAV to extract brain from MR images. Being given the input MR images, MIPAV can automatically extract brains without user intervention. Fig 1 shows the original MR image, the extracted brain, and the 3D view of the extracted brain of patient #1 in the dataset.

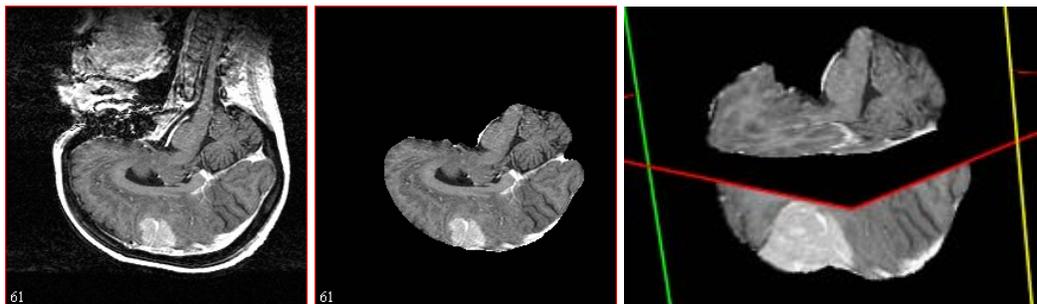


Fig 1. (Left) Original image (Middle) extracted brain (Right) 3D view of the extracted brain

The extracted brain is then registered [41] to the ICBM atlas space with MIPAV. The basic idea of registration is to find a matrix T that transform the extracted brain image to the atlas so that the cost function, which represents the quality of alignment between two images, is minimized. Fig 2 shows the ICBM452 atlas, the registered brain and the 3D view of the registered brain of patient #1.

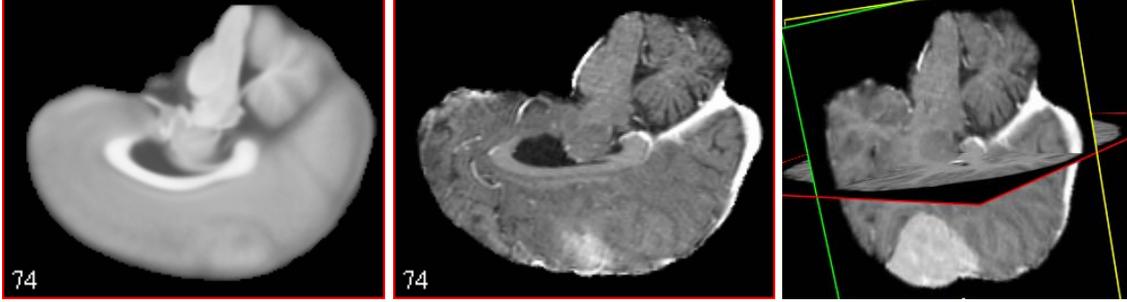


Fig 2. (Left) ICBM452 atlas (Middle) registered brain (Right) 3D view of the registered brain

III.2. Normalized Gaussian Mixture Model and Gaussian Bayesian Brain Map

In this section, a so-called Normalized Gaussian Mixture Model (NGMM) is proposed. Gaussian Mixture Model (GMM) had been utilized to represent the distribution of brain tissues in the literature [10, 11], and Expectation Maximization (EM) [42, 43, 44] algorithm was often used to learn the mixture model from a brain atlas or template. The basic idea of GMM is to use multiple Gaussian distributions to represent multiple brain tissues such as gray matter (GM), white matter (WM), and cerebrospinal fluid (CSF). To utilize GMM, the Gaussian distributions of the brain atlas and the brain tumor dataset must be aligned correspondingly. This means the Gaussian distribution of the GM in the brain atlas must be in the similar range of the Gaussian distribution of the GM in the brain tumor dataset, and the distributions of WM and CSF must match their peers too. Unfortunately, this is not true in our dataset.

Fig 3 shows the histogram of the ICBM452 atlas and the histogram of the MRIs of patient #1. The intensity regions of CSF, GM, and WM are marked. The intensity range of the atlas is [0, 712] while the intensity range of the MRIs of patient #1 is [0, 567]. Note the Gaussian distribution of the GM in the brain atlas does not match the Gaussian distribution of the GM in the brain tumor dataset, and the distributions of WM and CSF do not match their peers.

Contrast stretching technique is usually used to in this situation. If the lower and the upper intensity limits are a and b , the lowest and highest intensity values present in the current image are c and d respectively, the intensity value of current voxel is P_{in} , the stretched intensity value P_{out} is defined as

$$P_{out} = (P_{in} - c) \left(\frac{b - a}{d - c} \right) + a \quad (1)$$

However, this approach does not work in our test dataset because of the “long tail” problem that exists in all patients’ data. A “long tail” in histogram of the MR images of patient #1 can be clearly observed in Fig 3 (right). The “long tail” region contains a small number of voxels with high intensity values. These voxels could represent image noise and/or brain abnormalities. Noise filters can reduce the noise at the risk of eliminating potential brain abnormalities with high abnormalities, and cannot remove the “long tail” entirely. Contrast stretching cannot align the Gaussian distributions with the existence of the “long tail”.

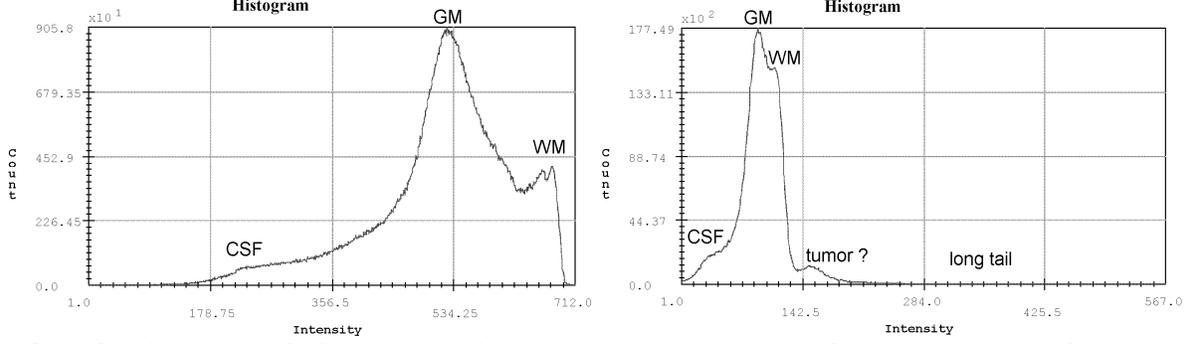


Fig 3. (left) histogram of ICBM452 brain atlas. (right) histogram of the MR images of patient #1.

