Clustering-based unsupervised brain MRI segmentation

Sarah Xu sarahxu@stanford.edu
Yuxin Hu yuxinh@stanford.edu
Zhaotian Fang z23fang@stanford.edu

Abstract

Supervised approaches to semantic segmentation of brain magnetic resonance imaging (MRI) has two main flaws. One is that labeling is slow and expensive since it must be done by radiologist with specialized skills. The other flaw is that models trained on MRIs produced by one machine do not generalize well to MRIs from other machines. We propose an unsupervised approach to brain MRI segmentation based on k-means clustering of pixel features. We investigate and experiment between different classes of methods for pixel feature extraction: heuristic methods based on patches, and learned methods based on neural networks. We evaluate our algorithm on brain images from the Human Connectome Project (HCP) public dataset.

Our best performing model achieves a DICE score of 0.8 on grey and white matter. We also perform some qualitative analysis on the segmentation results and study the effect different hyperparameters have on the characterstics of the segmentation.

1 Introduction

Magnetic resonance imaging (MRI), as a non-invasive and non-radiative modality, has been used in both clinical applications (e.g., tumor detection) and neuroscientific research (e.g., fiber tracking). For brain MRI, segmentation of tissues and tumors is a challenging task. Traditionally, it requires slow and manual annotation from radiologists with specialized expertise. As a result, there has been a push in efforts towards automating and accelerating the process. The most common automated method for brain neuroanatomy segmentation is a software tool named FreeSurfer. [1] However, this tool is computationally expensive, taking several hours to process one brain.

Deep learning approaches, specifically using convolutional neural networks, has achieved state-of-the-art results in several image-based tasks including image segmentation. These methods are usually trained based on huge amounts of labeled data. However, they do not transfer well to the medical imaging domain since the collection and labeling of a dataset large enough for training deep neural networks can be costly or even impossible.

Due to the difficulty in labeling, some efforts have focused on applying unsupervised learning for brain MRI segmentation. Generative models such as the AnoGAN framework and Variational Autoencoders (VAE) were trained purely on healthy brain images, assuming that the network would be able to detect the abnormal regions. [2, 3] A derivative work combines AnoGAN and VAE into AnoVAEGAN, used to segment brain tumors. However, these methods still need large sets of healthy images, and can only be used for rough tumor segmentation, since some high-resolution details are missed. Other unsupervised methods leverage clustering techniques. Moriya et al. extend JULE to handle 3D sub-volumes from micro-CT data of lungs. [4] Using a 3D CNN, they extract a feature vector from each patch and cluster the vectors using K-means. Using a similar approach, Moriya et al. also experimented with using spherical K-means to segment lung cancer in pathology images. [5] In this work, we want to achieve brain semantic segmentation without the need of labeled data.

Furthermore, our project provides a positive impact on society as it facilitates early-stage detection and diagnosis of brain-related diseases. Compared to the traditional segmentation technique of manual labeling by radiologists, unsupervised segmentation helps save some of the doctors time and also reduces the human cost associated with generating semantic segmentation results for brain MRIs. Driving down the cost of brain MRI segmentation will in turn make medical treatment more accessible and affordable for patients. By utilizing auto-

mated segmentation techniques, turnaround times for diagnosis of brain diseases will be drastically reduced, which may potentially save the lives of some patients.

2 Methods

For unsupervised brain MRI segmentation, we propose a segmentation pipeline based on K-means and various convolution neural network (CNN) based feature extractors.

2.1 K-means

K-means clustering is an unsupervised technique to cluster n observations into k clusters such that each observation belongs to the cluster with the nearest centroid. [6] Suppose the n observations are partitioned into k sets $S_1, ..., S_k$ where μ_i is the mean of observations in set S_i, then the objective is to minimize the sum of the variances in each set

$$\min_{S} \sum_{i=1}^{k} \sum_{x \in S_i} ||x - \mu_i||^2 \quad (1)$$

The K-means approach to unsupervised segmentation is described below and shown in Figure 1. Given an input image of size $D \times H \times W$ (depth, width, height), the algorithm is

1. Extract pixel-wise features from the image to get a segmentation mask of size $C \times H \times W$.
2. Run K-means clustering on the HW C-dimensional pixel features while setting $K = 5$.
3. After convergence, assign each pixel the class associated with the cluster the pixel feature is assigned to.

