Proceedings of the 2012 IEEE 12th International Conference on Bioinformatics & Bioengineering (BIBE), Larnaca, Cyprus, 11-13 November 2012

Automated Tumor Segmentation using Kernel Sparse Representations

Jayaraman J. Thiagarajan, Deepta Rajan, Karthikeyan Natesan Ramamurthy,
David Frakes and Andreas Spanias
SenSIP Center and Industry Consortium, School of ECEE, Arizona State University, Tempe, AZ, USA 85287.

Abstract— **In this paper, we describe a pixel based approach for automated segmentation of tumor components from MR images. Sparse coding with data-adapted dictionaries has been successfully employed in several image recovery and vision problems. Since it is trivial to obtain sparse codes for pixel values, we propose to consider their non-linear similarities to perform kernel sparse coding in a high dimensional feature space. We develop the kernel K-lines clustering procedure for inferring kernel dictionaries and use the kernel sparse codes to determine if a pixel belongs to a tumorous region. By incorporating spatial locality information of the pixels, contiguous tumor regions can be efficiently identified. A low complexity segmentation approach, which allows the user to initialize the tumor region, is also presented. Results show that both of the proposed approaches lead to accurate tumor identification with a low false positive rate, when compared to manual segmentation by an expert.**

Index Terms—**MRI, tumor segmentation, sparse representations, kernel methods**

I. Introduction

A robust method to automatically segment a medical image into its constituent heterogeneous regions can be an extremely valuable tool for clinical diagnosis and disease modeling. Given a reasonably large data set, performing manual segmentation is not a practical approach. Brain tumor detection and segmentation have been of interest to researchers over the recent years and currently, there exists no comprehensive algorithm built and adopted in the clinical setting [1]. Although patient scans can be obtained using different imaging modalities, Magnetic Resonance Imaging (MRI) has been commonly adopted for brain imaging over other modalities because of its non-invasive and non-ionizing nature, and ability for direct multi-plane imaging.

Tumors may be malignant or benign as determined by a biopsy, and are known to affect brain symmetry and cause damages to the surrounding brain tissues. Automated tumor segmentation approaches are often challenged by the variability in size, shape and location of the tumor, the high degree of similarity in the pixel intensities between normal and abnormal brain tissue regions, and the intensity variations of identical tissues across volumes. As a result, unsupervised thresholding techniques have not been very successful in accurate tumor segmentation [2]. Furthermore, approaches that incorporate prior knowledge of the normal brain from atlases require accurate non-rigid registration [3], [4], and hence generating adequate segmentation results potentially calls for user-intervention and/or a patient specific training system. In addition, these methods require elaborate pre-processing and they tend to over-estimate the tumor volume.

Approaches for tumor segmentation can be either region-based or pixel based. The active contours method [5] is a widely adopted region-based approach that is usually combined with a level-set evolution for convergence to a region of interest [6]. However, it is sensitive to the contour initialization, and has a high computational cost due to its iterative nature. Model-based approaches [7] employ geometric priors to extend the Expectation Maximization (EM) algorithm to augment statistical classification. In relatively homogeneous cases such as low grade gliomas, the outlier detection framework proposed by Prastawa *et al.* [2], [8] was shown to perform well.

Pixel based approaches such as Fuzzy C-Means (FCM) using neighborhood labels [9], Conditional Random Fields [10], Bayesian model-aware affinities extending the SWA algorithm [1], and the more recent graph-based techniques combined with Cellular-Automata (CA) algorithm [11] have also achieved some success in tumor segmentation. However, processing issues with respect to contour initialization, noise reduction, intensity standardization, cluster selection, spatial registration, and the need for accurate manual seed-selection leaves substantial room for improvement. In addition, building a robust automated approach that does not require user intervention is very important, particularly for processing large datasets.

In this paper, we propose a novel pixel based segmentation technique to automatically segment enhancing/active and necrotic tumor components from T1-weighted contrast-enhanced MR images. The success of sparse coding and dictionary learning in several image reconstruction applications has motivated us to apply them in the tumor segmentation problem. Sparse coding is typically performed on image patches or feature vectors, and it is trivial to obtain codes for pixel intensities, since they are one-dimensional. Hence we propose to perform kernel sparse coding, an approach where the codes for pixels are obtained using non-linear similarities measured between them in a high dimensional feature space. Furthermore, we develop the kernel K-lines clustering algorithm to learn kernel dictionaries for coding the pixels. We also describe two different approaches for incorporating spatial locality information of the pixels to localize the active tumor regions. Finally, we present a semi-automated segmentation technique for improved computational efficiency, wherein the

978-1-4673-4358-9/12/$31.00 ©2012 IEEE

user can initialize the tumor region. We evaluate the proposed algorithm on a set of T1-weighted contrast-enhanced MR images and compare the results with manual segmentations performed by an expert radiologist. We demonstrate that the proposed algorithm achieves high detection accuracy at a low computational cost.