To align the Gaussian distributions without eliminating potential brain abnormalities, we define the normalized intensity value P_{out} as:

$$P_{out} = \frac{P_{in}}{m} \quad (2)$$

where m is the mean of image intensities.

With this definition, the mean value of image intensity is normalized to 1 and the majority of image intensity is stretched to certain range around the mean value. In our experiment, 100% of voxels in the ICBM452 brain atlas and more than 99.9% of voxels in the test dataset are normalized into the range of $[0, 1.40]$. The Gaussian distributions are also aligned so that GMM can be utilized.

We present a Normalized Gaussian Mixture Model (NGMM) to represent the healthy or normal brain in the ICBM452 atlas. The basic idea of the NGMM is to estimate a Gaussian Mixture Model based on the normalized image intensities. The NGMM is defined as:

$$p(x_i) = \sum_{k=1}^K p(k)p(x_i | k) \quad (3)$$

where x_i is a voxel in the image, $p(k)$ is the prior probability, and $p(x_i | k)$ is the conditional probability density, which is define as:

$$p(x_i | k) = N(x_i; \mu_i, \sigma_i) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{(x_i - \mu_i)^2}{2\sigma_i^2}\right) \quad (4)$$

where μ_i is the mean and σ_i is the standard deviation of the Gaussian $N(x_i; \mu_i, \sigma_i)$.

Expectation Maximization

The Expectation Maximization (EM) algorithm [42, 43, 44] is often used to estimate the parameters of a Gaussian Mixture Model distribution. Quasi-Newton Method [45] had been proposed to accelerate the EM algorithm. The EM algorithm has two stages: *expectation* and *maximization*.

Expectation

With an initial guess for the parameters of the GMM, partial membership of each image voxel in each distribution is estimated by computing expectation values for the membership variables of each data voxel. For each data voxel x_j and distribution Y_i , the membership value $y_{i,j}$ is:

$$y_{i,j} = \frac{a_i f_Y(x_j)}{f_X(x_j)} \quad (5)$$

Maximization

Once the expectation values are computed for group membership, estimates are re-calculated for the distribution parameters. The blending coefficients a_i are the means of the membership values over the N data voxels.

$$a_i = \frac{1}{N} \sum_{j=1}^N y_{i,j} \quad (6)$$

The mean values θ_i are also computed by expectation maximization using image voxels x_j that have been weighted using the membership values.

$$\theta_i = \frac{\sum_j y_{i,j} x_j}{\sum_j y_{i,j}} \quad (7)$$

With new estimates for the blending coefficients a_i and mean values θ_i , the process will be repeated again until the parameters of GMM converge.

K-means

The initial guess for the parameters of the GMM is often given by K-means algorithm [46, 47, 48]. Given a set of N data points (x_1, x_2, \dots, x_N) , the K-means algorithm divides the N observations into K sets ($K < N$) $S = \{S_1, S_2, \dots, S_K\}$ to minimize the within-set sum of squares.

$$\arg \min_S \sum_{i=1}^K \sum_{x_j \in S_i} (x_j - \theta_i)^2 \quad (8)$$

where θ_i is the mean value of set S_i .

In this paper, $K=3$ because we aim to represent 3 types of brain tissues, *i.e.*, CSF, GM, and WM.

Prior probabilities

The ICBM Tissue Probabilistic Atlases are utilized to obtain the prior probabilities of different brain tissues. At a given voxel x_i , the prior probability $p(k, x_i)$ is defined as:

$$p(k, x_i) = \frac{\xi(k, x_i)}{\sum_{k=1}^K \xi(k, x_i)} \quad (9)$$

where $\xi(k, x_i)$ is image intensity of different brain tissues in of the three ICBM Probabilistic Atlases (CSF, GM, or WM).

Gaussian Bayesian Brain Map

At a given 3D location (u, v, w) the post probability $p(k | \xi)$ can be calculated by Bayes' Theorem:

$$p(k | \xi) = \frac{p(k) \cdot p(\xi | k)}{\sum_{k=1}^K p(k) \cdot p(\xi | k)} \quad (10)$$

The correlation coefficient $CC \in [-1,1]$ at this voxel is then calculated:

$$CC = \frac{Cov(\alpha, \beta)}{\sqrt{Cov(\alpha, \alpha) \cdot Cov(\beta, \beta)}}$$

where $\alpha = (p(CSF | \xi), p(GM | \xi), p(WM | \xi))$, $\beta = (p(CSF), p(GM), p(WM))$, and Cov is covariance.

The correlation coefficient CC can reveal the likelihood of finding a candidate tumor voxel at a given 3D location. When CC is close to -1, it means the intensity of this voxel disagrees with the NGMM and this voxel is probably abnormal or likely to be a tumor voxel. When CC is close to 1, it means the intensity of this voxel agrees with the NGMM and this voxel is likely to be normal. The correlation coefficient is then used to define the Gaussian Bayesian Brain Map ($GBMM$):

$$CM_{uvw} = \begin{cases} 1.0 - CC_{uvw} & \text{when } CC_{uvw} > 0.0 \\ 0.0 - CC_{uvw} & \text{elsewise} \end{cases} \quad (11)$$

$$GBMM = [a_{uvw}]_{m \times n \times o} \quad \text{where } a_{uvw} = \Omega \cdot CM_{uvw} \quad (12)$$

CC is first mapped to CM in the region of $[0, 1]$. Then $GBMM$ is calculated based on CM and Ω . Ω is a scaling parameter that represents the intensity range of $GBMM$. $\Omega = 127.0$ in this paper. The resulting $GBMM$ a 3D matrix or image with dimension of $m \times n \times o$, m, n and $o \in N$. It can reveal the likelihood of finding candidate tumor in the entire image domain.

