Lung Segmentation Using Incremental Sparse NMF

Ehsan Hosseini-Asl[1*], Jacek M. Zurada[1,3], and Ayman El-Baz[2]

[1]Electrical and Computer Engineering Department, University of Louisville, Louisville, KY, USA
[2]Bioengineering Department,University of Louisville, Louisville, KY, USA
[3]Information Technology Institute, Academy of Management, 90-113 Łódz, Poland

1 Method

The traditional way to model the visual appearance of the image is to define the spatial interactions of the image voxels in terms of their neighboring voxels. A new spatial interaction model was developed for the 3D lung data by extracting new spatial features based on NMF.

Let $\mathbf{G}_{\mathbf{N}_{x,y,z}} \in \mathcal{Q}^{I_x \times I_y \times I_z}$ be the image signals of the neighborhood of the voxel (x, y, z). By including the image signals of the neighborhood of all voxels, a 4D matrix $\mathbf{G} \in \mathcal{Q}^{XYZ \times I_x \times I_y \times I_z}$ is composed. To process the 3D lung data using NMF, $\mathbf{G}_{\mathbf{N}_{x,y,z}}$ of each voxel is represented as a vector $\mathbf{g}_{\mathbf{N}_{x,y,z}}$ in the input data matrix $\mathbf{A} \in \mathcal{Q}^{I_x I_y I_z \times XYZ}$, as shown in Fig. 2. Using conventional NMF, $\mathbf{A}$ is decomposed as follow:

$$\mathbf{A} \approx \mathbf{WH} \tag{1}$$

where $\mathbf{W} \in \mathbb{R^+}^{I_x I_y I_z \times J}$ contains the basis of voxel's neighborhood as its columns, and the vectors of $\mathbf{H} \in \mathbb{R^+}^{J \times XYZ}$ represent the new feature encoding of voxels. In another sense, $\mathbf{W}$ contains the basis vectors of the new feature space, and the vectors of $\mathbf{H}$ represent the new features of the voxels that model the visual appearance of the image, where each value in the J-length $\mathbf{H}$ vector represents the belonging of a given voxel to the J image clusters.

2 Proposed Framework

In this section, a framework is proposed for 3D image segmentation, which consists of three steps as shown in Fig. 2. In the first step, the image volume is preprocessed to remove its background. Then, we develop an ISNMF-based approach to model the first- and the second-order visual appearance of the 3D image. Finally, the image is segmented using the developed NMF-based appearance model. The above steps are discussed in detail below.

* Corresponding Author:— Tel: (502) 852 3165, Fax: (502) 852 3940, E-mail: ehsan.hosseiniasl@louisville.edu.

2.1 Preprocessing

Due to the similarity between background and gray values of the region of interest, the first step of the proposed framework is to remove the background from the image volume using a 3D region growing method. An illustration for removing the background using this method is shown in Fig. 1

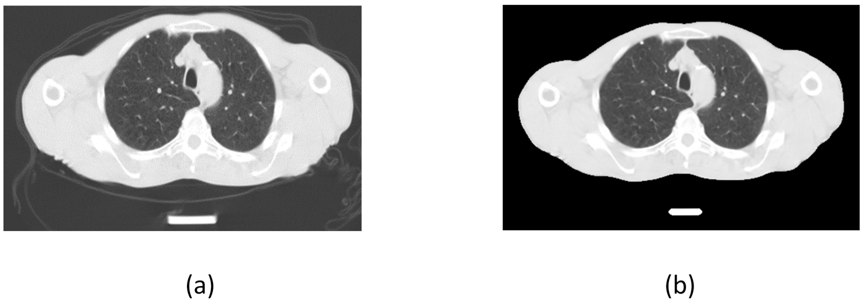

Fig. 1. (a) Original image, and (b) image after removing background using 3D region growing starting from a seed point at the corner of the image volume

2.2 ISNMF-based Visual Appearance Modeling

In this section, we describe a method to apply ISNMF for modeling the visual appearance of a 3D image. A conventional way to implement ISNMF is to incrementally add the neighborhood vector $\mathbf{g}_{\mathbf{N}_{x,y,z}}$ of each voxel and solve it using update rules given. Using this method, the $\mathbf{h}_k$ of each voxel is computed iteratively, while basis matrix $\mathbf{W}$ is updated as well. However, a 3D image contains too many voxels, and implementing ISNMF for all voxels (XYZ) is computationally expensive. To reduce the computational complexity of ISNMF, we propose that the voxels of each slice are added to matrix $\mathbf{A}$ at the same time, where each slice is processed sequentially. Using this method, the computed basis matrix $\mathbf{W}_k$ of the previous slice is used as the initialization of the update rule for $\mathbf{W}_{k+1}$ of the next slice.

