Neuroinformatics
https://doi.org/10.1007/s12021-018-9364-2

ORIGINAL ARTICLE

Patch-Based Label Fusion with Structured Discriminant Embedding for Hippocampus Segmentation

Yan Wang [1,2] · Guangkai Ma [3] · Xi Wu [4] · Jiliu Zhou [1,4]

Abstract
Automatic and accurate segmentation of hippocampal structures in medical images is of great importance in neuroscience studies. In multi-atlas based segmentation methods, to alleviate the misalignment when registering atlases to the target image, patch-based methods have been widely studied to improve the performance of label fusion. However, weights assigned to the fused labels are usually computed based on predefined features (e.g. image intensities), thus being not necessarily optimal. Due to the lack of discriminating features, the original feature space defined by image intensities may limit the description accuracy. To solve this problem, we propose a patch-based label fusion with structured discriminant embedding method to automatically segment the hippocampal structure from the target image in a voxel-wise manner. Specifically, multi-scale intensity features and texture features are first extracted from the image patch for feature representation. Margin fisher analysis (MFA) is then applied to the neighboring samples in the atlases for the target voxel, in order to learn a subspace in which the distance between intra-class samples is minimized and the distance between inter-class samples is simultaneously maximized. Finally, the k-nearest neighbor (kNN) classifier is employed in the learned subspace to determine the final label for the target voxel. In the experiments, we evaluate our proposed method by conducting hippocampus segmentation using the ADNI dataset. Both the qualitative and quantitative results show that our method outperforms the conventional multi-atlas based segmentation methods.

Keywords Multi-atlas based method · Margin fisher analysis · Structured discriminant embedding · Subspace learning · Patch-based label fusion

Introduction

As an advanced imaging modality, magnetic resonance (MR) has been extensively applied in pathology, diagnostic imaging, neuro-analysis, and clinical medicine (Wu et al. 2013; Zhou et al. 2014, 2016; Wang et al. 2016a, b). The segmentation of hippocampal structures in MR images is of particular importance to various neuroimaging studies, including brain disorders and brain anatomy (Zu et al. 2017; Dong et al. 2016; Wu et al. 2015a). However, due to the large amount of clinical data, manual segmentation is quite laborious, time-consuming and tedious. In addition, manual segmentation often suffers from the disagreement between different clinicians. Therefore, the development of automatic hippocampus segmentation has become a hot topic in the field of medical image analysis (Carmichael et al. 2005; Zarei et al. 2013).

The advances in computer vision and machine learning have made possible the ability to automate the segmentation process (Jafari-Khouzani et al. 2011; He et al. 2017; Rincón et al. 2017). Among the existing medical image segmentation techniques, atlas-based methods have attracted great attention (Zhu et al. 2017; Chen et al. 2017). This technique first employs deformable image registration to construct correspondences between the pre-labeled atlas images and the target. Then, using the obtained deformation field, labels in the atlas are further propagated to the target image space. Obviously, the anatomical differences between the target and the atlas images could influence the image registration accuracy, thus

✉ Yan Wang
wangyanscu@hotmail.com

[1] College of Computer Science, Sichuan University, Chengdu, China

[2] Fujian Provincial Key Laboratory of Information Processing and Intelligent Control (Minjiang University), Fuzhou 350121, China

[3] Space Control and Inertial Technology Research Center, Harbin Institute of Technology, Harbin, China

[4] Department of Computer Science, Chengdu University of Information Technology, Chengdu, China

Published online: 07 March 2018

greatly affecting the final segmentation performance (Wu et al. 2015b). To alleviate the impact of anatomical variability, multi-atlas based methods have been extensively studied (Wang et al. 2013; Sundar et al. 2009). By fusing the propagated labels of multiple atlases in the target image space, multi-atlas based methods can achieve more robust and accurate segmentation results. The performance of multi-atlas based methods relies on both the registration accuracy and the label fusion strategy. Consequently, in addition to optimizing the registration, many researchers focus on improving segmentation performance by exploring more effective label fusion strategies. Particularly, patch-based label fusion methods have been widely studied for multi-atlas based segmentation (Liao et al. 2013; Chen et al. 2015; Rekik et al. 2015). The patch-based methods are proposed based on a basic assumption, that is, if two image patches are similar in their appearance, they should have the same anatomical label. Since the hippocampal structures in MR images usually share similar intensity values with the neighboring tissues, the traditional patch-based methods which just utilize the intensity features as the feature appearance might restrict the segmentation performance. Due to the lack of discriminating features, the original feature space defined by image intensities may limit the description accuracy. Another limitation is that the label propagation in most traditional multi-atlas based methods is implemented under a voxel-wise strategy, which cannot adequately utilize the local label information to determine the final label of the target sample.

In order to tackle the above limitations, we propose a patch-based label fusion with structured discriminant embedding method for hippocampus segmentation. Specifically, we first linearly register each atlas to the target image. Then, multi-scale intensity features and texture features are extracted from the image patch for feature representation. After that, a local discriminant subspace is learned from the candidate training set based on the margin fisher analysis (MFA) strategy, where the distance between intra-class samples is minimized and the distance between inter-class samples is simultaneously maximized. Finally, the k-nearest neighbor (kNN) classifier is employed in the learned subspace to determine the final label for the target voxel. The novelties and contributions of the paper are as follows.

1) To enhance the feature description ability, we extract both the texture information and multi-scale intensity information from the image patch as the feature representation for each voxel.
2) To strengthen feature discrimination, we adopt margin fisher analysis (MFA) to ensure the minimization of the distance between intra-class samples and the maximization of the distance between inter-class samples.
3) In label fusion, instead of utilizing the label value of each voxel, we employ the label patch as the structured class label to preserve local anatomical structure information.

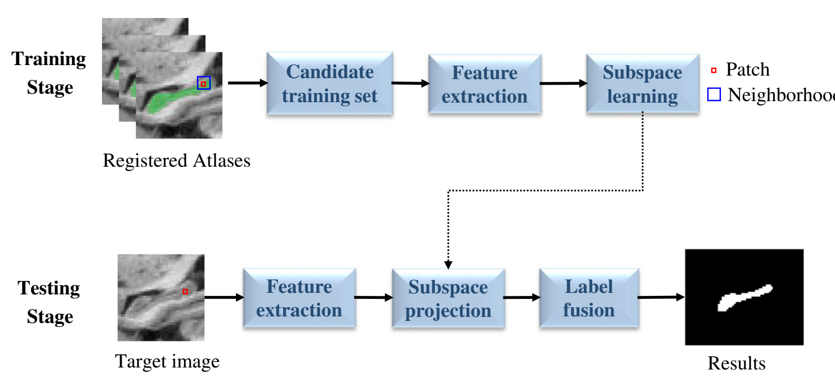
Our proposed method for segmenting hippocampus is validated on 133 MR images from ADNI dataset, including 46 normal controls (NC), 45 Mild Cognitive Impairments (MCI) and 42 Alzheimer's disease (AD) subjects. The experimental results indicated that our method could achieve better segmentation performance compared with the traditional multi-atlas based segmentation method.

The rest of the paper is organized as follows. The second section presents the details of the proposed method. In the third section, we extensively investigate the performance of the proposed method with respect to different parameters, and also verify the rationality and validity for every step of the proposed method. The comparison with the conventional multi-atlas based methods is discussed in the third section. Finally, we conclude this paper in the fourth section.

Method

Figure 1 schematically illustrates the overview of the proposed method which consists of a training stage and a testing stage. Similar to typical multi-atlas based segmentation methods, we first linearly register each atlas to the target image. In the training stage, we generate the candidate training set for each target voxel to be segmented in the target image,

Fig. 1 Schematic illustration of the proposed method



using voxels of registered atlases within a spatial neighborhood of the target voxel. Then, for each training sample, a set of features is extracted for capturing effective information from the hippocampus using abundant texture descriptors ad multi-scale patch intensities. Moreover, its associated class label is determined by the label of the voxel on the label map. In the subspace learning submodule, we learn a local discriminant subspace from the candidate training samples by using MFA. Afterwards, all the training samples are projected into the learned subspace. In the testing stage, a kNN classifier is employed in the learned subspace to estimate the label of the target voxel. Specifically, the feature representation of the target voxel is first extracted from its surroundings in the target image and then projected into the learned subspace. Finally, the target voxel is compared with each training sample in the learned subspace, and the final label of the target voxel is determined by the most common label of the *k*-nearest neighbor samples.