III.3. Post-processing

$GBMM$ is further processed to highlight candidate tumor region. This region will be used to automatically initialize the 3D Fluid Vector Flow algorithm, which will finally segment the brain tumor. The post-processing stage has 5 steps: removing boundary voxels, thresholding, morphological erosion, locating the largest 3D region, morphological dilation, and reverse transformation.

Brain tumor MR images are registered to brain atlas in the pre-processing stage. However, the registered image and the atlas are still different at a few voxels on the brain boundary. Therefore, the boundary voxels often have high intensity values. However, the boundary voxels are not in the region of any brain tumor in our dataset. Therefore, boundary voxels must be removed from *GBMM*. A boundary voxel is defined as a voxel that has at least one neighbor with intensity 0 in $3 \times 3 \times 3$ neighborhood.

A threshold Ψ is then applied to *GBMM* to create a Binary Gaussian Bayesian Brain Map (*BGBBM*):

$$b_{uvw} = \begin{cases} 1 & \text{when } a_{uvw} > \Psi \\ 0 & \text{elsewise} \end{cases} \quad (13)$$

$$BGBBM = [b_{uvw}]_{m \times n \times o} \quad (14)$$

In this paper, $\Psi = (\Omega + 1) / 2 = 64.0$

Morphological filters (dilation, erosion, etc) are often used as the post-processing techniques [12]. Erosion is applied to merge separated relatively big regions. Then, the largest 3D region is automatically located. This region represents the candidate tumor region. After that, dilation is used to restore it to approximately its original size and shape before the erosion. Fig 4 shows the registered brain image, the *GBMM*, and the candidate tumor region after the dilation step.

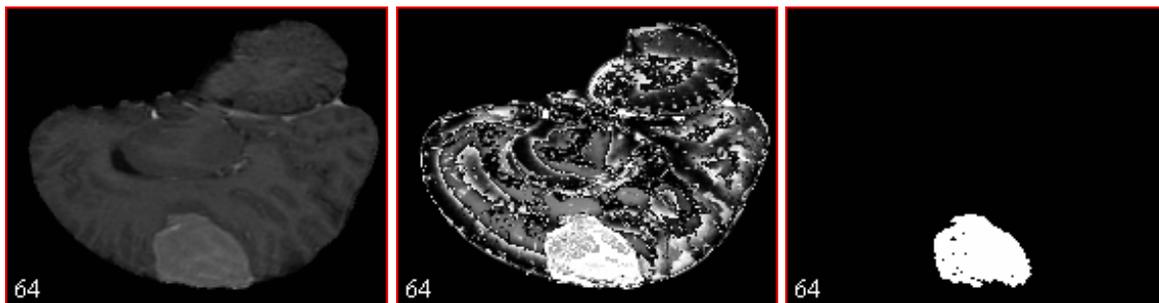


Fig 4. (Left) Registered brain image (Right) Gaussian Bayesian Brain Map of the brain (right) The candidate tumor region after dialation.

At last, we apply T^{-1} to the dilated image to transform it back to the original image space. T is the registration matrix calculated in the pre-processing stage. Fig 5 shows the original image, the segmentation ground-truth, and the candidate tumor region after the reverse transformation. The segmentation ground-truth was defined as the area of those brain tumor voxels in which at least three of four expert raters agreed regarding their identification [9]. Although the difference is still noticeable, the candidate tumor region is similar to the segmentation ground-truth. This candidate tumor region will be used to initialize the 3D Fluid Vector Flow algorithm, which will finally segment the brain tumor.

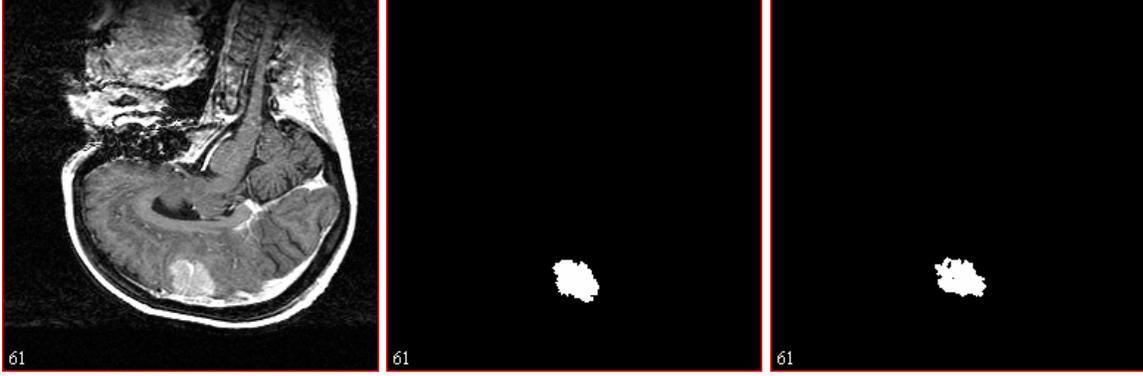


Fig 5. (left) Original image (middle) ground-truth (right) candidate tumor region after the reverse transformation

III.4. 3D Fluid Vector Flow

Fluid Vector Flow (FVF) [29] is an active contour model that addresses other active contour models' problems such as insufficient capture range and poor convergence for concavities. It had been applied to semi-automatic brain tumor segmentation in 2D space. One drawback of the previous 2D FVF algorithm was that an initial contour must be given to start the vector flow evolution. In this paper, FVF algorithm is extended to 3D space and the candidate tumor region is used to initialize the 3D FVF algorithm to make this process fully automatic.