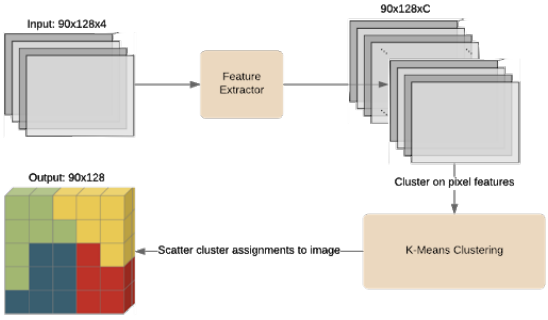

Figure 1: High level block diagram of the K-means approach to unsupervised segmentation assuming an input size of $90 \times 128 \times 4$ for the brain MRI.

2.2 Feature Extraction

Before applying K-means, we need to extract image features to run the clustering algorithm on. We have tried a total of three different classes of approaches.

Contrast features As shown in Figure 2, for each pixel, we use the four contrasts from the original input image.

Patch features As shown in Figure 2, for each pixel, we collect and flatten the contrast values from a square patch (e.g., 3×3) centered on that pixel to get the feature ($3 \times 3 \times 4 = 36$ features for a 3×3 patch).

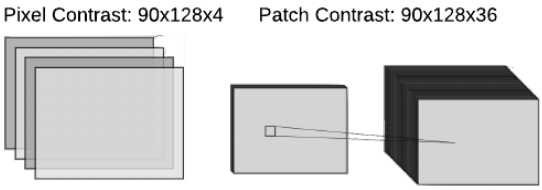

Figure 2: Illustration of contrast and patch feature extraction.

CNN features We apply transfer learning by using the image features extracted from different pretrained CNNs. The procedure for this approach is illustrated in Figure 3 and outlined below:

1. Use bilinear interpolation to upsample the input image to 224×224.
2. Use the first three contrasts of the upsampled image as the input tensor ($3 \times 224 \times 224$). Feed the input tensor into the network.
3. As the input passes through the network, its spatial dimension decreases while its channel dimension increases. The features become more high-level and domain-specific as you go deeper into the network. For this application, take an intermediate output after a few downsample operations to get a feature map of approximately $C \times 56 \times 56$ where $C >> 3$.
4. Using Principle Component Analysis (PCA), we select the top 5 most useful features from the C channels, creating a feature map of size $5 \times 56 \times 56$. [7] The motivation for dimensionality reduction is that k-means suffers from the curse of dimensionality. For features with higher dimension, points become more equidistant between each other, making

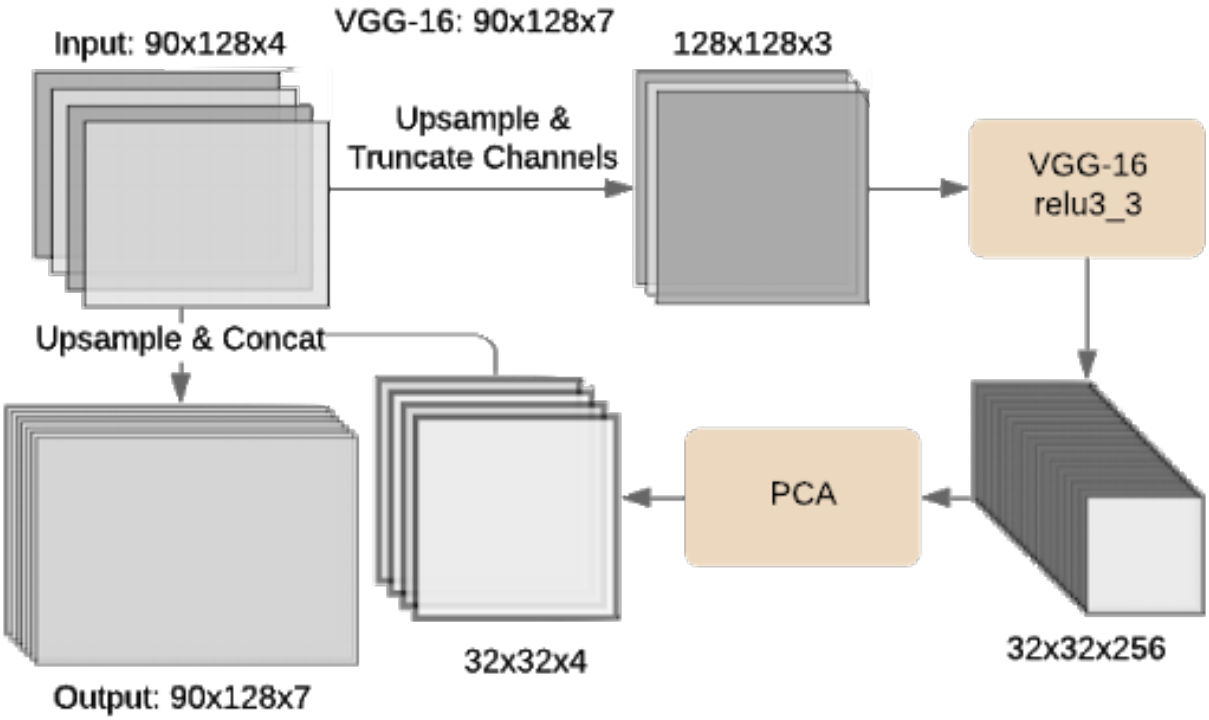

Figure 3: High level block diagram of the feature extraction process for a pre-trained VGG-16 network.