II. Sparse Coding and Dictionary Learning

Sparse models have become a significant paradigm in image understanding, since many naturally occurring signals and images can be expressed as a sparse linear combination of a few elementary data samples [12]. The elementary data samples, also known as *atoms*, are usually normalized to unit ℓ_2 norm, and stacked together in a matrix referred to as the dictionary. In addition to being useful in data representation problems, there has been a recent surge of interest in using sparse models in several supervised, semi-supervised and unsupervised learning tasks such as clustering [13] and classification [14]. Given a sample $\mathbf{y} \in \mathbb{R}^M$, and a dictionary $\mathbf{D} \in \mathbb{R}^{M \times K}$, the generative model for sparse coding is $\mathbf{y} = \mathbf{D}\mathbf{x} + \mathbf{n}$, where $\mathbf{x} \in \mathbb{R}^K$ is the sparse code with a small number of non-zero coefficients and $\mathbf{n}$ is the noise component. The sparse code can be computed by solving the convex problem

$$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \beta\|\mathbf{x}\|_1, \tag{1}$$

where $\|.\|_1$ indicates the ℓ_1 norm, and is a convex surrogate for the ℓ_0 norm which counts the number of non-zero elements in a vector.

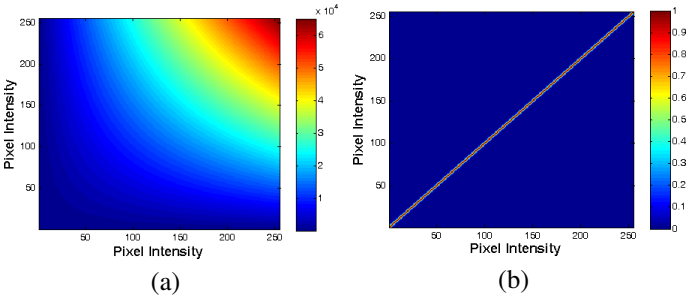
When presented with a sufficiently large set of training data samples, $\mathbf{Y} = [\mathbf{y}_i]_{i=1}^T$, the dictionary can be learned, and the sparse codes can be obtained by solving

$$\min_{\mathbf{D},\mathbf{X}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 + \beta \sum_{i=1}^{T} \|\mathbf{x}_i\|_1, \tag{2}$$

where $\mathbf{X} = [\mathbf{x}_i]_{i=1}^T$, and $\|.\|_F$ denotes the Frobenius norm of the matrix. This can be solved as an alternating minimization problem, where the dictionary is learned fixing the sparse codes, and the sparse codes are obtained fixing the dictionary.

III. Kernel Sparse Coding for Tumor Segmentation

Sparse coding algorithms are typically employed for vectorized patches extracted from the images, using an overcomplete dictionary. However, in the proposed tumor segmentation problem, we need to obtain sparse codes for representing each pixel in the image. This is trivial if we use the approach specified in (1), since $M = 1$ in this case. Furthermore, in order to discriminate between the pixels belonging to multiple segments, we may need to consider the *non-linear similarity* between them. These problems can be addressed by projecting the pixel intensities on to a feature space $\mathcal{F}$, using an appropriate non-linear transformation $\Phi : \mathbb{R} \rightarrow \mathcal{F}$, and then considering their linear similarity in the space $\mathcal{F}$. The sparse codes obtained in $\mathcal{F}$ will be discriminatory since they reflect the non-linear similarity and hence segmentation can be performed using these codes. Note that the choice of the non-linear transformation is crucial to ensure discrimination.



(a) (b)

Fig. 1. Similarity between grayscale pixel intensities (0 to 255) (a) Linear similarity ($y_i y_j$), (b) Non-linear similarity ($\mathcal{K}(y_i, y_j)$) using an RBF kernel.