In this paper, segmentation is formulized as the process of finding the surface which can delineate the region of brain tumor given the candidate tumor region in the 3D space of the MRIs. This is a nonlinear optimization problem, which is often addressed by using a variational approach [2]. The basic idea is to give an initial candidate tumor model and deform this model to better delineate the brain tumor until convergence is achieved. This deformable model is usually called active contour model or snakes [29, 49, 50]. Active contour models or snakes have been adopted as effective tools for segmentation [22, 29] in the literature.

The traditional active contour model proposed by Kass *et al* [49] is a 2D parametric active contour:

$$c(s) = (x(s), y(s)), s \in [0,1] \quad (15)$$

Given an initial contour, it evolves within an image $I(x, y)$ to minimize the energy function:

$$E_{snake} = \int_0^1 [E_i(c(s)) + E_e(c(s))] ds \quad (16)$$

where E_i is the internal (spline) energy and E_e is the external energy.

$$\text{The internal energy is given by: } E_i = \frac{\alpha(s) |c'(s)|^2 + \beta(s) |c''(s)|^2}{2} \quad (17)$$

where α and β are first-order and second-order blending parameters.

Many snakes in the literature share the same internal energy and differ mostly in the external energy. A snake should evolve to minimize the energy functional E_{snake} . This problem can be formulated with the Euler-Lagrange equation.

$$\alpha c''(s) - \beta c''''(s) + \nabla E_e = 0 \quad (18)$$

To find a numeric solution of (16), the snake is treated as a function of time t as well as s :

$$\alpha c''(s, t) - \beta c''''(s, t) + \nabla E_e = 0 \quad (19)$$

A solution is obtained when the contour stabilizes and the time term vanishes [49]. There are two problems with the traditional active contour model. The first problem is its limited capture range, *i.e.*, it is not able to deform far from its initial contour. The second problem is its poor convergence for concavities. These two problems were addressed in our previous work [29].

While parametric active contour models mainly focus on the 2D domain, level set snakes had been used for 3D segmentation and reconstruction [50]. A level set [51, 52, 53] snake is an implicit model, which is not explicitly expressed as a parametric model but is implicitly specified as a level set of a scalar function ϕ . A 3D surface may be written as:

$$f(x, y, z) = (x(s_1, s_2), y(s_1, s_2), z(s_1, s_2)) \quad (20)$$

$F = F(K)$ is the speed function of the surface evolution

$$F^2(f_x^2 + f_y^2 + f_z^2) = 1 \quad (21)$$

where K is the mean curvature:

$$K = \frac{f_{xx}(f_y^2 + f_z^2) + f_{yy}(f_x^2 + f_z^2) + f_{zz}(f_y^2 + f_x^2) - 2f_{xy}f_xf_y - 2f_{xz}f_xf_z - 2f_{yz}f_zf_y}{(f_x^2 + f_y^2 + f_z^2)^{3/2}} \quad (22)$$

The motion of that surface is formulated as a Partial Differential Equation (PDE):

$$\frac{\partial \phi}{\partial t} - F(K) |\nabla \phi| = 0 \quad (23)$$

where $\phi(x, y, z, t)$ is a scalar function such that at time t the zero level set of ϕ is the surface.

Level set snakes have a few drawbacks. The 1st drawback is that it requires a lot of computations to solve these equations over the entire domain. Adalstein and Sethian [52] proposed a fast level set method for propagating interfaces based on a narrow-band method, which used a finite band of 6–12 grid voxels on either side of the level set to reduce the computations. Whitaker [50] presented a sparse-field method, which took the narrow-band method to its extreme by calculating updates on a band of grid voxels that is only one voxel wide. The 2nd drawback is that level set snakes may segment wrong objects when there are multiple similar objects near the expected object. The 3rd drawback of level set method is the absence of explicit and direct representation of the surface during its evolution.

3D Fluid Vector Flow (FVF) is used to segment the brain tumor in this paper. It is an extension of our 2D FVF algorithm [29], which had been applied to semi-automatic brain tumor segmentation in 2D space. 3D FVF takes the candidate tumor region obtained in the previous

section and fully automatically segment the brain tumor. It uses a parametric representation of the initial surface and also takes advantage of the level set method.

We first address the 2nd drawback of level set snakes: they may segment wrong objects when there are multiple similar objects near the expected object. Fig 6 demonstrates this problem. There are 4 objects in the left figure. The upper left one is the expected object and the rectangle is the initial contour given by user. Level set snakes segment all the 4 objects, which is not the desired result because real intension of the user is to segment the upper left object only.

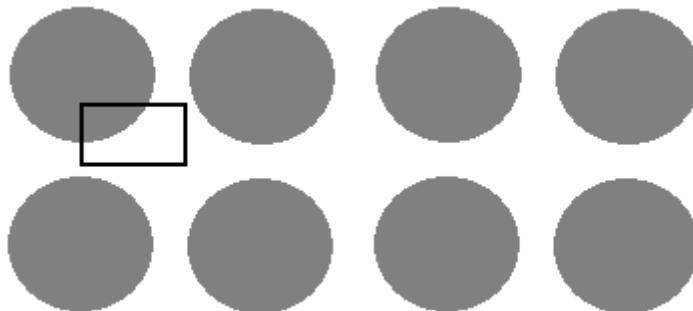


Fig. 6. (Left) initial rectangle contour and 4 objects (Right) 4 objects are segmented by level set snakes.

To address this problem, we need to determine the spatial relationship between the initial contour and the object. Note that we explain this concept in 2D only for the purpose of illustration. The algorithm is implemented in 3D. Once we have obtained the candidate tumor region, we can calculate its “center”.