In the following subsection, we present the details of the proposed segmentation method. This includes generation of a candidate training set, feature extraction, structured subspace learning, and patch-wise label fusion, respectively.

Generation of a Candidate Training Set

Due to variability among different atlases, linear image registration cannot achieve perfect alignment of all voxels in an image. Therefore, the direct usage of corresponding voxels in the atlases for a target voxel, to determine the target voxel label, may lead to unsatisfactory results. In this paper, we employ a local patch based method to generate a candidate training set. Specifically, given one target voxel x of the target image, voxels in its neighborhood $V(x)$ (with the size $\omega \times \omega \times \omega$) of all atlases are used to generate the training samples. This produces $N \times \omega \times \omega \times \omega$ candidate training samples $\left\{ \left(\overrightarrow{f}_{i,j}, l_{i,j} \right) | i = 1, 2, \ldots N, j \in V(x) \right\}$ from N registered atlases, where $\overrightarrow{f}_{i,j}$ is a feature vector extracted from voxel j of the ith atlas based on the feature extraction method which would be discussed in the next subsection, and $l_{i,j} \in \{+1, -1\}$ is the segmentation label of each candidate training sample. The generated candidate training samples include different degrees of similarity with the target voxel. In order to balance the positive and negative samples for subspace learning, we respectively extract the same number of training samples from the hippocampus and non-hippocampus regions. Specifically, for the positive training samples, we select the q_1 most similar samples with the target image patch and randomly select another q_2 samples from the remaining training samples in the hippocampus region. In a similar manner, $q_1 + q_2$ negative samples are selected from the non-hippocampus regions, in which q_1 are the most similar with the target image patch and q_2 are randomly selected. The similarity is determined by the following well-known structural similarity (ss) measure:

$$ss = \frac{2\mu_x \mu_{i,j}}{{\mu_x}^2 + {\mu_{i,j}}^2} \times \frac{2\sigma_x \sigma_{i,j}}{{\sigma_x}^2 + {\sigma_{i,j}}^2} \quad (1)$$

where μ represents the mean and σ represents the standard deviation of patches centered at the target voxel x and voxel j of the ith atlas.

Feature Extraction

As described above, solely extracting image intensity information may not be sufficient to distinguish hippocampus structures. In order to obtain more discriminative features, we extract texture information and multi-scale intensity information as the feature representation. The texture features consist of outputs from the first-order difference filters (FODs), second-order difference filters (SODs), 3D Hyperplan filters, 3D Sobel filters, Laplacian filters and range difference filters. The detailed description of the textual features is presented in the Appendix. By extracting the above features, we can capture rich textual information which embedded in the target image.

For the intensity feature extracting in most patch-based segmentation methods, each voxel in an image patch contributes equally to generate intensity features. However, this approach may not adequately capture the complex tissue appearance patterns expressed in hippocampus structure. In this paper, we use a multi-scale strategy which encodes both local and semi-local image information to characterize an image patch (Wu et al. 2015c). Specifically, the entire image patch is first divided into several non-overlapping scales, propagating from the center voxel to the boundaries of the patch. Different Gaussian filters are used in this paper to replace the original intensity values with convolved intensity values. Using three layers as an example, Fig. 2 shows the procedure for constructing a multi-scale image patch via Gaussian filters. Considering the final purpose of the segmentation procedure is to determine the label of the center voxel, we use a fine scale (the right subgraph in Fig. 2) to capture the center information of the image patch (in red in Fig .2), and increasingly larger scales to capture coarse scale information as the distance to the patch center increases (in yellow and blue). In this paper, we use three scales to generate the intensity features.

Structured Subspace Learning

A candidate training set can be generated using the above feature extraction method. It is further used to train a subspace in order to enhance the discernibility of the feature representation for each voxel. As described in "Generation of a Candidate Training Set" section, $q_1 + q_2$ samples were respectively selected from hippocampus and non-hippocampus

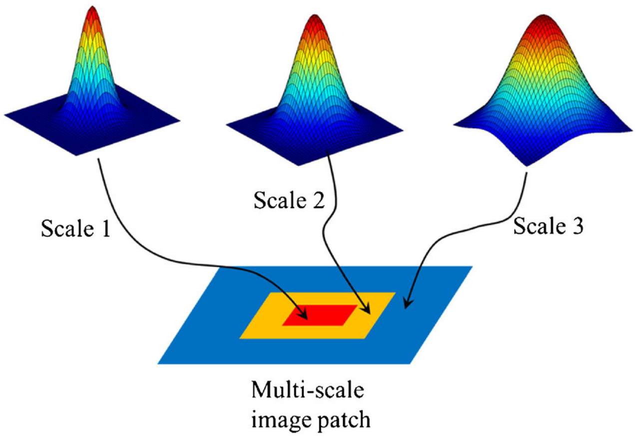


Fig. 2 Multi-scale image patch with different Gaussian filters. Three regions labeled by different colors are filtered by different Gaussian kernels, respectively

regions and used as the final positive and negative training samples. For simplicity, we use $U = [u_1, u_2, \ldots, u_M]$, $u_i \in \mathbb{R}^d$ to denote training samples together with the corresponding class labels $Y = [y_1, y_2, \ldots, y_M]$, $y_i \in \{+1, -1\}$. Here, $M = 2(q_1 + q_2)$ denotes the number of training samples, and d is the feature dimension. MFA aims to learn a subspace in which the intra-class manifold is compacted (i.e., intra-class compactness), while the manifold margin between different classes is enlarged based on the margin criteria (i.e., inter-class separability).

Two undirected graphs, the intrinsic graph $G^I = \{U, S^I\}$ and the penalty graph $G^P = \{U, S^P\}$, were constructed according to graph embedding theory to respectively characterize the intra-class manifold structure and the manifold margin of different classes. In this case, U is the training samples set, and $S^I, S^P \in \mathbb{R}^{M \times M}$ are the corresponding similarity matrices. To characterize intra-class compactness with the margin criteria, the similarity matrix S^I of the intrinsic graph G^I is defined as follows:

$$S_{ij}^I = \begin{cases} 1, & \text{if } i \in N_{k_1}^+(j) \text{ or } j \in N_{k_1}^+(i) \\ 0, & \text{else} \end{cases} \tag{2}$$

where $N_{k_1}^+(j)$ indicates the index set of the k_1 nearest neighbors of u_j in the same class. On the other hand, MFA defines the similarity matrix S^P of the penalty graph G^P to characterize interclass separability with the margin criteria, as follows:

$$S_{ij}^P = \begin{cases} 1, & \text{if } (i,j) \in P_{k_2}(y_i) \text{ or } (i,j) \in P_{k_2}\left(y_j\right) \\ 0, & \text{else} \end{cases} \tag{3}$$

where $P_{k_2}(y)$ is a set of data pairs that are the k_2 nearest pairs among the set $\{(i,j), y_i = y, y_j \neq y\}$. Then, the corresponding diagonal matrix D and the Laplacian matrix L of the intrinsic graph G^I are defined as

$$L = D - S, \quad D_{ii} = \sum_{j \neq i} S_{ij} \quad \forall i. \tag{4}$$

The Lapaican matrix L^P of the penalty graph G^P can be defined similarly to Eq. (4). Finally, according to the graph embedding framework, the subspace projection matrix $\Phi_{\text{MFA}} \in \mathbb{R}^{d \times p}$, where p is the dimension of the subspace, can be computed by solving the objective function

$$\Phi_{\text{MFA}}^* = \arg\max_{\Phi_{\text{MFA}}} \frac{Tr\left(\Phi_{\text{MFA}}^T U L U^T \Phi_{\text{MFA}}\right)}{Tr\left(\Phi_{\text{MFA}}^T U L^P U^T \Phi_{\text{MFA}}\right)}, \tag{5}$$

where $Tr(\cdot)$ is the trace operator of the matrix. Specifically, the projection matrix Φ_{MFA} consists of the eigenvectors corresponding to the largest eigenvalues of matrix $\left(\Phi_{\text{MFA}}^T U L U^T \Phi_{\text{MFA}}\right)^{-1} \cdot \left(\Phi_{\text{MFA}}^T U L^P U^T \Phi_{\text{MFA}}\right)$. After obtaining the projection function Φ_{MFA}, each feature representation u_i is projected into the learned space as follows:

$$w_i = \Phi_{\text{MFA}}^T \cdot u_i \tag{6}$$

To preserve local anatomical structure information, we extend the traditional subspace learning for patch-wise label fusion. For the voxel-wise label fusion, we can directly extract voxel labels from the label map and use them as class labels to perform supervised subspace learning. In contrast, for patch-wise label fusion, each voxel is assigned with one label patch. Specifically, we extend the class label $y_i \in \mathbb{R}$ $(i = 1, \ldots, M)$ of each voxel to the structured class label patch $\mathcal{Y}_i \in \mathbb{R}^{t \times t \times t}$ corresponding to the label patch centered at the voxel, where $t \times t \times t$ is the size of the label patch. However, for the training set U together with the corresponding structured class label set $\{\mathcal{Y}_1, \ldots, \mathcal{Y}_M\}$, MFA cannot be directly used to train the subspace. In order to perform MFA similar to that in the original class label space, we adopt k-means cluster method to $\{\mathcal{Y}_1, \ldots, \mathcal{Y}_M\}$, and thus obtain the corresponding subclass labels $\{y_1', \ldots, y_M'\}$, $y_i' \in \{+1, -1\}$ in the unsupervised manner. Then, we replace $\{\mathcal{Y}_1, \ldots, \mathcal{Y}_M\}$ by $\{y_1', \ldots, y_M'\}$ and combine them with U to learn the subspace using MFA. In this case, samples with similar label patches are compacted and samples with dissimilar label patches are separated.

Patch-Wise Label Fusion

In the testing stage, a kNN classifier is employed in the learned subspace to estimate the label of the target voxel. Specifically, the feature representation of the target voxel is first extracted from its surroundings in the target image and then projected into the learned subspace. By comparing the feature similarity between the target voxel and the training samples in the learned subspace, k-nearest training samples are selected to determine the final label of the target voxel, as shown in Fig. 3.

Learned subspace

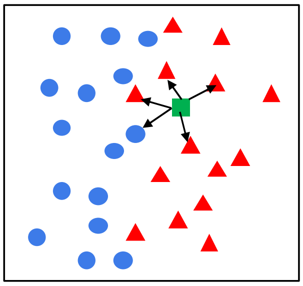


Fig. 3 Label fusion based on k-nearest training samples, green block is the image patch of the target voxel, red triangle the positive samples and blue circle the negative samples

Then, the estimation of the label patch centered at the target voxel x is obtained by:

$$\hat{L}_x = \frac{\sum_{i=1}^{k} w(P_x, P_i)\mathcal{Y}_i}{\sum_{i=1}^{k} w(P_x, P_i)} \tag{7}$$

where $\mathcal{Y}_i$ is the corresponding label patch of the selected k training samples, $w(P_x, P_i)$ is the weight which is typically determined by the similarity measure.

$$w(x, x_i) = e^{-\frac{\|P_x - P_i\|^2}{\sigma_x}}, i = 1, 2, \dots, k \tag{8}$$

where $\|\cdot\|_2$ represents the L2 norm between P_x and P_i, σ_x is the decay parameter controlling the strength of penalty in the exponential way. It is worth noting that when testing the target voxel x, the labels of its neighbor voxels are also estimated, leading the overlapped estimations. Here, we adopt an averaging method to fuse the overlapped estimations and use it as the final label estimation. Compared with voxel-wise label fusion, patch-wise label fusion uses the whole label patches to estimate the final label, thus preserving the local anatomical structure information in the segmentation.

Experiments

Dataset Described in this section are several experiments conducted to evaluate the performance of our method for hippocampus segmentation using the publically available ADNI dataset. In the experiments, we randomly selected 46 NC subjects, 45 MCI subjects, and 42 AD subjects from the ADNI dataset. The hippocampal segmentations from ADNI were regarded as the ground truth. For all selected subjects, three standard preprocessing steps were first performed, including 1) skull-stripping using a learning-based meta-algorithm (Shi et al. 2012), 2) N4-based bias field correction (Tustison et al. 2010), and 3) histogram matching to normalize the intensity range (Shen 2007). In the experiments, we used a leave-one-out cross-validation to evaluate the performance of our method, i.e., when one subject is tested, the other subjects are regarded as atlases. Affine registration was performed to align each atlas image with the target image using FLIRT in the FSL toolbox, with 12 degrees of freedom and default parameters. These included normalized mutual information similarity metric and search range of ±20 in all directions.

Evaluations For comparison purposes, the conventional patch-based method by non-local weighting (Nonlocal-PBM) and the recently proposed sparse patch-based labeling (Sparse-PBM) were also evaluated on the same dataset. To quantitatively evaluate the proposed method, five metrics were used for performance evaluation. The degree of overlap was measured for two ROIs V_s and V_g, where V_s and V_g are the sets of object (hippocampus) voxels automatically segmented by the segmentation method and manually segmented by clinical expert, respectively.

1) Dice similarity coefficient (DSC) is a comprehensive similarity metric that measures the degree of overlap of two ROIs

$$DSC = 2 \times \frac{|V_s \cap V_g|}{|V_s| + |V_g|} \tag{9}$$

where $|\cdot|$ is the cardinality of a set.

2) Jaccard similarity coefficient (JSC), which is a statistic used for comparing the similarity and diversity of two ROIs, is described as follows:

$$JSC = \frac{|V_s \cap V_g|}{|V_s \cup V_g|} = \frac{|V_s \cap V_g|}{|V_s| + |V_g| - |V_s \cap V_g|} \tag{10}$$

3) Precision Index (PI) is the ratio between the overlap of two ROIs and the ROI manually segmented by clinical expert, as follows:

$$PI = \frac{|V_s \cap V_g|}{|V_g|} \tag{11}$$

4) Recall Index (RI) is the ratio between the overlap of two ROIs and the ROI segmented by the segmentation method, as follows:

$$RI = \frac{|V_s \cap V_g|}{|V_s|} \tag{12}$$

5) Hausdorff distance (HD) measures surface distance between segmentation results:

$$HD = \max\{\max_{a \in V_s}\min_{b \in V_g} d(a,b), \max_{b \in V_g}\min_{a \in V_s} d(a,b)\} \quad (13)$$

where $d(a, b)$ represents the Euclidean distance between a and b.

Theoretically, segmentation results with a higher DSC, JSC, PI, RI, R and lower HD represent better segmentation performance.

Parameter Tuning In order to determine of the optimal values for k_1 and k_2 in the MFA analysis, we test the combinations of k_1, $k_2 \in \{5, 10, 20, 40\}$. For the candidate training set, the search neighborhood V_x is set as $7 \times 7 \times 7$, and the number of the candidate training samples is set from 100 to 500 with a step of 100. For the dimensionality of the learned subspace, we test p from 10 to 150 with a step of 20. For the label fusion, we select the side length of label patches from 1 to 15 with a step of 2 and set each side length to be equal. For the kNN classifier, k selected from $\{1\ 5\ 10\ 20\}$.

Influence of Components in the Proposed Method

In this section, we analyze four main components of the proposed method and their influence on the performance of our method given different input parameters: 1) multi-scale intensity patch and texture features, 2) structured subspace embedding, 3) patch-wise label fusion, and 4) kNN classifier.

Multi-Scale Intensity Patch and Texture Features

In this section, we evaluate the performance of the features used in our method including multi-scale intensity patch and multiple texture descriptors. To evaluate the segmentation performance of each feature type, we compared our method to procedures using only intensity patches with the original scale (Patch), intensity patches with three different scales (Patch-scale1, Patch-scale2, and Patch-scale3), texture descriptors (Texture), multi-scale intensity patch (Multi-scale Patch), the combination of intensity patch and texture descriptors (Patch+Texture) and the combination of multi-scale intensity patch and texture descriptors (Multi-scale Patch+Texture). Moreover, we varied the size of intensity patch from $3 \times 3 \times 3$ to $11 \times 11 \times 11$. In this experiment, we used DSC measure and implemented our method by using voxel-wise label fusion with MFA-based and NN classifier. For MFA, the dimension of the subspace was set to 150, and k_1 and k_2 were set to 10 and 20, respectively. For Patch-scale1, Patch-scale2 and Patch-scale3, we respectively set the variances of the three Gaussian filters to 1.0, 1.414, and 2.0 moving from the fine to coarse scales. Figure 4 shows the mean DSCs for the left and right hippocampus segmentation for several feature types plotted against image patch size.