Assume that the T pixels in an image are represented by $\mathbf{Y} = [y_i]_{i=1}^T$. The transformation $\Phi(.)$ is chosen such that $\mathcal{F}$ is a Hilbert space with the reproducing kernel $\mathcal{K}(.,.)$ and hence the non-linear similarity between two samples in $\mathcal{F}$ can be measured as $\mathcal{K}(y_i, y_j) = \Phi(y_i)^T\Phi(y_j)$. Note that the feature space is usually high dimensional (sometimes infinite) and the closed form expression for the transformation $\Phi(.)$ may be intractable or unknown. Therefore, we simplify the computations by expressing them in terms of inner products $\Phi(y_i)^T\Phi(y_j)$, which can be replaced using $\mathcal{K}(y_i, y_j)$ whose value is always known. This is referred to as the *Kernel Trick*. Note that in order for a kernel to be valid, the kernel function or the kernel matrix should be symmetric positive semidefinite according to Mercer's theorem [15]. In this work, we use the Radial Basis Function (RBF) kernel of the form $\mathcal{K}(y_i, y_j) = \exp(-\gamma(y_i - y_j)^2)$, which leads to discriminative sparse codes. The difference between linear similarity of grayscale pixel intensities (0 to 255) and the non-linear similarities obtained using the RBF kernel ($\gamma = 0.3$) are illustrated in Figure 1(a) and (b). The linear similarities depend predominantly on the individual intensities of the pixels and not on the closeness of intensities. Whereas, when the RBF kernel is used, the pixel intensities that are close to each other have high non-linear similarity irrespective of the intensities. Pixels with intensities that are far apart have zero non-linear similarity. Therefore, the pixelwise sparse codes that we obtain using such a kernel will also follow a similar behavior, i.e., pixels with intensities close to each other will have similar sparse codes.

A. Kernel Sparse Coding

The generative model in $\mathcal{F}$ for kernel sparse coding is given by $\Phi(\mathbf{y}) = \Phi(\mathbf{D})\mathbf{x} + \mathbf{n}$. The feature space (kernel) dictionary, $\Phi(\mathbf{D}) = [\Phi(\mathbf{d}_i)]_{i=1}^K$, can be written as $\Phi(\mathbf{Y})\mathbf{A}$, since any vector in $\mathcal{F}$ lies in the linear span of $\Phi(\mathbf{Y})$. Here, $\Phi(\mathbf{Y}) = [\Phi(y_i)]_{i=1}^T$ and $\mathbf{A} \in \mathbb{R}^{T \times K}$. The kernel sparse codes for any pixel y can be now obtained by solving [16]

$$\min_{\mathbf{x}} \|\Phi(y) - \Phi(\mathbf{Y})\mathbf{A}\mathbf{x}\|_2^2 + \beta\|\mathbf{x}\|_1. \tag{3}$$

We will repose the problem as follows since $\Phi(.)$ is unknown:

$$\min_{\mathbf{x}} \mathbf{x}^T\mathbf{A}^T\mathbf{K}_{\mathbf{YY}}\mathbf{A}\mathbf{x} - 2\mathbf{K}_{y\mathbf{Y}}\mathbf{A}\mathbf{x} + \beta\|\mathbf{x}\|_1, \tag{4}$$

where the matrix $\mathbf{K}_{\mathbf{YY}} = \Phi(\mathbf{Y})^T\Phi(\mathbf{Y})$ and $\mathbf{K}_{y\mathbf{Y}} = \Phi(y)^T\Phi(\mathbf{Y})$. Note that (3) has a quadratic term with an ℓ_1 penalty and hence can be solved using a reasonably fast solver such as the feature sign search algorithm [17].

B. Dictionary Design

In order to design the dictionary $\Phi(\mathbf{D})$, we will adapt (2) to the feature space, with the constraint that only one element in the sparse code can be non-zero. This is a special case of the kernel dictionary learning proposed in [18]. This procedure is equivalent to the kernel version of K-lines clustering, which attempts to fit K 1-D subspaces to the training data in $\mathcal{F}$ [19]. Learning a dictionary using this simple clustering scheme results in a good performance for our tumor segmentation problem. The clustering procedure can be solved using

$$\min_{\mathbf{A},\mathbf{X}} \|\Phi(\mathbf{Y}) - \Phi(\mathbf{Y})\mathbf{A}\mathbf{X}\|_F^2 \text{ such that } \|\mathbf{x}_i\|_0 \leq 1, \forall i. \quad (5)$$

Each dictionary atom $\Phi(\mathbf{d}_i)$ corresponds to a cluster center and each coefficient vector $\mathbf{x}_i$ encodes the cluster association as well as the weight corresponding to the i^{th} pixel. Let us define K membership sets $\{\mathcal{C}_k\}_{k=1}^K$, where $\mathcal{C}_k$ contains the indices of all training vectors that belong to the cluster k. The alternating optimization for solving (5) consists of two steps: (a) cluster assignment, which involves finding the association and weight of each training vector and hence updating the sets $\{\mathcal{C}_k\}_{k=1}^K$, and (b) cluster update, which involves updating the cluster center by finding the centroid of training vectors corresponding to each set $\mathcal{C}_k$.