The mostly often used definition of center is arithmetic mean. For a given region R with n object voxels $\{P_1, P_2, \dots, P_n\}$, the arithmetic mean is defined as:

$$\sum_{i=1}^n P_i / n \quad (24)$$

However, the center defined by arithmetic mean may be outside a given region. Figure 7 (Left) shows the center is out of the given region. In this figure, white points are object points and black points form background. The red points are the centers in different definitions. In [54], some other frequently used definitions of center, such as Tukey median, Liu median, Oja median, depth-based trimmed mean, coordinate median, and spatial median, are discussed and compared. According to [54], spatial median stands as the best overall. The spatial median is defined as:

$$\arg \min_p \left(\sum_{i=1}^n |p - P_i| / n \right) \quad (25)$$

where $|\cdot|$ is the Euclidean distance. Spatial median works very well for convex regions. However, in the context of this paper, the brain tumor may not be convex therefore spatial median may fail. Figure 7 (Middle) demonstrates that the center defined by spatial median is on the boundary of the region.

We proposed a Valence Driven Spatial Median (VDSM) algorithm in [55] to compute the center of a region. The center $A = (x_0, y_0, z_0)$ is given by:

$$\arg \min_p \left(\sum_{i=1}^n \frac{1}{V(p)} |p - P_i| / n \right) \quad (26)$$

where $V(p)$ is the valence of the point p . The valence $V(p)$ of an object point p is defined as the number of object points in p 's $3 \times 3 \times 3$ neighborhood, excluding p itself. $\frac{1}{V(p)}$ assigns penalties to boundary points which have smaller valences than inside points. As a result, the derived center is attracted towards the middle of the region.

In our implementation, we modify Equation (26) to Equation (27) to eliminate the computational cost in performing the square root and division operations.

$$\arg \min_p \left(\sum_{i=1}^n \frac{1}{V(p)} |p - P_i|^2 \right) \quad (27)$$

Since n is a constant, removing n does not affect the value for which the expression attains its minimum. Fig. 7 (Right) shows the center obtained by using VDSM.

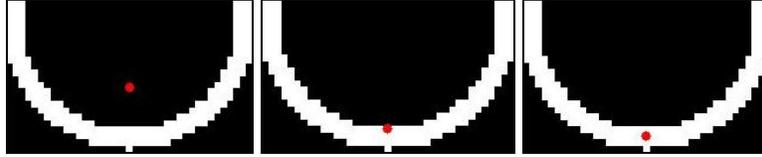


Fig. 7. The red (gray in B&W) point denotes the “center” of a region (white). From left to right, the locations of center defined by arithmetic mean, spatial median and VDSM are shown respectively.

Now we have the candidate tumor region and its center, no matter the region is convex or concave. The initial active surface S is then defined as

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = r^2 \quad (28)$$

where (x_0, y_0, z_0) is the center of the candidate tumor region and radius r is a constant. To make sure that the sphere is inside the candidate tumor region, we require r to be smaller than the radius of the maximum inscribed sphere of the candidate tumor region at point (x_0, y_0, z_0) . Then, equation (23) becomes:

$$\begin{cases} \phi(0, x, y, z) = S \\ \frac{\partial \phi}{\partial t} = F(K) |\nabla \phi| \end{cases} \quad (29)$$

Chan and Vese [56] pointed out the evolution of level set should not always rely on gradient (2D) or surface normal (3D). The basic idea of FVF is to add a directional component to the

external force and keep the normal component. At a given point $B = (x_1, y_1, z_1)$ on the level set surface, there are two straight line L_1 and L_2 :

$$L_1 : \frac{x-x_1}{l_1} = \frac{y-y_1}{m_1} = \frac{z-z_1}{n_1} \quad (30)$$

$$L_2 : \frac{x-x_1}{l_2} = \frac{y-y_1}{m_2} = \frac{z-z_1}{n_2} \quad (31)$$

L_1 is the normal of the surface at point (x_1, y_1, z_1) and L_2 is the straight line determined by vector \vec{AB} that starting from center $A = (x_0, y_0, z_0)$ and pointing to $B = (x_1, y_1, z_1)$. See Fig 8.

Then, we extend the external energy function in [29] to 3D

$$E_e(x, y, z) = \chi(f_x + \delta\gamma_x, f_y + \delta\gamma_y, f_z + \delta\gamma_z) \quad (32)$$

where χ is a normalization operator, $\delta = \pm 1$ (controls the inward or outward direction, when the surface is “outside” or “inside” the candidate tumor region), and γ is the angle between L_1 and L_2 , and γ is determined by:

$$\cos \gamma = \frac{l_1 l_2 + m_1 m_2 + n_1 n_2}{\sqrt{l_1^2 + m_1^2 + n_1^2} \sqrt{l_2^2 + m_2^2 + n_2^2}} \quad (33)$$

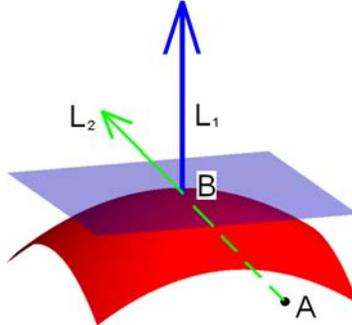


Fig. 8. The red surface is the level set surface, the blue plane is the tangent plane to that surface, the blue arrow is the surface normal, the black dot is the center of the candidate tumor region, and the green arrow represents the directional component of the external energy.

The external energy $E_e(x, y, z)$ has two components: a normal component and a directional component. The normal force is computed in a manner similar to the traditional snake [49] and GVF snake [57]. The characteristic of FVF lies in the computation of the directional force. When the active surface is within the candidate tumor region, the directional force can push the surface towards the boundary of tumor. When the surface is close to the boundary of tumor, the normal force fits the surface to the tumor.

IV. Experimental Results

We test the proposed method with the SPL Brain Tumors Image Dataset [8, 9, 24]. The Tanimoto Metric [58] is used for quantitative analysis. Tanimoto Metric is defined as:

$$TM = \frac{\|R_X \cap R_G\|}{\|R_X \cup R_G\|}, 0 \leq TM \leq 1, \text{ where } R_X \text{ is the region enclosed by the surface generated by the}$$

proposed method, R_G is the region of the ground-truth segmentation provided in the SPL Brain Tumors Image Dataset, and $\|\cdot\|$ is set cardinality (number of elements). $TM = 0$ would indicate two completely distinct segmentation; while $TM = 1$ would indicate completely identical segmentation. Table I shows the test results.