It is evident in the figure that using texture descriptors alone as the feature representation leads to a worse performance than using only intensity features. However, combining texture descriptors and intensity features results in an improvement. Specifically, compared with the Patch method for left hippocampus segmentation, the Patch+Texture method obtains mean DSC increases of 0.7%, 0.7%, 0.4%, 0.4% and 0.8% from the size of the intensity patch $3 \times 3 \times 3$ to $11 \times 11 \times 11$. Compared with the Multi-scale Patch method for left hippocampus segmentation, the Multi-scale Patch+Texture method also results in an improvement, with mean DSC increases of 0.3%, 0.1%, 0.3%, 0.2% and 0.8% over different image patch sizes. For the right hippocampus, the Patch+Texture and the Multi-scale Patch+Texture methods also

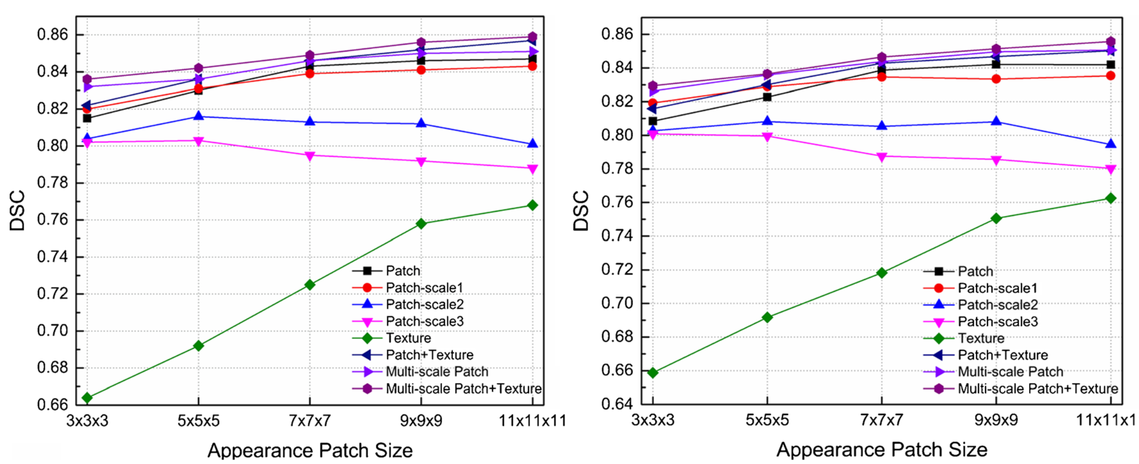


Fig. 4 Left: mean DSC for left hippocampus segmentation produced by the Patch, Patch-scale1, Patch-scale2, Patch-scale3, Texture, Patch+Texture, Multi-scale Patch and Multi-scale Patch+Texture methods. Right: mean DSC for right hippocampus segmentation produced by the corresponding methods

Table 1 The mean of DSC, JSC, PI, RI, R and HD (mm) of left and right hippocampus on ADNI dataset, produced by the Patch and Multi-scale patch+Texture method respectively

	DSC	JSC	PI	RI	HD(mm)
Patch	0.842/0.847	0.739/0.746	0.877/0.892	0.841/0.855	4.12/3.95
Multi-scale Patch+Textures	0.856/0.859	0.752/0.755	0.891/0.903	0.859/0.862	4.01/3.89

The value in the left '/' denotes the segmentation accuracy of the left hippocampus, while the value in the right '/' denotes the segmentation accuracy of the right hippocampus

respectively obtain improvements over methods using different image patch sizes, when compared with the Patch and the Multi-scale Patch methods. These analysis indicate that a combination of intensity and texture features can improve segmentation accuracy. From Fig. 3, we can also see that, when using patch sizes of $3 \times 3 \times 3$ and $5 \times 5 \times 5$, intensity patch on a fine scale (Patch-scale1) yields better performance than the original scale (Patch). However, for patch sizes of $7 \times 7 \times 7$, $9 \times 9 \times 9$ and $11 \times 11 \times 11$, the Patch method obtains a higher mean DSC than the Patch-scale1 method. This indicates that when using a small patch size, the Patch method is inferior to the Patch-scale1 method. This could be due to a lack of local information in the larger region. However, when increasing the patch size, the Patch method with fine resolution is superior to Patch-scale1. The performance of Patch-scale2 and Patch-scale3 is much worse than Patch and Patch-scale1 over different image patch sizes because the resolution is too low. When combining intensities with different resolutions in one patch, the Multi-scale Patch method achieves the best performance among methods using intensities with a single resolution. It is worth noting that, since MFA can extract discriminant features and remove redundancy features, we use the largest patch size ($11 \times 11 \times 11$) to obtain the best performance.

Table 1 lists the mean of five DSC, JSC, PI, RI and HD measures for the left and right hippocampus for the Patch and Multi-scale patch+Texture methods. It can be seen that the Multi-scale patch+Texture method achieves an improvement over the measures. On both the left and right hippocampus, compared with the Patch method, the Multi-scale patch+Texture method obtains statistically significant improvements with P-values of $P < 0.001$ for all measures.

Table 2 The mean DSC of left and right hippocampus, produced by our method with different parameters k_1 and k_2, and the best results are in bold

DSC	$k_2 = 5$	$k_2 = 10$	$k_2 = 20$	$k_2 = 40$
$k_1 = 5$	0.813/0.803	0.823/0.825	0.834/0.832	0.845/0.841
$k_1 = 10$	0.815/0.817	0.831/0.835	0.843/0.846	**0.856**/0.859
$k_1 = 20$	0.818/0.818	0.836/0.837	0.847/0.851	0.853/**0.861**
$k_1 = 40$	0.816/0.821	0.834/0.841	0.851/0.853	0.852/0.859

See Table 1 for description of '/'

Structured Subspace Learning

To evaluate the subspace projection based on MFA, we primarily investigate the effect which MFA parameters have on segmentation performance in our proposed method. Specifically, the considered parameters include the number of nearest intra-class samples k_1, the number of nearest inter-class samples k_2, the number of training samples M and the dimension of the subspace p. In this experiment, we use multi-scale patch intensities and texture descriptors with the patch size of $11 \times 11 \times 11$ as the feature representation and perform our method using voxel-wise label fusion with MFA and NN classifier. To highlight the effect of each parameter on segmentation performance, parameters are individually tested as other parameters are fixed. To determine the parameters k_1 and k_2, we set $k_1, k_2 \in \{5, 10, 20, 40\}$ and considered all their combinations. To test the parameter M, we vary M from 100 to 500 with a step of 100. For the parameter p, we vary p from 10 to 150 with a step of 20. Table 2 shows the mean DSC for left and right hippocampus segmentations with different parameters k_1 and k_2 by the proposed method.