In cluster assignment, we compute the correlations of a training sample, with the dictionary atoms as $\Phi(y_i)^T\Phi(\mathbf{D}) = \mathbf{K}_{y_i\mathbf{Y}}\mathbf{A}$. If the k^{th} dictionary atom resulted in maximum absolute correlation, the index i is placed in set $\mathcal{C}_k$, and the corresponding non-zero coefficient is the correlation value itself. For the cluster k, let $\Phi(\mathbf{Y}_k) = \Phi(\mathbf{Y})\mathbf{E}_k$ be the set of member vectors and $\mathbf{x}_k^R$ be the row of corresponding non-zero weights. The cluster update involves solving

$$\min_{\mathbf{a}_k} \|\Phi(\mathbf{Y})\mathbf{a}_k\mathbf{x}_k^R - \Phi(\mathbf{Y})\mathbf{E}_k\|_F^2. \quad (6)$$

Denoting the singular value decomposition of

$$\Phi(\mathbf{Y}_k) = \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^T, \quad (7)$$

the rank-1 approximation, which also results in the optimal solution for (6), is given by

$$\Phi(\mathbf{Y})\mathbf{a}_k\mathbf{x}_k^R = \mathbf{u}_{k1}\sigma_{k1}\mathbf{v}_{k1}^T, \quad (8)$$

where σ_{k1} is the largest singular value, and $\mathbf{u}_{k1}$ and $\mathbf{v}_{k1}$ are the columns of $\mathbf{U}_k$ and $\mathbf{V}_k$ corresponding to that singular value. Eqn. (8) implies that $\Phi(\mathbf{Y})\mathbf{a}_k = \mathbf{u}_{k1}$ and $\mathbf{x}_k^R = \sigma_{k1}\mathbf{v}_{k1}^T$. Let the eigen decomposition of $\mathbf{K}_{\mathbf{Y}_k\mathbf{Y}_k}$ be $\mathbf{V}_k\mathbf{\Delta}_k\mathbf{V}_k^T$ and hence we have $\sigma_{k1} = \sqrt{\Delta_k(1,1)}$, assuming the eigen values are in descending order. From (7), we also have $\Phi(\mathbf{Y}_k)\mathbf{v}_{k1} = \sigma_{k1}\mathbf{u}_{k1}$. Substituting for $\Phi(\mathbf{Y}_k)$ and $\mathbf{u}_{k1}$, we obtain $\Phi(\mathbf{Y})\mathbf{E}_k\mathbf{v}_{k1} = \sigma_{k1}\Phi(\mathbf{Y})\mathbf{a}_k$, and this results in

$$\mathbf{a}_k = \sigma_{k1}^{-1}\mathbf{E}_k\mathbf{v}_{k1}. \quad (9)$$

Note that $\mathbf{a}_k$ completely defines $\mathbf{d}_k$. The cluster assignment and update steps are repeated until convergence, which is when $\{\mathcal{C}_k\}_{k=1}^K$ does not change over iterations.

IV. Proposed Algorithm

In this section, we describe the proposed algorithm for automated tumor segmentation based on kernel sparse codes. Furthermore, we present a simplified approach for complexity reduction, by allowing the user to initialize the tumor region.

A. Automated Tumor Segmentation

As described earlier, the proposed algorithm employs a pixel based approach to determine tumor regions in the MR image. In order to determine if a pixel belongs to a tumor region, adaptive thresholding techniques can be used. However, building more sophisticated tools can improve the segmentation performance. In this paper, we propose to use the kernel sparse codes to identify pixels in the active tumor regions. This is carried out by designing two separate kernel dictionaries for modeling tumor and non-tumor pixels respectively. Given a set of training images, containing active tumor regions, the users manually mark tumor and non-tumor regions in the images. Using the selected regions, we construct two RBF kernel matrices as described in Section III. Next, we employ the kernel K-lines clustering algorithm proposed in Section III-B to learn the tumor and non-tumor dictionaries, $\Phi(\mathbf{D}_T)$ and $\Phi(\mathbf{D}_N)$, respectively. Note that dictionary learning is performed only once, and as we will show in our experiment results, the dictionaries generalize well to a reasonably large dataset.