it harder for the algorithm to discriminate between different clusters.

5. Use bilinear interpolation to upsample the feature map to the original spatial dimension of the input image to create a feature map of size $5 \times 224 \times 224$.

6. Concatenate the upsampled feature map with the input image, giving you 9 features per pixel.

2.2.1 ILSVRC Networks

ILSVRC (ImageNet Large Scale Visual Recognition Challenge) is a premier competition in computer vision research held yearly for the tasks of image classification and object detection. [8] ILSVRC uses ImageNet as a benchmark dataset, one of the largest labelled datasets of natural images with a rich class taxonomy. [9] The most successful models originating from ILSVRC have become the standard in transfer learning based approaches for image-based tasks. The models used for brain MRI segmentation are described below.

VGG VGG is a CNN model developed by researchers from the Visual Geometry Group at Oxford. [10] It was one of the top performing models at ILSVRC 2014 in the tasks of image classification and localization. [8] The novelty of a VGG network is in using smaller 3x3 filters to build a deeper network with the same receptive field as a shallower network with larger filters while using less weights. We experimented with both the VGG-16 and VGG-19 variants, which consists of 13 and 16 convolutional layers respectively, followed by 3 fully connected layers.

ResNet ResNet, winner of ILSVRC 2015, improves upon previous models by introducing residual blocks, which learns the "difference" between the input and output features. This new shift in CNN architecture allows for training even deeper networks without running into the vanishing gradient problem. [11] We used the ResNet101 and ResNet152 variants, which consist of 101 and 152 layers before the classification block. These ResNet variants use a "bottleneck" version of a residual block, which adds pointwise convolutions before and after the main 3x3 convolution to reduce the number of parameters without losing representational power.

DenseNet DenseNet improves upon ResNet architectures by introducing dense blocks, where each feature map has skip connections to all previous feature maps and feature maps are con-

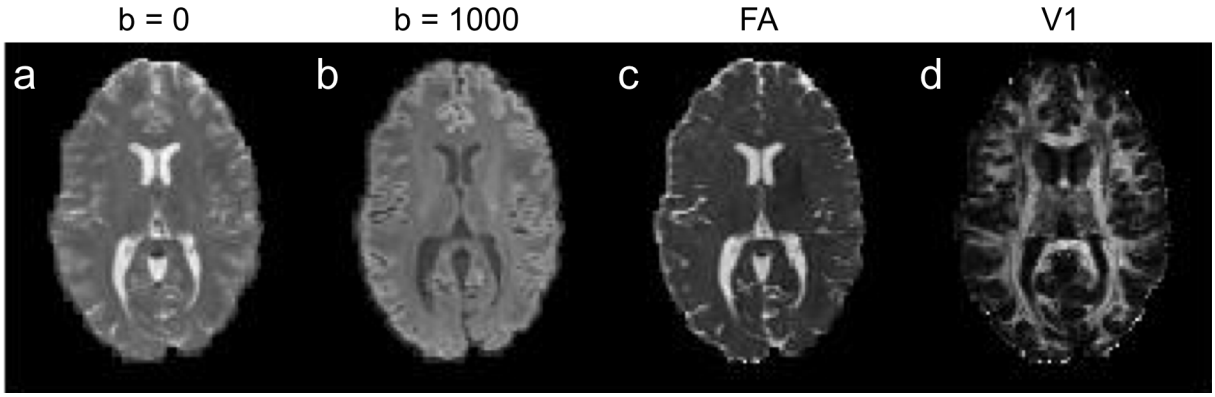

Figure 4: One representative example of preprocessed images from the HCP dataset (a: averaged b=0 image, b: averaged b=1000 image, c/d: FA /V1 maps derived from all b=1000 images).

catenated instead of summed together. [12] The growth behavior in the channel dimension within a dense block reduces the number of parameters compared to residual blocks while improving predictive accuracy. We tried both the DenseNet161 and DenseNet201 variants, which both have four dense blocks consisting of convolutions.