From Table 2, we observe that when $k_1 = 10$ and $k_2 = 40$, the highest average DSC (0.856) for left hippocampus is obtained, and when $k_1 = 20$ and $k_2 = 40$, the highest averaged

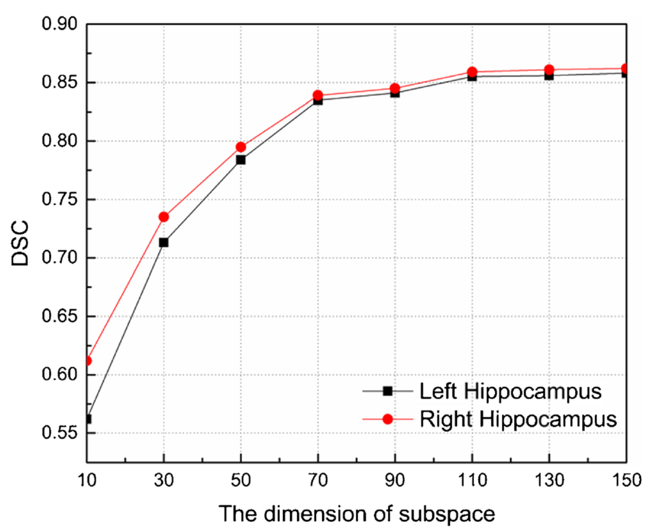


Fig. 5 Mean DSC for segmentation of left and right hippocampus using MFA with different subspace dimensions

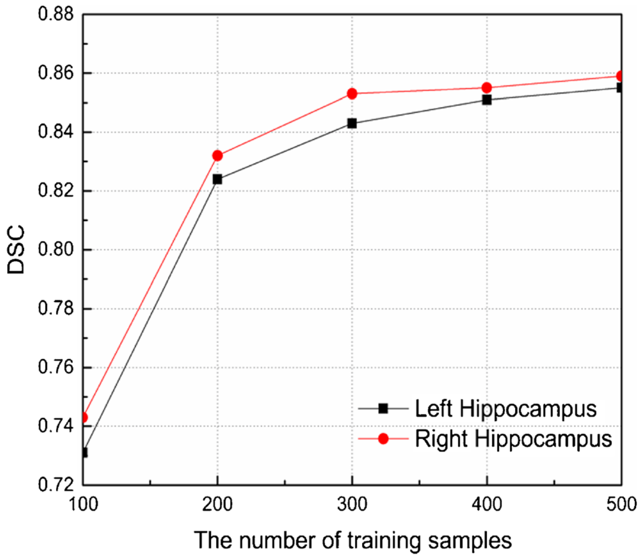


Fig. 6 Mean DSC for segmentation of left and right hippocampus generated using our method with different numbers of training samples

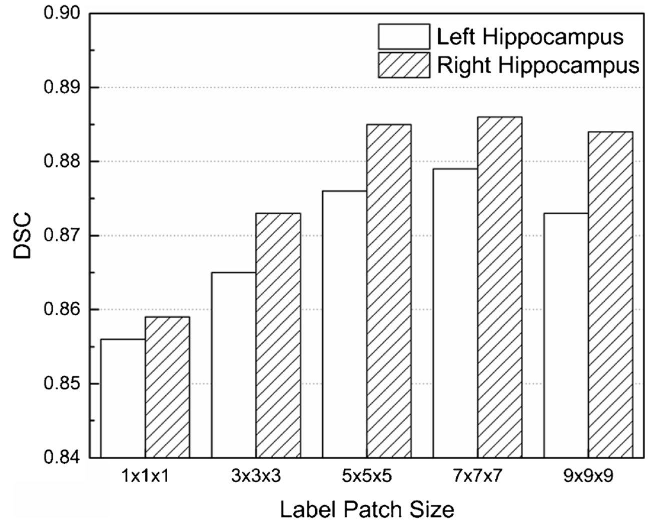


Fig. 7 The effect of using different label patch sizes for segmentation for left and right hippocampus

DSC (0.861) for right hippocampus is achieved. When fixing k_1 at 5, 10, 20, and 40, we increase k_2 from 5 to 40 and then respectively obtain an average DSC increase of 3.2%, 4.1%, 3.5% and 3.6% for left hippocampus and an average DSC increase of 3.8%, 4.2%, 4.3% and 3.8% for right hippocampus. In turn, when fixing k_2 at 5, 10, 20, and 40, we increase k_1 from 5 to 40 and then respectively obtain an averaged DSC increase of 0.3%, 1.1%, 1.7% and 0.7% for left hippocampus and an averaged DSC increase of 1.8%, 1.6%, 2.1% and 1.8% for right hippocampus. Experimental results indicate that increasing the number of nearest inter-class samples k_2 is more effective for improving the segmentation performance than increasing the number of nearest intra-class samples k_1. In addition, it also indicates that for margin criteria in MFA, the selecting of k_1 and k_2 is of great significance to segmentation performance, in order to characterize intra-class compactness and interclass separability with margin criteria. Figure 5 plots the mean DSCs of our method with different dimensions of the learned subspace using MFA. As shown in Fig. 5, along with an increase in the subspace dimension, the average DSCs for both left and right hippocampus experience rapid growth in the beginning from 10 to 70, a stable increase trend in the middle from 70 to 90, and maintain a high segmentation performance level in the end from 90 to 150. Experimental results indicate that the segmentation performance of our method can remain stable with a large enough subspace dimension when using MFA.

Figure 6 shows the mean DSCs of our method using a different number of training samples. It can be seen in the figure that increasing the number of training samples improves segmentation accuracy.

In order to demonstrate the efficacy of subspace learning using MFA, we compare the results of our method with and without a subspace projection procedure. Table 3 gives the comparison results regarding five measures. For simplicity, we use MFA + NN and NN to denote our methods with and without the subspace projection procedure using voxel-wise label fusion, respectively. We also present the Label patch based MFA + NN and the Label patch based NN to denote our methods with and without the subspace projection procedure using patch-wise label fusion. Compared with NN, MFA + NN obtains increases of 2.1%, 1.0%, 1.7%, and 1.3% for the left hippocampus segmentation and 2.0%, 1.4%, 2.4% and 0.4% for the right hippocampus segmentation in terms of DSC, JSC, PI, and RI, respectively. It also produced an HD decrease of 0.28 mm and 0.6 mm for left and right hippocampus segmentation, respectively. Meanwhile, compared with the Label patch based NN, the Label patch based MFA + NN respectively obtains increases of 2.3%,

Table 3 The mean of DSC, JSC, PI, RI, and HD (mm) of left and right hippocampus on ADNI dataset, produced by NN, MFA + NN, Label patch based NN and Label patch based MFA + NN, respectively

Method	DSC	JSC	PI	RI	HD (mm)
NN	0.835/0.831	0.742/0.739	0.874/0.871	0.846/0.851	4.29/4.36
MFA + NN	0.856/0.859	0.752/0.755	0.891/0.903	0.859/0.862	4.01/3.89
Label patch based NN	0.846/0.851	0.749/0.753	0.884/0.895	0.852/0.855	3.87/3.76
Label patch based MFA + NN	0.879/0.889	0.773/0.789	0.902/0.914	0.889/0.891	3.01/2.84

See Table 1 for description of '/'

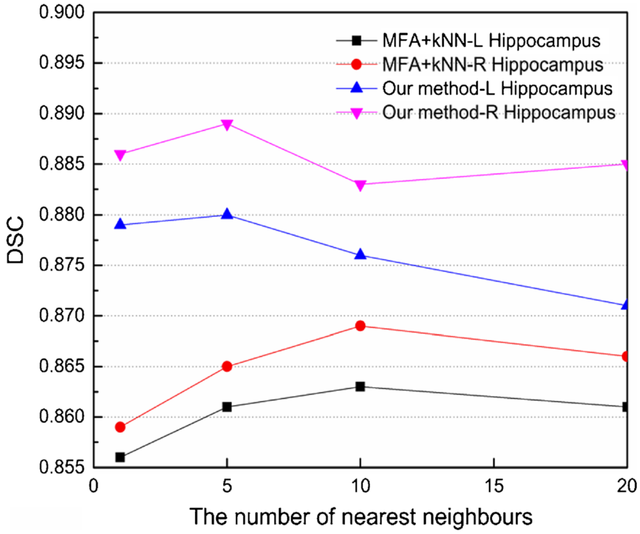


Fig. 8 The effect of using kNN classifier with different numbers of nearest neighbors for segmenting left and right hippocampus

2.1%, 1.1% and 3.0% for the left hippocampus segmentation and 2.7%, 3.4%, 1.1% and 2.9% for the right hippocampus segmentation in terms of DSC, JSC, PI and RI. It also produced an HD decrease of 1 mm and 1.05 mm for left and right hippocampus segmentation, respectively. Experimental results indicate that the employment of a subspace learning procedure based on MFA can improve the segmentation performance for both voxel-wise and patch-wise label fusion. In addition, it is clearly evident in Table 3 that the performance of patch-wise label fusion is superior to voxel-wise label fusion. The impact of label patch in label fusion would be explored in more detail in "Patch-Wise Label Fusion" section.