For a test image, we obtain kernel sparse codes for each pixel y_i using $\Phi(\mathbf{D}_T)$ and $\Phi(\mathbf{D}_N)$, and denote the respective sparse codes as $\mathbf{x}_i^T$ and $\mathbf{x}_i^N$. Since the dictionaries are optimized for two different types of regions, we expect the tumor pixels to be better modeled by the tumor dictionary. Hence we classify a pixel as belonging to an active tumor region if the approximation error obtained with the tumor dictionary is less than that obtained with the non-tumor dictionary:

$$\mathcal{J}(y_i) = \begin{cases} \textit{Tumor}, & \text{if } E_N - E_T \geq \epsilon, \\ \textit{Non-tumor}, & \text{otherwise.} \end{cases} \quad (10)$$

Here the approximation errors with respect to the two dictionaries are $E_N = \|\Phi(y_i) - \Phi(\mathbf{D}_N)\mathbf{x}_i^N\|_2$ and $E_T = \|\Phi(y_i) - \Phi(\mathbf{D}_T)\mathbf{x}_i^T\|_2$, respectively. Note that the threshold for the error difference, ϵ, can be tuned using a validation dataset before applying the algorithm to the test data.

In order to localize the tumor regions in the image, pixel based segmentation approaches typically incorporate constraints to ensure connectedness among pixels in a segment. In the proposed algorithm, spatial locality information can be included in two different ways. The first approach is to employ spectral clustering methods [20], which exploit the spatial locality information of the pixels by constructing a graph on the tumor pixels as determined by (10). An alternate approach is to include the spatial locality information as part of the kernel matrix itself, before obtaining the kernel sparse

codes. In addition to allowing the design of non-linear models, kernel methods provide an elegant way to combine multiple features. We can construct ensemble kernels by fusing kernel matrices corresponding to different features. In our approach, we combine the RBF kernel matrices computed for pixel intensities and locations as follows

$$\mathbf{K} = \mathbf{K}_i \odot \mathbf{K}_l, \tag{11}$$

where $\mathbf{K}_i$ and $\mathbf{K}_l$ are the two kernel matrices. The spatial locality kernel $\mathbf{K}_l$ is constructed as

$$\mathbf{K}(i,j) = \mathcal{K}(y_i, y_j) = \begin{cases} \exp^{\|\mathbf{l}_i - \mathbf{l}_j\|_2^2}, & \text{if } \; j \in \mathcal{N}(i), \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

Here, $\mathcal{N}(i)$ denotes the neighborhood of the pixel y_i, and $\mathbf{l}_i$ and $\mathbf{l}_j$ are the locations of the pixels, y_i and y_j, respectively. Note that when combining kernel matrices we need to ensure that the resulting kernel matrix also satisfies the Mercer's conditions. Sparse codes obtained with a dictionary learned in the ensemble feature space model the similarities with respect to both intensity and location of pixels. Instead of employing a simple approximation error-based classifier in this case, we use the kernel codes to train a linear SVM classifier to determine if a pixel belongs to a tumor region. Though these two approaches introduce spatial locality at different stages of the algorithm, and the latter approach uses a more sophisticated classifier, we observed in our experiments that the two methods perform similarly.

B. Complexity Reduction using a Semi-Automated Approach

The amount of training required and the computational complexity are two important factors that determine the efficiency of an automated segmentation algorithm. Since the dictionary training is performed using pixels, the number of training images used is quite limited. Details on the training setup can be found in Section V. Though the computational complexity of the automated segmentation algorithm described earlier is comparable to several existing methods, its efficiency can be further improved by allowing the user to initialize the tumor region. Computing the kernel sparse codes for all pixels in a test image incurs the maximum complexity and hence initializing the tumor regions reduces the number of pixels to be processed drastically. Furthermore, there is no need to explicitly include the location information in the algorithm, since the tumor region has already been localized by the user. Hence, the segmentation is carried out by using the simple error-based classifier on the kernel sparse codes. We refer to this as the *semi-automated* segmentation approach. We observed from our experiments that for an average sized tumor region, we achieve significant speedup by using the semi-automated approach. However, the segmentation obtained using the two methods are quite comparable, though the automated approach can potentially generate more false positives when compared to the semi-automated approach.

V. Experiments

In this section, we provide details about the dataset used to evaluate our algorithm and present the segmentation results. The results are compared to manual segmentations performed by a radio-oncology specialist, both visually as well as using standard metrics such as *Accuracy (Acc)* and *Correspondence Ratio (CR)*.