2.2.2 U-Net

The U-Net is a convolutional network architectures that has been widely utilized in supervised learning for medical image segmentation. [13, 14] A U-Net architecture consists of an encoder network and decoder network with skip connections in intermediate layers from the encoder to decoder. In this work, we start from a U-Net with a pre-trained ResNet-18 encoder. Then we fine-tune the network with half of our data (49 subjects with 90 slices each). Finally we use the fine-tuned network as a feature extractor. The network has five layers for the encoder and decoder respectively.

2.3 Evaluation

The evaluation of unsupervised segmentation results of MR images is an intrinsically complex problem and worth a separate research effort to provide a comprehensive review of the methods. [15] In this project, we provide two metrics for evaluating the quality of segmentation results.

The first metric is considered traditionally to be the only authentic way of validating real patient MRI data. It involves surveying experience radiologists to gather qualitative ratings on the segmentation results relative to the ones annotated manually. However, the manual segmentation is considered to be prone to errors, difficult to reproduce even by the same expert, and obtaining the manual segmentation will dramatically increase the scope of this project. [16] As a result, we invited a neuroscientist to comment on the structures captured by our segmentation method and the anatomical accuracy.

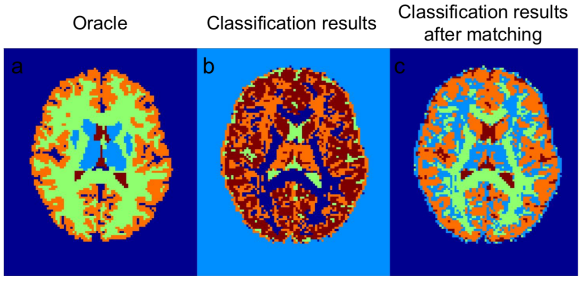

Figure 5: An example showing the reassigning results (c) with the Hungarian matching algorithm to match the target segmentation labels (a).

The second is a quantitative calculation of the per-pixel accuracy of the segmentation output for our proposed method. The ground truth we used is the FreeSurfer segmentation result that is considered as the oracle for this project. According to the FreeSurfer documentation, there are 1292 different label classes with label numbers ranging from 0 to 14175. [17] To make the prediction matching problem more straightforward, we preprocess the label data to the actual 114 classes that are present in the HCP dataset and relabeled them with numbers 0 - 113. The labels are further categorized into 5 classes: brain mask, white Matter, grey Matter, deep grey Matter, and cerebrospinal fluid (CSF). To match the predicted segmentation labels with the target ones, we use the Hungarian method to find a match between the predicted value and the target label of each pixel. [18] One example of match-

ing result is shown in Figure 5. The accuracy of the segmentation result is then evaluated based on the pixel accuracy ratio between the prediction and ground truth and a DICE score for each of the five classes respectively.

We use X to represent the prediction result and Y to represent the target segmentation. The pixel accuracy is calculated as

$$\frac{|X \cap Y|}{|Y|}$$

The DICE score is calculated for all $k = 1, \ldots 5$,

$$\frac{2|X_k \cap Y_k|}{|X_k| + |Y_k|}$$

where k is the class of interest.

3 Dataset

Currently, we are working on data from the Human Connectome Project (HCP) WU-Minn-Ox Consortium public database (https://www.humanconnectome.org). The raw data were acquired on a customized 3-Tesla (3T) Skyra scanner (Siemens Healthcare, Erlangen, Germany) equipped with a 32-channel head coil. Diffusion data were acquired along 90 uniformly sampled diffusion-encoding directions at b = 1,000 s/mm^2 and 18 interspersed b = 0 volumes using a 2D single-refocused Stejskal-Tanner diffusion-weighted spin-echo EPI sequence with the following acquisition parameters: 90 axial slices, slice thickness = 1.25 mm, resolution = 1.25 mm isotropic.

Data of 99 subjects were processed and used. Diffusion-weighted images along different diffusion-weighted directions were corrected for eddy current distortion and bulk motion, and co-registered using the "eddy" function from the FMRIB Software Library (FSL, https://fsl.fmrib.ox.ac.uk/fsl). The diffusion tensor imaging (DTI) model was fitted using FSL's "dtifit" function to derive the fractional anisotropy (FA) and the primary eigenvector (V1). All 90 diffusion-weighted (b=1000) images and 18 non-diffusion-weighted (b=0) images were then averaged. Finally, we got four images with different contrasts as shown in Figure 4.