Patch-Wise Label Fusion

In our method, we adopt k-means cluster method to assign label patches of training samples with corresponding class labels. These class label, together with the feature representations of training samples, are then used to train the subspace in which patch-wise label fusion is performed using the NN classifier. In the evaluation of patch-wise label fusion performed in our method, we analyze the effect of label patch size on segmentation performance. In this experiment, we varied the label patch size from $1 \times 1 \times 1$ to $9 \times 9 \times 9$ with a

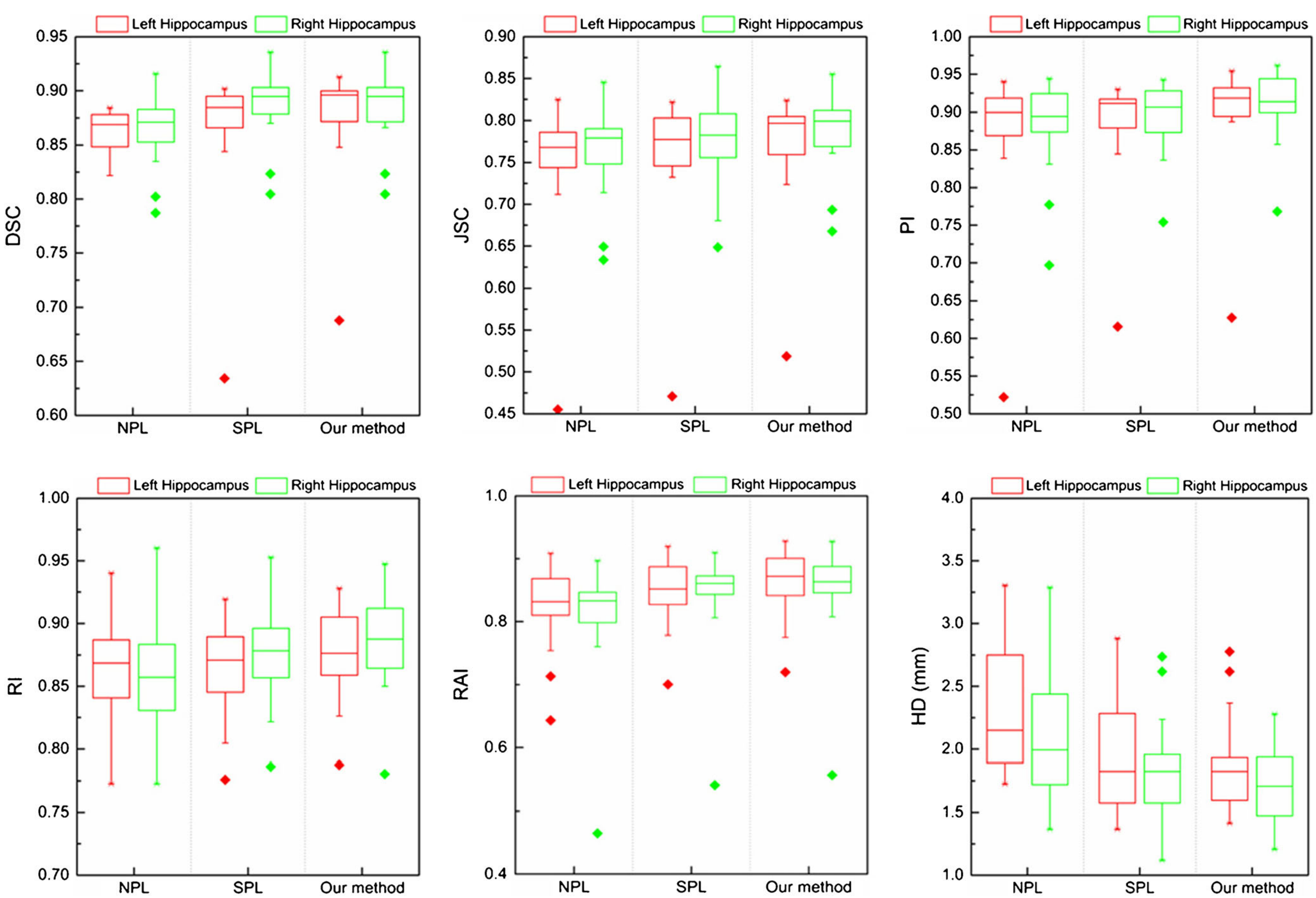


Fig. 9 Box plots of mean DSC, JSC, PI, RI, RAI, and HD (mm) for left and right hippocampus. In each box, the central mark is the median and edges of the box denote the 25th and 75th percentiles. Whiskers extend from each end of the box to adjacent values in the dataset and the extreme values within 1 interquartile range from the ends of the box. Outliers are data with values beyond the ends of the diamond

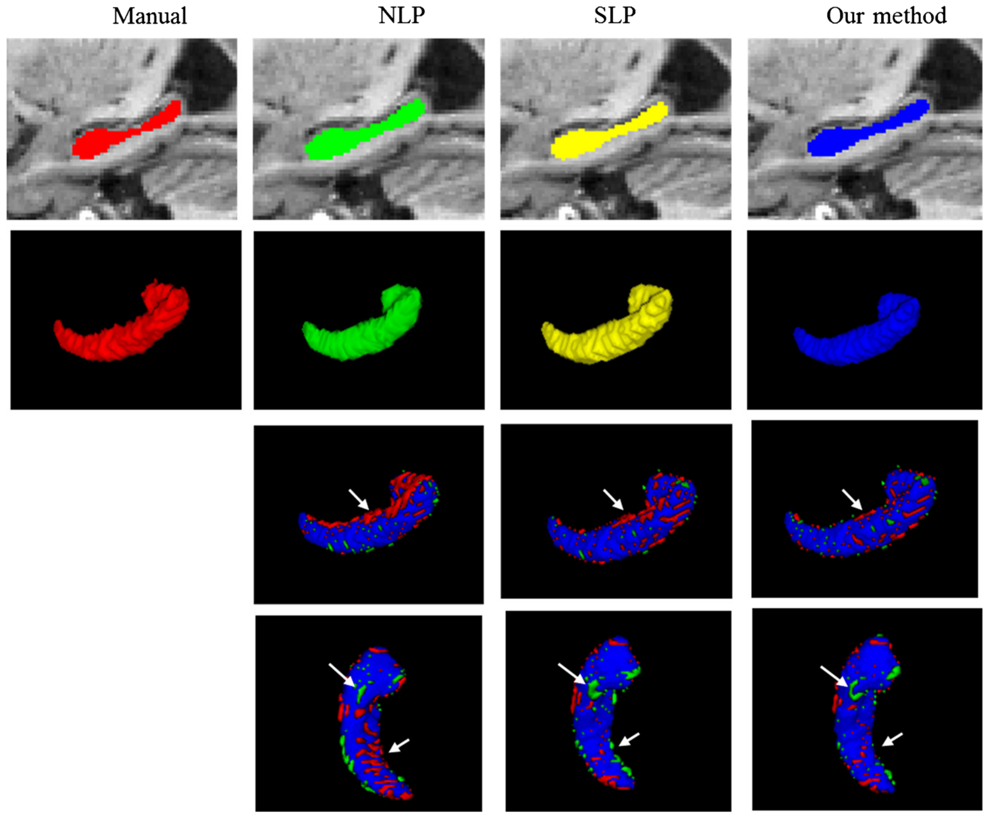


Fig. 10 Hippocampal segmentation results by different methods. The first row shows segmentation results obtained by different methods, the second row presents their corresponding surface rendering results, and the third and fourth row show the comparisons between the results of manual and automatic segmentation methods from different perspectives (blue: overlap between manual and automated segmentation results, red: unidentified voxels, green: miss-identified voxels)

step of 2 for each side. When $1 \times 1 \times 1$ label patch is used, patch-wise label fusion becomes voxel-wise label fusion. Figure 7 shows the mean DSC for our method with different label patch sizes.

As seen in the figure, our method yields the best performance (0.856 for left hippocampus and 0.859 for right hippocampus) when $7 \times 7 \times 7$ label patch is used. Along with an increase in the size of label patch from $1 \times 1 \times 1$ to $7 \times 7 \times 7$, the segmentation accuracy is gradually improved. However, for patch size of $9 \times 9 \times 9$, the segmentation accuracy decreases slightly. Experimental results indicate that using our method, with patch-wise label fusion, can improve segmentation accuracy. Table 3 in "Structured Subspace Learning" section also demonstrates that the Label patch based NN and Label patch based MFA + NN respectively outperform the NN and MFA + NN across all five metrics. This demonstrates the superiority of our method using patch-wise label fusion, compared with the use of voxel-wise label fusion.