A. Dataset

The algorithm was tested on a set of T1-weighted (spin echo) contrast-enhanced, 2-D Dicom images acquired with a 1.5T GE Genesis Signa MR scanner. Each axial slice is 5 mm thick with a 7.5 mm gap between slices, and the size of the image matrix is 256×256 pixels. Patients were administered 20cc Bolus of Gadolinum contrast agent, and were already diagnosed with Glioblastoma Multiforme (GBM), the most common and dangerous malignant primary brain tumor. These tumors are characterized by jagged boundaries with a ring enhancement, possibly a dark core necrotic component, and are accompanied by edema (swelling). The ground truth (GT) images were obtained from the manual segmentation carried out by an expert radiologist at the St. Joseph's Medical Hospital in Phoenix. We tested our algorithm on the pre- and post-treatment images (approximately 25 per patient) for 6 patients where all the slices (approximately 150) showed the presence of GBM.

B. Results

Simulations were carried out independently for both the semi-automated and automated algorithms for every axial slice. For the RBF kernel we set $\gamma = 0.3$, and the tumor dictionary with 128 atoms was learned using $10,000$ randomly chosen pixels obtained from the tumor regions in 4 images. The non-tumor dictionary was learned similarly. Given an input image, the kernel sparse codes for each pixel in the image were obtained using both of these dictionaries. The parameter $\beta = 0.1$ was used for sparse coding using the feature sign search algorithm.

The resulting segmented images were compared to the GT and performance was measured using the metrics *Accuracy (Acc)* and *Correspondence Ratio (CR)* computed as [4]

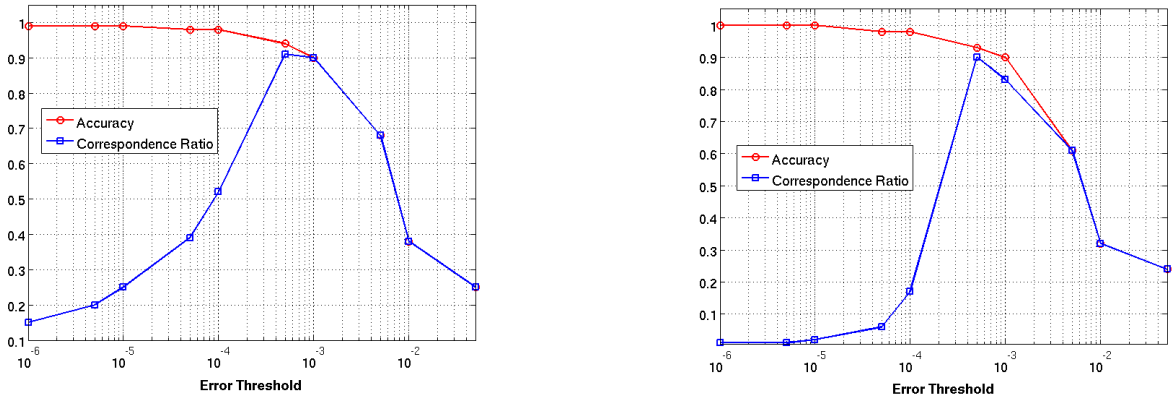
$$\text{Acc} = \frac{\text{TP}}{\text{Total \# tumor pixels in the GT}}, \tag{13}$$

and

$$\text{CR} = \frac{\text{TP} - 0.5\text{FP}}{\text{Total \# tumor pixels in the GT}}, \tag{14}$$

where TP indicates the number of true positives (the pixels indicated as tumorous by the ground truth and our algorithm), and FP denotes the number of false positives (pixels indicated as non-tumorous by the ground truth but tumorous by our algorithm).

Fig. 2 shows the relationship between Acc and CR vs the error threshold (ϵ) for one example input image. It was observed that as the value of ϵ increased, accuracy decreased



(a) (b)

Fig. 2. Plots depicting the relationship of Accuracy and Correspondence Ratio with respect to the error threshold for an example axial MR slice using: (a) semi-automated segmentation method and (b) automated segmentation method.

only slightly, while the correspondence ratio increased significantly, which indicates a reduction in FP. Any attempt to increase ϵ further drastically reduced both Acc and CR, owing to the reduction in the number of true positives. Hence, the ϵ value at which CR reaches its maxima can be considered to give the best segmentation result. Through the process of data validation, the range of thresholds which gives an optimized segmentation result was estimated.