Let $\mathbf{A}_{z+1}$ contains the neighborhood vectors $\mathbf{g}_{\mathbf{N}_{x,y,z}}$ of all voxels belonging to $(z+1)$-th slice. To compute the encoding matrix $\mathbf{H}_{z+1}$ of this slice and also to update $\mathbf{W}_{z+1}$, the following update rules are derived:

$$\mathbf{H}_{z+1} \leftarrow \mathbf{H}_{z+1} \frac{\mathbf{W}_{z+1}^T \mathbf{A}_{z+1}}{\mathbf{W}_{z+1}^T \mathbf{W}_{z+1} \mathbf{H}_{z+1} + \lambda_2 \mathbf{H}_{z+1}} \tag{2}$$

$$\mathbf{W}_{z+1} \leftarrow \mathbf{W}_{z+1} \frac{\mathbf{A}_{0:z} \mathbf{H}_{0:z}^T + \mathbf{A}_{z+1} \mathbf{H}_{z+1}^T}{\mathbf{W}_{z+1} (\mathbf{H}_{0:z} \mathbf{H}_{0:z}^T + \mathbf{H}_{z+1} \mathbf{H}_{z+1}^T) + \lambda_1 \mathbf{W}_{z+1}} \tag{3}$$

where $\mathbf{A}_{0:z}$ denotes the neighborhood vectors of slices 0 through z and $\mathbf{H}_{0:z}$ indicates the calculated encoding vectors of the corresponding slices. For computing $\mathbf{W}_{z+1}$ in Eq. (3), $\mathbf{W}_z$ calculated for z-th slice is used as the initialization.

Algorithm 1 ISNMF-based Visual Appearance Modeling

1: **Initialization**
 (i) Initialize $\mathbf{A}_0$ by the neighborhood vectors $\mathbf{g}_{\mathbf{N}_{x,y,z}}$ of the first slice ($x = 0,\ldots,X; y = 0,\ldots,Y; z = 0$)
 (ii) Set the number of image clusters (i.e. J)
 (iii) Initialize $\mathbf{H_0}$ and $\mathbf{W_0}$ randomly
 (iv) Alternatively update $\mathbf{H_0}$ and $\mathbf{W_0}$ using Eq. (**??**) and (**??**), respectively, until convergence criterion satisfied.

2: **Incremental Iterations:** For each slice $z = [1,\ldots,Z]$
 (i) Add the neighborhood vector of the voxels $\mathbf{g}_{\mathbf{N}_{x,y,z}}$ belonging to slice z to the data matrix, which compose $\mathbf{A}_z$.
 (ii) Initialize $\mathbf{H}_z$ randomely, and $\mathbf{W}_z = \mathbf{W}_{z-1}$
 (iii) Perform alternative update of $\mathbf{H}_z$ and $\mathbf{W}_z$ using Eq. (2) and (3), respectively, until convergence

3: **Output:** $\mathbf{H}_{0:Z}$ matrix that represents the new visual appearance of the image

---

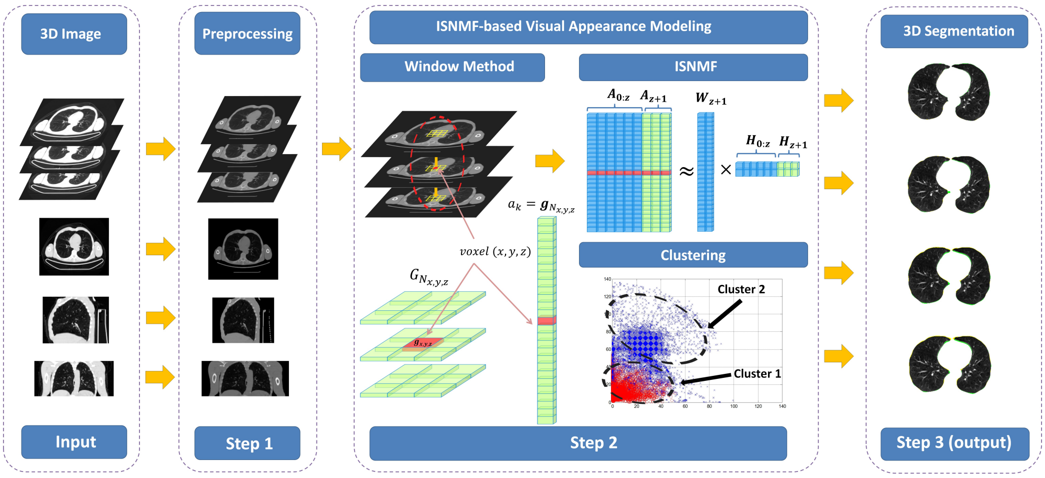

Fig. 2. The proposed framework for 3D Image segmentation

2.3 Segmentation

In this step, we use a clustering approach to obtain the segmentation of the 3D image. First, the K-means clustering approach is applied on voxels in the $\mathbf{H}$ space using J as number of clusters. Then the ℓ^2-norm of each cluster centroid is calculated. Since the neighborhood vector $\mathbf{g}_{\mathbf{N}_{x,y,z}}$ of the voxels belonging to a darker modality (smaller $\mathbf{g}_{x,y,z}$ values) results in smaller ℓ^2-norm, we employ it to identify different modalities. Therefore, various modalities in the image could be identified based on their relative brightness in the image. It means the brighter the modality, the farther its voxels are from the origin in the $\mathbf{H}$ space. Finally, 3D connected component analysis is applied to refine the segmentation by removing the mis-clustered voxels inside the segmented region. The proposed method of segmentation is outlined in Algorithm 2.