The kNN Classifier

For evaluation of the kNN classifier used in our method, we perform our method and also MFA + kNN (without the patch-wise label strategy) using kNN classifier with a different number of nearest neighbors. In this experiment, we set the number of nearest neighbors $k \in \{1, 5, 10, , 20\}$ for the kNN classifier. Figure 8 shows the mean DSC of MFA + kNN and our method using kNN classifier for left and right hippocampus segmentation, plotted against different numbers of nearest neighbors.

It can been seen in Fig. 8 that the highest average DSCs for both the left and right hippocampus are obtained at $k = 5$ and the corresponding ones of MFA + kNN is obtained at $k = 10$. This implies that, compared with the patch-wise label fusion (our method), voxel-wise label fusion (MFA + kNN) requires kNN classifier with more nearest neighbors to obtain the best segmentation performance. In our method using patch-wise label fusion, kNN classifier with a small number of nearest neighbors can yield a high segmentation performance.

Comparison with Conventional Methods

We compared our method with two conventional multi-atlas based segmentation methods, i.e., non-local patch-based label fusion (NPL) (Coupé et al. 2011) and sparse patch-based label fusion (SPL) (Tong et al. 2013). In this experiment, we implemented our method using patch-wise label fusion with MFA-based subspace and kNN classifier. Specifically, we set the number k of nearest neighbors in the kNN classifier to 5, the dimension p of subspace learning using MFA to 150, the intensity patch size to $11 \times 11 \times 11$ and the label patch size for label fusion to $7 \times 7 \times 7$. For MFA, the numbers k_1 and k_2 of the nearest intra-class and inter-class samples were respectively set to 10 and 40. For a fair comparison, we also

Table 4 The mean and standard deviation of DSC, JSC, PI, RI and HD (mm) of left and right hippocampus, produced by NLP, SPL and Our method respectively

	DSC	JSC	PI	RI	RAI	HD
NLP	0.848/0.865 (0.071/0.031)	0.752/0.764 (0.074/0.050)	0.878/0.883 (0.087/0.059)	0.857/0.864 (0.045/0.039)	0.810/0.825 (0.065/0.052)	2.341/2.139 (0.532/0.536)
SPL	0.868/0.880 (0.057/0.033)	0.763/0.778 (0.073/0.050)	0.887/0.878 (0.067/0.044)	0.866/0.875 (0.037/0.039)	0.846/0.852 (0.054/0.039)	1.999/1.796 (0.445/0.388)
Our method	0.879/0.889 (0.047/0.030)	0.773/0.789 (0.064/0.045)	0.903/0.914 (0.066/0.043)	0.879/0.889 (0.033/0.039)	0.851/0.865 (0.042/0.034)	1.857/1.753 (0.369/0.348)

See Table 1 for description of '/'

implemented NPL and SPL using the same patch-wise label fusion strategy used in our method. To take into morphological unity of segments, we also reported one of the structural metrics, i.e. rand index (RAI) (Rand 1971) for quantification. Figure 9 shows box plots of segmentation performance measures (DSC, JSC, PI, RI, RAI and HD) for left and right hippocampus of NLP, SLP, and our method, respectively. For visual inspection, Fig. 10 shows segmentation results for a subject randomly chosen from the dataset. As indicated by the white arrows, it is evident that compared with the NLP and SLP methods, the segmentation obtained by our method is closer to the manual segmentation. This suggests that the proposed method could be a more effective method for hippocampus segmentation.

The mean and standard deviation values for the segmentation performance achieved using NLP, SLP, and our method are reported in Table 4.

As shown in the table, our method obtains the best segmentation performance, followed by the SPL and NPL methods. Compared with the SPL method, our method achieves a significant increase in several metrics. This includes increase of 0.011/0.008 and 0.031/0.024 in the mean DSC, 0.01/0.011 and 0.021/0.025 in the mean JSC, 0.016/0.02 and 0.025/0.031 in the mean PI, and 0.013/0.015, 0.022/0.025 in the mean RI, and 0.005/0.013 in the mean RAI. It also included a decrease of 0.142/0.043 and 0.484/0.386 in the mean HD for the left and right hippocampus segmentation. In terms of standard variance deviation values for the segmentation performance measures, our method also achieves the smallest variance in segmentation accuracy over all test subjects. This indicates that our method is more stable, robust, and reliable than other label fusion methods. Among the compared methods, the NLP method achieved the worst performance due to the lack of discriminating features and highly-correlated candidate patches which repeatedly produced the labeling errors. In terms of average performance for DSC, JSC, PI, RI, and HD, our method obtains statistically significant improvements (p-value<0.001) compared with NLP and SPL. All experimental results indicated that the proposed method performed consistently better than other segmentation methods.

Conclusion

In this paper, we propose a structured discriminant embedding for patch-based label fusion segmentation method to effectively increase the similarity of samples with similar label patches and simultaneously decrease the similarity of samples with dissimilar label patches. Specifically, other than using simple patch intensities to determine the similarity, multi-scale patch intensities and texture information are extracted from image patches to describe appearance information for a voxel. In order to enhance feature discrimination for different local regions in the anatomical structure, MFA is adopted to ensure minimization of the distance between intra-class samples and maximization of the distance between inter-class samples. Based on the extracted features, kNN classifier is employed to determine the final label for the target voxel. Furthermore, to take advantage of anatomical structure information in the segmentation, we extend MFA to the application of patch-wise label fusion, i.e., each voxel in the atlas is assigned with a label patch, rather than a single voxel label for label fusion. Experimental results demonstrate the superior performance of our method for hippocampus segmentation, as compared with traditional multi-atlas based segmentation methods. However, we just conduct our method on hippocampus segmentation in this paper. The proposed method is supposed to be applied in other medical segmentation applications. In the future, we will evaluate our method on segmenting other significant anatomical structures of human brain, such as corpus callosum and amygdala.

Information Sharing Statement

The data used in this paper is downloaded from the publically available ADNI dataset (RRID:SCR_003007) whose website is http://adni.loni.usc.edu/.

The code has been released through the Github https://github.com/wangyan88/Subspace-Learning.

Acknowledgements This work is supported in part by NSFC project 61701324, Sience&Technology Department of Sichuan Province 2016JZ0014, Open Fund Project of Fujian Provincial Key Laboratory of Information Processing and Intelligent Control (Minjiang University) (No. MJUKF201715).

Compliance with Ethical Standards

Conflict of Interest The authors declare no conflict of interest.

Appendix

Given a sampled voxel $\boldsymbol{z}$, the texture information from the image H we extract in this paper includes:

1. Outputs of FODs:

$$\{H(\boldsymbol{z}+\boldsymbol{u})-H(\boldsymbol{z}-\boldsymbol{u}), \boldsymbol{u}=(r\cos\theta\sin\phi, r\sin\theta\sin\phi, r\cos\phi)\}$$

2. Outputs of SODs:

$$\{H(\boldsymbol{z}+\boldsymbol{u})+H(\boldsymbol{z}-\boldsymbol{u})-2H(\boldsymbol{z}), \boldsymbol{u}=(r\cos\theta\sin\phi, r\sin\theta\sin\phi, r\cos\phi)\}$$

3. Outputs of 3D Hyper plan filters:

$$\left\{\Psi_1{}^*\left(H\left(C_{3,3,1}(\boldsymbol{z}+\boldsymbol{u})\right)-H\left(C_{3,3,1}(\boldsymbol{z}-\boldsymbol{u})\right)\right), \boldsymbol{u}=(0,0,1), \Psi_1=\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix}\right\}$$

4. Outputs of 3D Sobel filters:

$$\left\{\Psi_2{}^*\left(H\left(C_{3,3,1}(\boldsymbol{z}+\boldsymbol{u})\right)-H\left(C_{3,3,1}(\boldsymbol{z}-\boldsymbol{u})\right)\right), \boldsymbol{u}=(0,0,1), \Psi_2=\begin{bmatrix}1 & 2 & 1\\ 2 & 3 & 2\\ 1 & 2 & 1\end{bmatrix}\right\}$$