Fig. 3 shows the original and segmented images, each row of which corresponds to a slice acquired at a different depth. Rows 1 and 2 correspond to pre-treatment images with the first row having two separate tumor masses, while the second row of images has a single large tumor mass with a more complex boundary. Rows 3 and 4 correspond to post-treatment images where a re-growth of the enhancing/active tumor region was observed. Both the semi-automated and automated segmentation methods detected these tumor regions with very high accuracy as demonstrated by the extensive results in Table 1. We observed that the performance of the automated algorithm is equivalent to that of the semi-automated algorithm in many cases and very closely comparable in the remaining cases. As expected, the semi-automated algorithm is significantly faster when compared to the automated approach.

TABLE I
PERFORMANCE OF THE PROPOSED APPROACHES.

Image	Fully Automated			Semi Automated		
	Acc(%)	CR	Time (s)	Acc(%)	CR	Time (s)
Slice 1	0.93	0.90	123.45	0.94	0.91	8.36
Slice 2	0.92	0.90	119.20	0.95	0.94	10.42
Slice 3	0.94	0.92	107.80	0.97	0.96	10.19
Slice 4	0.96	0.95	103.40	0.97	0.95	12.06
Slice 5	0.94	0.93	97.24	0.94	0.93	5.90
Slice 6	0.81	0.70	139.33	0.84	0.82	6.33
Slice 7	0.90	0.86	122.65	0.90	0.86	2.64
Slice 8	0.94	0.85	117.30	0.94	0.87	6.22
Slice 9	0.90	0.83	108.90	0.96	0.85	6.80
Slice 10	0.92	0.82	125.50	0.92	0.81	9.80
Slice 11	0.94	0.76	143.00	0.95	0.72	11.65
Slice 12	0.99	0.94	150.03	0.99	0.94	2.70
Slice 13	0.98	0.95	153.10	0.98	0.95	2.70
Slice 14	0.98	0.92	161.54	0.98	0.92	3.18
Slice 15	0.92	0.81	142.36	0.92	0.84	3.71

VI. DEVELOPMENT OF JAVA TOOLBOX

In this section, we briefly discuss the Java modules that we have developed for the implementation of the approaches presented in this paper. These Java modules have been integrated as a separate toolbox in the Java-DSP package. Java-DSP is an NSF sponsored online programming environment for signal/image processing education and research [21]. Though initially designed to enable students and distance learners to perform laboratories over the internet, more recently toolboxes that support multidisciplinary research have been developed. As part of this software package, we have developed a toolbox that includes Java modules for: (a) loading and viewing Dicom images, (b) sparse coding and dictionary learning, and (c) automated/semi-automated tumor segmentation. These modules were implemented using the JavaCV interface for the OpenCV package. This toolbox is currently under testing and will be disseminated shortly for public use.

VII. CONCLUSIONS

A novel, automated segmentation technique to detect brain tumors was proposed in this paper. The tumor and non-tumor regions were modeled using kernel dictionaries learned in a high dimensional feature space. Kernel sparse codes obtained for each pixel with both the dictionaries were used to detect the tumor regions. Two different methods for incorporating spatial locality information to perform tumor segmentation were discussed. A simplified semi-automated approach was also proposed and both the approaches resulted in accurate tumor identification with a low false positive rate. Future work involves extending the approach to include other types of MR imaging methods such as T2-weighted, FLAIR, Perfusion-weighted, and Diffusion-weighted images. Segmentation along

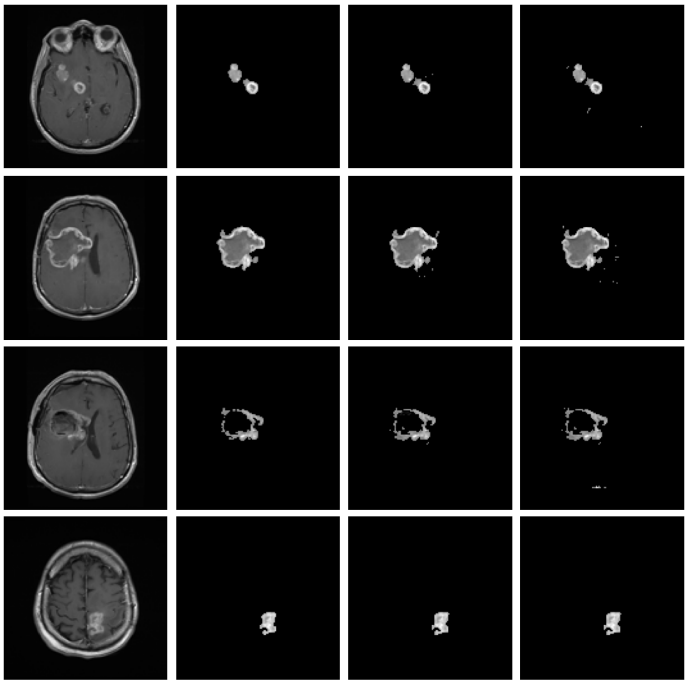


Fig. 3. (a) Original input MR image, (b) ground truth, (c) results from semi-automated segmentation and (d) results from automated segmentation.

with volumetric registration on different slices can be used to quantify the volume of the tumor region and model the growth of tumors over a period of time. The algorithm can also be extended to classify different types of tumors by building specific dictionaries for each type.