4 Results

4.1 Baseline and Oracle

For the baseline of the project, the anatomy segmentation is obtained using a unsupervised segmentation method called mean-shift clustering. This method perform iterative mean-shift algorithm at each pixel until they converge to a set of clusters. [19] For the example shown in image (b) of Figure 6, the brain image is segmented into 17 clusters.

We use the FreeSurfer segmentation results ((b) in Figure 6) as the oracle for neuroanatomy segmentation on MR images, which classifies the brain regions into labels ranging from 0 to 14175. [17] The labels are then classified into five classes using the method mentioned before.

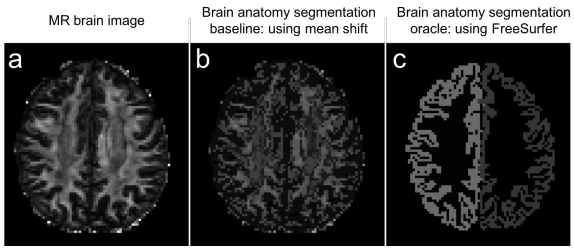

Figure 6: Example anatomy segmentation of brain MRI with mean shift (b: baseline) and FreeSurfer (c: oracle). The different segmentation intensity levels represent different classes.

4.2 Comparing different patch sizes

Figure 7 shows the segmentation results while varying the patch size. For small patch sizes, we see that the granularity of the segmentation is very fine-grained. More specifically, we can see the branching structure (orange color in (b)) which appears in the oracle's segmentation too. In addition, we see a lot of noise in the segmentation (orange dots in (b)). As the patch size increases, the finer branches in the segmentation merge into blobs (blue color in (d)). In addition, the amount of noisy predictions is reduced (no more orange dots on red segmentation like in (b)).

4.3 Influence of including CNN features

In Figure 9, we compare the clustering results with and without using the features from pretrained VGG network. For a patch size of one, if we compare against the oracle's result (a), excluding VGG features causes many misclassifications (orange and red in (b)). When including VGG features, it not only makes the segmentation less noisy, but it removes most of the misclassifications too (image e).

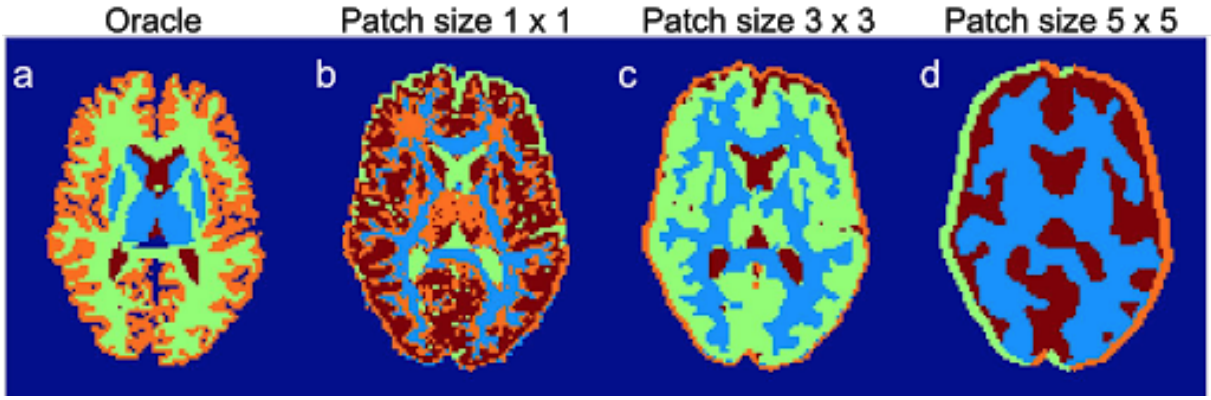

Figure 7: A comparison of segmentation results using different patch sizes (b-d) for a single brain MRI slice.