Table I: Tanimoto metric of the proposed method

Case	Tumor type	GBMM time	FVF time	Total time	Tanimoto score
1	meningioma	361 sec	51 sec	412 sec	0.88
2	meningioma	372 sec	51 sec	423 sec	0.83
3	meningioma	375 sec	52 sec	427 sec	0.57
4	low grade glioma	374 sec	50 sec	424 sec	0.66
5	astrocytoma	375 sec	51 sec	426 sec	0.22
6	low grade glioma	389 sec	52 sec	441 sec	0.53
7	astrocytoma	388 sec	51 sec	439 sec	0.67
8	astrocytoma	384 sec	53 sec	437 sec	0.57
9	astrocytoma	361 sec	56 sec	417 sec	0.30
10	low grade glioma	397 sec	52 sec	449 sec	0.70

The proposed method was tested on a Pentium 4 (3GHz CPU, 2GB RAM) desktop computer with Windows XP (Version 2002, SP3) operating system. The resolution is 256x256x124 for each test case. The proposed method is implemented in MATLAB 7.5 (R2007b) and not optimized for speed.

In Table I, the GBMM time represents the CPU time of calculating the Gaussian Bayesian Brain Map (GBMM), the FVF time reveals the CPU time of executing the proposed 3D Fluid Vector Flow (FVF) algorithm to segment brain tumor. The total time is the sum of GBMM time and the FVF time, which shows the entire time consumption of proposed method. The pre-processing stage requires about 9 minutes. The post-processing stage requires about 5 minutes. So the total overhead time is about 14 minutes for each test case. Note that the Normalized Gaussian Mixture Model (NGMM) was trained off-line. The training time was about 25 minutes. Once the training was finished, the result had been used repeatedly in each test case.

V. Conclusions and discussions

A brain tumor segmentation method is presented and validated in this paper. The method is able to segment brain tumors fully automatically by taking advantages of the proposed Normalized Gaussian Mixture Model (NGMM) and 3D Fluid Vector Flow (FVF) algorithm. This technique can be utilized to generate brain tumor segmentation images that display clinically important neuroanatomic and neuropathologic information.

Utilizing multiple MRI protocols often provide easier segmentation [7]. Other factors that can simply segmentation include using 2D images, semi automatic implementation, and testing fewer types of tumor. Our technique implements fully automatic 3D segmentation of 3 types of brain tumor in T1 MRIs. It over-performs the other discussed methods by segmenting most types of brain tumor with least MRI protocol. The accuracy (0.22-0.88) of our method matches a recent work by Corso *et al* [16], where the accuracy was in the range of 0.27-0.88 for segmentation of only one type of tumor glioblastoma multiforme (GBM).

Acknowledgement

The authors would like to thank Drs. Simon Warfield, Michael Kaus, Ron Kikinis, Peter Black and Ferenc Jolesz for sharing the brain tumor database [8, 9].

References

- [1] K. Brindle, New approaches for imaging tumour responses to treatment, *Nature Reviews Cancer* vol 8, pp. 94-107, 2008.
- [2] D. Cobzas, N. Birkbeck, M. Schmidt, M. Jagersand and A. Murtha, 3D variational brain tumor segmentation using a high dimensional feature set, *MMBIA 2007*, in conjunction with *ICCV 2007*.
- [3] Y. Zhu and H. Yan, Computerized tumor boundary detection using a hopfield neural network, *IEEE Trans. Med. Imag.*, vol. 16(1), pp.55–67, Feb. 1997.
- [4] M. C. Clark, L. O. Hall, D. B. Goldgof, R. Velthuizen, R. Murtagh, and M. S. Silbiger, Automatic tumor segmentation using knowledge-based techniques, *IEEE Trans. Med. Imag.*, vol. 17(2), pp. 187–201, Apr. 1998.
- [5] J. G. Smirniotopoulos, The new WHO classification of brain tumors, *Neuroimag. Clinics North Amer.*, vol. 9, no. 4, pp. 595–613, Nov. 1999.
- [6] M. R. Patel and V. Tse, Diagnosis and staging of brain tumors, *Seminars Roentgenol.*, vol. 39, no. 3, pp. 347–360, 2004.
- [7] S. Vinitiski, C. F. Gonzalez, R. Knobler, D. Andrews, T. Iwanaga, and M. Curtis, Fast tissue segmentation based on a 4D feature map in characterization of intracranial lesions, *J. Magn. Reson. Imag.*, vol. 9, no. 6, pp. 768–776, 1999.
- [8] S. K. Warfield, M. Kaus, F. A. Jolesz, and R. Kikinis, Adaptive, Template Moderated, Spatially Varying Statistical Classification, *Med Image Anal.* Vol 4(1):43-55, 2000.
- [9] M. Kaus, S. K. Warfield, A. Nabavi, P. M. Black, F. A. Jolesz, and R. Kikinis. Automated Segmentation of MRI of Brain Tumors, *Radiology.* 218(2):586-91, 2001.
- [10] M. Prastawa, E. Bullitt, N. Bullitt, K. V. Leemput, and G. Gerig, Automatic brain tumor segmentation by subject specific modification of atlas priors, *Acad. Radiol.*, vol. 10, pp. 1341–1348, Dec. 2003.
- [11] M. Prastawa, E. Bullitt, S. Ho, and G. Gerig, A brain tumor segmentation framework based on outlier detection, *Med. Image Anal. J.*, vol. 8, no. 3, pp. 275–283, Sep. 2004.
- [12] J. Zhang, K. Ma, M. H. Er, and V. Chong, Tumor segmentation from magnetic resonance imaging by learning via one-class support vector machine, in *Proc. Int. Conf. Intell. Mechatronics Automation*, pp. 207–211, 2004.
- [13] J. Liu, J. K. Udupa, D. Odhner, D. Hackney, and G. Moonis, A system for brain tumor volume estimation via MR imaging and fuzzy connectedness, *Comput. Med. Imag. Graphics*, vol. 29, no. 1, pp. 21–34, 2005.