VIII. Acknowledgements

This work was supported in part by the NSF Net Centric I/UCRC - ASU SenSIP site award 1035086, the SenSIP Center and the Consortium award no. S0322. We thank Dr. Mark Preul and Milad Behbahaninia from Barrow Neurological Institute, and Anup Puri from Arizona State University for providing us with the tumor data.

References

[1] J. J. Corso *et. al.*, "Efficient multilevel brain tumor segmentation with integrated bayesian model classification," *Medical Imaging, IEEE Transactions on*, vol. 27, no. 5, pp. 629–640, 2008.

[2] M. Prastawa *et. al.*, "Automatic brain tumor segmentation by subject specific modification of atlas priors," *Academic Radiology*, vol. 10, no. 12, pp. 1341–1348, 2003.

[3] M.R. Kaus *et. al.*, "Automated segmentation of MR images of brain tumors," *Radiology*, vol. 218, no. 2, pp. 586–591, 2001.

[4] M. C. Clark and *et. al.*, "Automatic tumor segmentation using knowledge-based techniques," *Medical Imaging, IEEE Transactions on*, vol. 17, no. 2, pp. 187–201, 1998.

[5] T. Chan and L. Vese, "Active contours without edges," *Image Processing, IEEE Transactions on*, vol. 10, no. 2, pp. 266–277, 2001.

[6] S. Ho, E. Bullitt, and G. Gerig, "Level-set evolution with region competition: automatic 3-d segmentation of brain tumors," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 1. IEEE, 2002, pp. 532–535.

[7] N. Moon *et. al.*, "Model-based brain and tumor segmentation," in *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, vol. 1. IEEE, 2002, pp. 528–531.

[8] M. Prastawa *et. al.*, "A brain tumor segmentation framework based on outlier detection," *Medical Image Analysis*, vol. 8, no. 3, pp. 275–283, 2004.

[9] M.N. Ahmed *et. al.*, "A modified fuzzy c-means algorithm for bias field estimation and segmentation of MRI data," *Medical Imaging, IEEE Transactions on*, vol. 21, no. 3, pp. 193–199, 2002.

[10] C. H. Lee *et. al.*, "Segmenting brain tumors with conditional random fields and support vector machines," *Computer Vision for Biomedical Image Applications*, pp. 469–478, 2005.

[11] A. Hamamci *et. al.*, "Tumor-cut: Segmentation of brain tumors on contrast enhanced MR images for radiosurgery applications," *Medical Imaging, IEEE Transactions on*, no. 99, pp. 1–1, 2011.

[12] J. J. Thiagarajan, K. N. Ramamurthy, and A. Spanias, "Multilevel dictionary learning for sparse representation of images," in *Proceedings of the IEEE DSP Workshop, Sedona*, 2011.

[13] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *IEEE CVPR*, Jun. 2010, pp. 3501 –3508.

[14] J. Wright *et.al.*, "Robust face recognition via sparse representation," *IEEE Trans. on PAMI*, vol. 31, no. 2, pp. 210–227, 2001.

[15] N. Cristianini and J. Shawe-Taylor, *An introduction to support Vector Machines: and other kernel-based learning methods*. Cambridge University Press, 2000.

[16] S. Gao, I. Tsang, and L. Chia, "Kernel sparse representation for image classification and face recognition," *Computer Vision–ECCV 2010*, pp. 1–14, 2010.

[17] H. Lee *et. al.*, "Efficient sparse coding algorithms," *Advances in neural information processing systems*, vol. 19, p. 801, 2007.

[18] H. V. Nguyen *et. al.*, "Kernel dictionary learning," in *Proceedings of the IEEE ICASSP*, 2012.

[19] J. J. Thiagarajan, K. N. Ramamurthy, and A. Spanias, "Optimality and stability of the K-hyperline clustering algorithm," *Pattern Recognition Letters*, 2010.

[20] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 2, pp. 849–856, 2002.

[21] A. Spanias and V. Atti, "Interactive online undergraduate laboratories using J-DSP," *IEEE Transactions on Education*, vol. 48, no. 4, pp. 735–749, nov 2005.