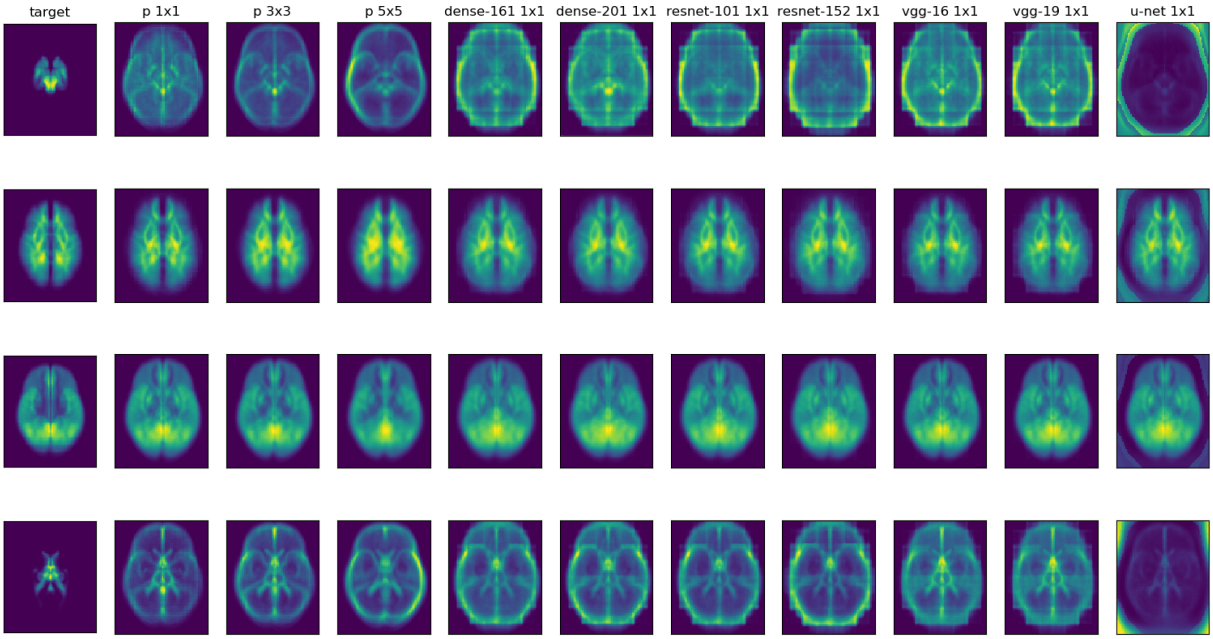

Figure 8: A comparison of averaged target segmentation label heat map of all slices across all subjects with the predictions from different feature extractors. Row 1 is the deep grey matter, row 2 is the white matter, row 3 is the grey matter, row 4 is the CSF.

4.4 Comparing different CNNs for feature extraction

Inspired by the results shown in Figure 9, we perform a similar analysis by taking the output of some intermediate layer for several other pretrained networks visualized them in a similar fashion, shown in Figure 10. All those methods show consistent results for segmentation of grey matter and white matter (yellow and light blue in (a)). However, for deep grey matter (dark blue region indicated by the black arrow in (a)), the results are not as good. Although the extent of the deep grey matter region is discriminated well, many of the pixels within the region are misclassified as other tissues.

Figure 8 shows the segmentation class probabilities averaged over all slices and cases to provide an intuition of the distributions of predictions for different feature extractors. For deep grey matter, we see the general structure is captured better by DenseNet-201 and VGG networks. For CSF, all feature extractors contain brain mask false positives but still manage to also capture the CSF structure. For grey and white matter, all feature extractors performed well on capturing the general distribution.

Table 1 summarizes the DICE scores and pixel accuracy ratios over all cases and slices using different methods. The DICE score is calculated for each class of interest (grey matter, white matter,

	Grey matter	White matter	Deep grey matter	CSF	Brain mask	Pixel accuracy
Patch size: 1x1	0.78 ± 0.09	$\mathbf{0.79 \pm 0.04}$	0.05 ± 0.08	0.22 ± 0.26	$\mathbf{0.94 \pm 0.01}$	0.84 ± 0.04
Patch size: 3x3	0.80 ± 0.10	0.75 ± 0.09	0.03 ± 0.06	0.20 ± 0.28	$\mathbf{0.94 \pm 0.01}$	0.84 ± 0.04
Patch size: 5x5	0.74 ± 0.15	0.67 ± 0.14	0.01 ± 0.04	0.16 ± 0.27	0.93 ± 0.01	0.81 ± 0.05
DenseNet-161	0.80 ± 0.08	0.70 ± 0.12	0.02 ± 0.07	0.23 ± 0.27	0.82 ± 0.07	0.73 ± 0.06
DenseNet-201	0.78 ± 0.07	0.67 ± 0.10	0.04 ± 0.08	0.25 ± 0.28	0.86 ± 0.07	0.74 ± 0.06
ResNet-101	0.81 ± 0.07	0.73 ± 0.12	0.01 ± 0.04	0.25 ± 0.26	0.86 ± 0.04	0.77 ± 0.06
ResNet-152	$\mathbf{0.82 \pm 0.12}$	0.72 ± 0.20	0.01 ± 0.01	0.23 ± 0.26	0.75 ± 0.06	0.68 ± 0.04
VGG-16	0.80 ± 0.09	0.78 ± 0.09	0.03 ± 0.06	$\mathbf{0.29 \pm 0.28}$	0.91 ± 0.05	0.82 ± 0.05
VGG-19	0.80 ± 0.10	0.78 ± 0.08	0.02 ± 0.06	$\mathbf{0.29 \pm 0.28}$	0.91 ± 0.04	0.72 ± 0.05
U-Net	0.81 ± 0.11	0.72 ± 0.20	0.01 ± 0.03	0.24 ± 0.27	0.78 ± 0.08	0.70 ± 0.08

Table 1: DICE score and pixel accuracy (last column) results for models with different feature extractors across all the classes of interest.