- [14] J. Wang, Q. Li, T. Hirai, S. Katsuragawa, F. Li, and K. Doi, An Accurate Segmentation Method for Volumetry of Brain Tumor in 3D MRI, Proc. of SPIE Vol. 6914 69144D-1, 2008.
- [15] S. Dube, J. J. Corso, A. Yuille, T. F. Cloughesy, S. El-Saden, and U. Sinha, Hierarchical Segmentation of Malignant Gliomas via Integrated Contextual Filter Response, Proc. of SPIE Vol. 6914 69143Y-1, 2008.
- [16] J. J. Corso, E. Sharon, S. Dube, S. El-Saden, U. Sinha, and A. Yuille, Efficient Multilevel Brain Tumor Segmentation With Integrated Bayesian Model Classification, IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. 27, NO. 5, MAY 2008.
- [17] C. H. Lee, M. Schmidt, A. Murtha, A. Bistriz, J. Sander, and R. Greiner, Segmenting brain tumor with conditional random fields and support vector machines, in Proc. Workshop Comput. Vision Biomed. Image Appl. Int. Conf. Comput. Vision, pp. 469–478, 2005.
- [18] C. H. Lee, S. Wang, A. Murtha, M. R.G. Brown, and R. Greiner, Segmenting Brain Tumors Using Pseudo-Conditional Random Fields, MICCAI, Part I, LNCS 5241, pp. 359–366, 2008.
- [19] M. Wels, G. Carneiro, A. Aplas, M. Huber, J. Hornegger, and D. Comaniciu, A Discriminative Model-Constrained Graph Cuts Approach to Fully Automated Pediatric Brain Tumor Segmentation in 3-D MRI, MICCAI 2008, Part I, LNCS 5241, pp. 67–75, 2008.
- [20] W. E. Phillips, R. P. Velthuizen, S. Phupanich, L. O. Hall, L. P. Clarke, and M. L. Silbiger, Application of fuzzy c-means segmentation technique for tissue differentiation in MR images of a hemorrhagic glioblastoma multiforme, Magn. Reson. Imag., vol. 13, no. 2, pp. 277–290, 1995.
- [21] N. B. Karayiannis and P. I. Pai, Segmentation of magnetic resonance images using fuzzy algorithms for learning vector quantization, IEEE Trans. Med. Imag., vol. 18, no. 2, pp. 172–180, Feb. 1999.
- [22] S. Ho, E. Bullitt, and G. Gerig, Level set evolution with region competition: Automatic 3-D segmentation of brain tumors, in Proc. Int. Conf. Pattern Recognit., vol. I, pp. 532–535, 2002.
- [23] N. Ray, B. Saha, and M. Brown, Locating brain tumor from MR imagery using symmetry, Asilomar conf. on signals, systems, and computers, 2007.
- [24] <http://www.spl.harvard.edu/publications/item/view/1180>, retrieved in March 2009.
- [25] P. L. Bazin, D. L. Pham, W. Gandler, and M. McAuliffe, Free Software Tools for Atlas-based Volumetric Neuroimage Analysis, Progress in Biomedical Optics and Imaging - Proceedings of SPIE 5747 (III), art. no. 212, pp. 1824-1833.
- [26] <http://mipav.cit.nih.gov/>, retrieved in April 2009.
- [27] http://www.loni.ucla.edu/ICBM/Downloads/Downloads_452T1.shtml, retrieved in March 2009.
- [28] http://www.loni.ucla.edu/ICBM/Downloads/Downloads_ICBMprobabilistic.shtml, retrieved in March 2009.
- [29] Tao Wang, Irene Cheng and Anup Basu, Fluid Vector Flow and Applications in Brain Tumor Segmentation, IEEE Transactions on Biomedical Engineering, Vol. 56(3), pages 781-789, 2009.
- [30] J. Talairach and P. Tournoux, *Co-Planar Stereotaxic Atlas of the Human Brain*, Thieme, 1988.
- [31] J. Lancaster, J. Summerlin, L. Rainey, C. Freitas, and P. Fox, The talairach daemon, a database server for talairach atlas labels, *Neuroimage* 5(4), 1997.
- [32] W. L. Nowinski and D. Belov, The cerefy neuroradiology atlas: a talairach-tournoux atlas-based tool for analysis of neuroimages available over the internet, *NeuroImage* 20, pp. 50–57, 2003.
- [33] <http://www.nitrc.org/>, retrieved in April 2009.