Algorithm 2 3D Image Segmentation Using ISNMF-based Visual Appearance Model

1: Preprocess image for removing background
2: Extract ISNMF-based visual appearance model of the 3D image using Algorithm 1
3: Cluster image voxels into J groups using the K-means clustering algorithm [**?**]
4: Discriminate between image clustes based on their relative brigthness and ℓ^2-norm of cluster centroids
5: Refine the segmented regions using a 3D connected component analysis

The proposed framework is completely automatic, and it does not required any user information, unless the number of clusters in the image. It does not have any limitation regarding type of scan, thickness of slices. It is designed to segment healthy and pathological lung. The algorithm does not need any training. As explained, the algorithm does not search for boundaries and fissures, and it will not use airway information.

3 Experimental Results

In order to evaluate the accuracy of the proposed segmentation approach, we assess its performance on fourteen data sets. The CT images have been acquired with a multidetector GE Light Speed Plus scanner (General Electric, Milwaukee, USA) with the following scanning parameters: slice thickness of 2.5 mm reconstructed every 5 mm, scanning pitch 1.5, 140 KV, 100 MA, and F.O.V 36 cm. The size of each 3D test data is $512 \times 512 \times 390$. The CT images contain two classes ($K = 2$): darker lung tissue and the brighter chest region. The system used for the experiment is a Windows 7 64-bit operating system, CPU Intel Core i7 2.67 GHz, RAM 8 GB.

The projected 2D results of our lung segmentation method in the axial, sagittal, and coronal planes are shown in Fig. 3. The performance of our segmentation technique is evaluated using the Dice Similarity Coefficient (DSC). The proposed method achieved mean DSC value of 0.966 for $N_{x,y,z}$ of size $(3 \times 3 \times 3)$. To investigate the sensitivity and robustness of our method, we use $N_{x,y,z}$ of different sizes, where the segmentation results of sizes $(3 \times 3 \times 3)$, $(7 \times 7 \times 3)$, $(11 \times 11 \times 3)$, and $(15 \times 15 \times 3)$ are shown in Fig. 3-(b), 3-(c), 3-(d), and 3-(e) respectively. In addition, the computational time of segmentation is also compared for different size of $N_{x,y,z}$, as shown in Table 1. It is demonstrated that the segmentation time increases by expanding the size of $N_{x,y,z}$, as indicated in Table 1. The reason is that by expanding the size of $N_{x,y,z}$, the dimension size of matrix A will be increased, which directly affects the computational time of factorization using NMF. Moreover, by expanding the size of $N_{x,y,z}$, DSC measure decreased, and the segmentation quality decreased, as depicted in Fig. 3.

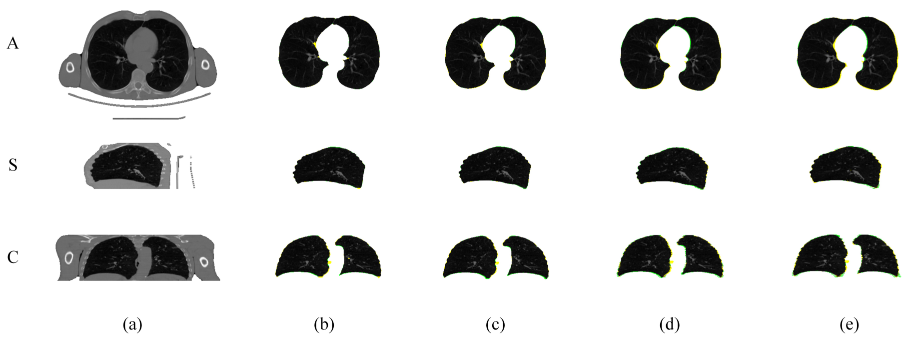

Fig. 3. Results of 3D lung segmentation projected onto 2D axial (A), sagittal (S), and coronal (C) planes for visualization: 2D profiles of the original CT images with segmentation of CT using (a), $N_{x,y,z}$ of size 3x3x3 (b), $N_{x,y,z}$ of size 7x7x3 (c), $N_{x,y,z}$ of size 11x11x3 (d), $N_{x,y,z}$ of size 15x15x3

Table 1. Comparison of segmentation time vs. DSC for different size of $N_{x,y,z}$

$N_{x,y,z}$ size	$3 \times 3 \times 3$	$5 \times 5 \times 3$	$7 \times 7 \times 3$	$11 \times 11 \times 3$	$15 \times 15 \times 3$
Times (sec)	364	638	992	1488	2918
DSC	0.988	0.984	0.981	0.969	0.960