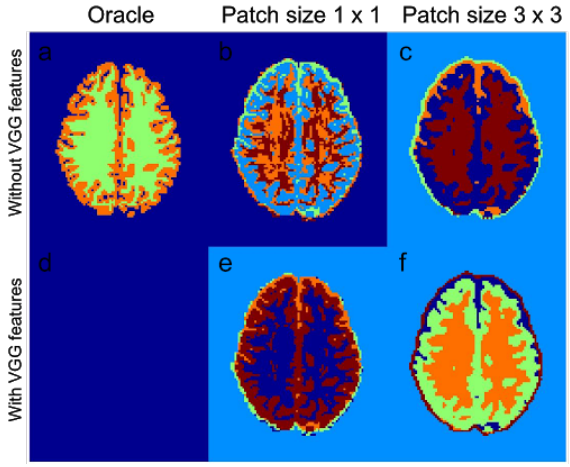

Figure 9: A comparison of segmentation results between including VGG features (e, f) or not (b, c) for a single brain MRI slice. Two different patch sizes are used.

deep grey matter, CSF). Based on this table, using only the contrasts from the preprocessed image as features shows good initial results. When including the features from pretrained networks such as ResNet-101 and ResNet-152, the DICE scores for some tissues (grey matter) improve while others regress (white matter). All those methods show pretty low DICE scores for the deep grey matter. In addition, including network features lead to worse brain mask segmentation compared with using patch features only. Looking at Figure 10, the brain boundaries are often noisy and overestimated (especially in (e) for U-Net), which explains why the DICE scores for the brain mask are lower compared to patch-based methods.

5 Discussion

Patch sizes The choice of patch size has a significant effect on characteristics of the segmentation, influencing both the amount of noise and detail in the result. Patch size is a hyperparameter of our model that should be tweaked to balance the trade-off between noise and structural detail. More specifically, as we increase the patch size, we trade away structural detail in our segmentation in favor of less noise in our predictions. The optimal hyperparameter setting is affected by a number of different factors. For example, as the image resolution increases, the patch size must increase such that its receptive field does not change in world space. In addition, the optimal patch size may depend on the patch's location within the image. With domain expertise, if all images follow a certain structure, then the patch size can be optimized for different regions in the structure based on whether less noise or more detail is preferred. From visually inspecting the results in Figure 7, a patch size of three strikes a good balance between noise and detail for our data.

Pretrained networks Various pretrained CNN-based models are used as feature extractors. Results from Figure 10 show that these models perform well on grey and white matter. In terms of deep grey matter (dark blue region indicated by the dark arrow in (a)), the spatial extent of the deep grey matter region has been discriminated well by CNNs, but they are usually misclassified into other tissues. We attribute this behavior to two reasons: the features do not contain enough structural detail, or the portion of deep grey matter is too small and hard to segment realistically.