5. Outputs of Laplacian filters:

$$\sum_{z_1\in O_p(z)}\left(H(\boldsymbol{z}_1)-H(\boldsymbol{z})\right), O_p(\boldsymbol{z})\subseteq C_{3,3,3}(\boldsymbol{z})$$

6. Outputs of range difference filters:

$$\max_{z_1\in O_p(z)}\left(H(\boldsymbol{z}_1)\right)-\min_{z_1\in O_p(z)}\left(H(\boldsymbol{z}_1)\right), O_p(\boldsymbol{z})\subseteq C_{3,3,3}(\boldsymbol{z})$$

where $C_{a,\,b,\,c}(\boldsymbol{z})$ represents a cube centered at $\boldsymbol{z}$ with size of $a \times b \times c$, $\boldsymbol{u}$ is the offset vector, r is the length of $\boldsymbol{u}$, θ and ϕ are two rotation angles of $\boldsymbol{u}$, $O_p(\boldsymbol{z})$ denotes the voxels in the p-neighborhood of $\boldsymbol{z}$, * denotes the convolution operation. FODs and SODs detect intensity change along a line segment. Here, we set $r \in \{1, 2, 3\}$, $\theta \in \{0, \pi/4, \pi/2, 3\pi/4\}$, and $\phi \in \{0, \pi/4, \pi/2\}$. 3D Hyperplane filters and 3D Sobel filters are the extensions of FODs and SODs in the plane. Filters along two other directions are also implemented. Laplacian filters are isotropic and detect second-order intensity changes. Range difference filters compute the difference between maximal and minimal values in the neighborhood for each voxel. In this paper, we determine the size of a neighborhood $p \in \{7, 19, 27\}$.

References

Carmichael, O. T., Aizenstein, H. A., Davis, S. W., Becker, J. T., Thompson, P. M., Meltzer, C. C., & Liu, Y. (2005). Atlas-based hippocampus segmentation in Alzheimer's disease and mild cognitive impairment. *NeuroImage, 27*(4), 979–990.

Chen, Z., Jie, B., Liu, M., Chen, S., Shen, D., & Zhang, D. (2015). Label-aligned multi-task feature learning for multimodal classification of Alzheimer’s disease and mild cognitive impairment. *Brain Imaging & Behavior*, 1–12.

Chen, Z., Wang, Z., Zhang, D., Liang, P., Shi, Y., Shen, D., & Wu, G. (2017). Robust multi-atlas label propagation by deep sparse representation. *Pattern Recognition, 63*, 511–517.

Coupé, P., Manjón, J. V., Fonov, V., Pruessner, J., Robles, M., & Collins, D. L. (2011). Patch-based segmentation using expert priors: Application to hippocampus and ventricle segmentation. *NeuroImage, 54*, 940–954.

Dong, P., Wang, L., Lin, W., Shen, D., & Wu, G. (2017). Scalable joint segmentation and registration framework for infant brain images. *Neurocomputing, 229*, 54–62.

He, X., Lum, A., Sharma, M., Brahm, G., Mercado, A., & Li, S. (2017). Automated segmentation and area estimation of neural foramina with boundary regression model. *Pattern Recognition, 63*, 625–641.

Jafari-Khouzani, K., Elisevich, K. V., Patel, S., & Soltanian-Zadeh, H. (2011). Dataset of magnetic resonance images of nonepileptic subjects and temporal lobe epilepsy patients for validation of hippocampal segmentation techniques. *Neuroinformatics, 9*(4), 335–346.

Liao, S., Gao, Y., Lian, J., & Shen, D. (2013). Sparse patch-based label propagation for accurate prostate localization in CT images. *IEEE Transactions on Medical Imaging, 32*, 419–434.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association, 66*(336), 846–850.

Rekik, I., Li, G., Wu, G., Lin, W., & Shen, D. (2015). Prediction of infant MRI appearance and anatomical structure evolution using sparse patch-based metamorphosis learning framework. *In International Workshop on Patch-based Techniques in Medical Imaging* (pp. 197–204). Springer, Cham.

Rincón, M., Díaz-López, E., Selnes, P., Vegge, K., Altmann, M., Fladby, T., et al. (2017). Improved automatic segmentation of white matter hyperintensities in mri based on multilevel lesion features. *Neuroinformatics*, 1–15.

Shen, D. (2007). Image registration by local histogram matching. *Pattern Recognition, 40*, 1161–1172.

Shi, F., Wang, L., Dai, Y., Gilmore, J. H., Lin, W., & Shen, D. (2012). LABEL: Pediatric brain extraction using learning-based meta-algorithm. *NeuroImage, 62*, 1975–1986.

Sundar, H., Litt, H., & Shen, D. (2009). Estimating myocardial motion by 4D image warping. *Pattern Recognition, 42*, 2514–2526.

Tong, T., Wolz, R., Coupé, P., Hajnal, J. V., & Rueckert, D. (2013). Initiative, A. D. N. & others segmentation of MR images via discriminative dictionary learning and sparse coding: Application to hippocampus labeling. *NeuroImage, 76*, 11–23.

Tustison, N. J., Avants, B. B., Cook, P. A., Zheng, Y., Egan, A., Yushkevich, P. A., & Gee, J. C. (2010). N4ITK: Improved N3 bias correction. *IEEE Transactions on Medical Imaging, 29*, 1310–1320.

Wang, H., Suh, J. W., Das, S. R., Pluta, J. B., Craige, C., & Yushkevich, P. A. (2013). Multi-atlas segmentation with joint label fusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*, 611–623.

Wang, Y., Ma, G., An, L., Shi, F., Zhang, P., Wu, X., Zhou, J., & Shen, D. (2016a). Semi-Supervised Tripled Dictionary Learning for Standard-dose PET Image Prediction using Low-dose PET and Multimodal MRI. *IEEE Transactions on Biomedical Engineering*, 1–1.

Wang, Y., Zhang, P., An, L., Ma, G., Kang, J., Shi, F., Wu, X., Zhou, J., Lalush, D. S., Lin, W., et al. (2016b). Predicting standard-dose PET image from low-dose PET and multimodal MR images using mapping-based sparse representation. *Physics in Medicine and Biology, 61*, 791.

Wu, G., Wang, Q., Lian, J., & Shen, D. (2013). Estimating the 4D respiratory lung motion by spatiotemporal registration and super-resolution image reconstruction. *Medical Physics, 40*(3), 532–539.

Wu, G., Wang, Q., Zhang, D., Nie, F., Huang, H., & Shen, D. (2015a). A generative probability model of joint label fusion for multi-atlas based brain segmentation. *Med Image Anal, 18*(6), 881.

Wu, G., Kim, M., Wang, Q., Munsell, B. C., & Shen, D. (2015b). Scalable high-performance image registration framework by unsupervised deep feature representations learning. *Deep Learning for Medical Image Analysis, 63*(7), 1505–1516.

Wu, G., Kim, M., Sanroma, G., Qian, W., Munsell, B. C., & Shen, D. (2015c). Hierarchical multi-atlas label fusion with multi-scale feature representation and label-specific patch partition. *NeuroImage, 106*, 34–46.

Zarei, M., Beckmann, C. F., Binnewijzend, M. A., Schoonheim, M. M., Oghabian, M. A., Sanz-Arigita, E. J., Scheltens, P., Matthews, P. M., & Barkhof, F. (2013). Functional segmentation of the hippocampus in the healthy human brain and in Alzheimer's disease. *NeuroImage, 66*, 28–35.

Zhou, L., Wang, L., & Ogunbona, P. (2014). Discriminative sparse inverse covariance matrix: Application in brain functional network classification. *Computer Vision and Pattern Recognition*, 3097–3104.

Zhou, L., Wang, L., Liu, L., Ogunbona, P., & Shen, D. (2016). Learning discriminative bayesian networks from high-dimensional continuous neuroimaging data. *IEEE Transactions on Pattern Analysis & Machine Intelligence, 38*(11), 2269–2283.

Zhu, H., Cheng, H., Yang, X., & Fan, Y. (2017). Metric learning for multi-atlas based segmentation of hippocampus. *Neuroinformatics, 15*(1), 41–50.

Zu, C., Wang, Z., Zhang, D., Liang, P., Shi, Y., Shen, D., & Wu, G. (2017). Robust multi-atlas label propagation by deep sparse representation. *Pattern Recogn, 63*, 511–517.