- [34] <http://www.slicer.org>, retrieved in March 2009.
- [35] <http://www.fmrib.ox.ac.uk/fsl>, retrieved in March 2009.
- [36] <http://crl.med.harvard.edu/software/STAPLE>, retrieved in March 2009.
- [37] <http://bioimagesuite.org>, retrieved in March 2009.
- [38] www.itksnap.org, retrieved in March 2009.
- [39] <http://www-sop.inria.fr/asclepios>, retrieved in March 2009.
- [40] <http://www.loni.ucla.edu>, retrieved in March 2009.
- [41] M. Chen, T. Kanade, D. Pomerleau, J. Schneider, 3-D Deformable Registration of Medical Images Using a Statistical Atlas, MICCAI, LNCS 1679, pp. 621-630, 1999.
- [42] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum Likelihood from Incomplete Data via the EM Algorithm, *Journal of the Royal Statistical Society. Series B (Methodological)* 39 (1): 1–38, 1977.
- [43] R. Neal, G. E. Hinton, A view of the EM algorithm that justifies incremental, sparse, and other variants. *Learning in Graphical Models* (Cambridge, MA: MIT Press): 355–368. ISBN 0262600323, 1999.
- [44] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning*. New York: Springer. pp. 236–243. ISBN 0-387-95284-5, 2001.
- [45] M. Jamshidian, R. I. Jennrich, Acceleration of the EM Algorithm by using Quasi-Newton Methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 59 (2): 569–587, 1997.
- [46] J. B. MacQueen, Some Methods for classification and Analysis of Multivariate Observations, in *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*. 1: 281–297, University of California Press, 1967.
- [47] P. Brucker, On the complexity of clustering problems, *Optimization and operations research*, pp. 45–54, *Lecture Notes in Economics and Mathematical Systems* 157, Berlin: Springer, 1978.
- [48] P. Drineas, A. Frieze, R. Kannan, S. Vempala, V. Vinay, Clustering large graphs via the singular value decomposition, *Machine Learning* 56 (1): 9–33, 2004.
- [49] M. Kass, A. Witkin, and D. Terzopoulos, Snakes: Active contour model, *Intl. J. of Computer Vision*, vol. 1(4), pp. 321-331, 1988.
- [50] R. Whitaker. A level-set approach to 3D reconstruction from range data. *Int. J. of Comp. Vision*, Vol. 29(3), pp. 203–231, 1998.
- [51] S. Osher and J. Sethian, Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, Vol 79(1), pp. 12–49, 1988.
- [52] D. Adalstein, and J.A. Sethian, A fast level set method for propagating interfaces. *Journal of Computational Physics*, Vol 118(2), pp. 269–277, 1995.
- [53] J.A. Sethian, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Material Sciences*, Cambridge University Press, 1996.
- [54] J. C. Masse and J. F. Plante, A Monte Carlo study of the accuracy and robustness of ten bivariate location estimators, *Comput. Statistics & Data Analysis* vol. 42, pp. 1-26, 2003.
- [55] T. Wang and I. Cheng, Generation of Unit-width curve skeletons based on Valence Driven Spatial Median (VDSM), *Int. Sym. on Visual Comput.*, LNCS 5358, pp. 1061-1070, 2008.
- [56] T. Chan and L. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, February 2001.
- [57] C. Xu and J. L. Prince, Snakes, shapes, and gradient vector flow, *IEEE Trans. on Image Processing*, pp. 359-369, 1998.

[58] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, USA: Academic Press, p. 366, 1999.

Appendix

Table A. Comparison of brain tumor segmentation methods. NA means not available or not applicable. The 1st column shows the method names: Hopfield Neural Network (HNN), Knowledge Based (KB), Support Vector Machine (SVM), Fuzzy connectness (FC), Variational Method (VM), Spiral Scanning (SS), Segmentation by Weighted Aggregation (SWA), Conditional Random Fields (CRF), Fuzzy C-means Segmentation (FCM), Fuzzy Algorithms for Learning Vector Quantization (FALVQ), K Nearest Neighbour (KNN), Level Set (LS), Expectation Maximization (EM), Symmetry Based (SB), and Graph Cuts (GC). The 2nd column lists the two categories (CA) of the methods: training based (TB) or non-training based (NB). The 3rd column reveals the availability of the database (DB): private (PR) or public (PU). The 4th column indicates automatic (AU) status of the method: fully (F) automatic or semi (S) automatic. The 5th column displays the dimensionality (DI) of the method: 2D or 3D. The 6th column demonstrates the modality of the MRI: T1 weighted (T1), T2 weighted (T2), T1 with contrast enhancement (T1c), or Proton Density weighted (PD). The 7th column shows the tumor types: glioblastoma multiforme (GBM), astrocytoma (AA), low grade glioma (GA), meningioma (MA). The 8th column shows the number of training cases (TR #). The 9th column compares the number of test cases (TE #). The 10th column displays the executing time. Note that the methods were implemented with various programming languages on a variety of platforms. The executing times may not be compared directly. The 11th column lists the accuracy. Note that some papers used Jaccard coefficient while other papers used Tanimoto metric or other metrics. The accuracies may not be compared directly.

Method	CA	DB	AU	DI	Modality	Tumor	TR #	TE #	Time	Accuracy
HNN[3]	TB	PR	F	2D	NA	NA	NA	2	NA	NA
KB[4]	TB	PR	F	3D	T1,T2,PD	GBM	3	7	NA	.70
KNN[7]	TB	PR	NA	2D	T1,T1c,T2,PD	GA	1	9	1-2m	NA
ATM[8,9]	TB	PR	F	3D	NA	MA,GA	NA	20	10m	.99
EM[10,11]	TB	PR	F	3D	T1,T1c,T2	GBM,MA	1	5	1h4m	.49-.94
SVM[12]	TB	PR	NA	2D	NA	NA	NA	11	6.85s	.86-.94
FC[13]	TB	PR	S	3D	T1,T1c,FLAIR	GBM	NA	10	16m	.99
VM[2]	TB	PR	F	3D	T1,T2	AA,GBM	8	1	NA	NA
SS [14]	TB	PR	F	2D	T1	NA	NA	16	21s	.66-.99
SWA[15,16]	TB	PR	F	3D	T1,T1c,T2,FLAIR	GBM	10	10	7m	.27-.88
CRF[17,18]	TB	PR	F	2D	T1,T2,T1c	AA,GBM	11	11	38s	.41-.90
GC[19]	TB	PR	F	3D	T1,T1c,T2	AA	NA	6	5m	.78±.17
FCM[20]	NB	PR	S	3D	T1,T2	GBM	0	1	NA	NA
FALVQ[21]	NB	PR	NA	2D	T1,T2	MA	0	1	NA	NA
LS[22]	NB	PR	F	3D	T1,T1c	GBM	0	3	NA	.85-.93
SB[23]	NB	PR	F	2D	T1,T1c,T2,FLAIR	NA	0	14	NA	.40-.71
Proposed	TB	PU	F	3D	T1	MA,GA,AA	1	10		.22-.88