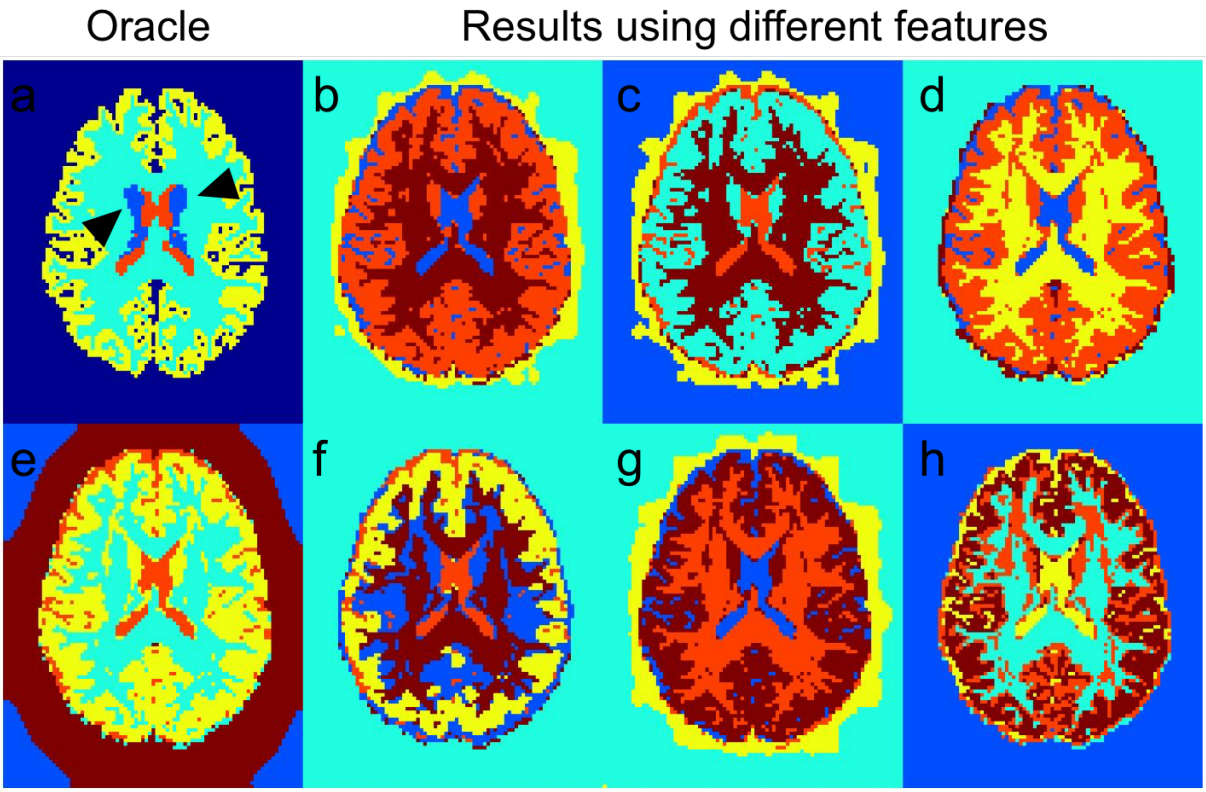

Figure 10: A comparison of segmentation results using features from different pretrained networks for a single brain MRI slice, where (a) is from FreeSurfer as oracle, (b-h) uses features from DenseNet-161, ResNet-101, VGG-16, U-Net, DenseNet-201, ResNet-152, VGG-19, respectively.

Structural information As we can observe in Table 1 and Figure 8, our approach works relatively well on white matter and grey matter. These classes are simpler to segment since it does not require as much domain knowledge about brain structure compared to other classes like deep grey matter and Cerebrospinal fluid (CSF).

In Figure 8, we see the deep grey matter structure shown in row 1 column 1 appears in the VGG feature extractor prediction. In addition, the CSF structure in row 4 column 1 appears in almost all segmentation results. However, we can see that the amount of noises in the deep grey matter segmentation is immense and could be further improved with various localization methods that filters out regions based on the distribution of the tissues observed.

3D volumes Currently, our method does segmentation on 2D slices of the brain. To obtain more structural information and improve on the current results, we can experiment with using features extracted from 3D volumes as opposed to 2D slices. Furthermore, a 3D matching of labels between target and predictions could potentially further improve on the results.

Combining features Since the k-means algorithm uses the Euclidean distance metric, it implicitly weighs each feature equally. In future work, we can explore distance metrics that can weigh each feature differently. There also exists several hyperparameters in the post-processing of the network features such as the number of components in PCA and the normalization techniques of the features before feeding them into K-means. Fine-tuning those hyperparameters may improve the performance.

Evaluation metrics Currently, we are using DICE score and per-pixel accuracy as quantitative measurements of the segmentation results. However, they are not necessarily informative when it comes to evaluating the actual segmentation performance. For example, if we compare the CSF evaluation in DICE score in Table 1, the CSF scores for DenseNets and ResNets is higher than that of patch size 1x1 and patch size 3x3. However, by visually inspecting the heat map of CSF in Figure 5, patch size 1x1 and patch size 3x3 captures stronger structural information of CSF than DenseNets and ResNets. Finding metrics that better evaluates the segmentation results is still an active research area.

Domain Shift One potential issue that may negatively influence the performance for models using pretrained network feature extractors is domain shift. More specifically, the types of images used in pretraining and the types of images used in our model differs greatly. Its unclear how well the weights in the network that are optimized for ImageNet's natural images transfer over to brain MRIs. A potential solution to this issue is to fine-tune the network on brain MRI data. More specifically, the last few layers and BatchNorm layers would be unfrozen while all other layers are kept frozen. This allows the network to learn higher level features as well as the optimal normalization parameters in the BatchNorm layers specific to brain MRI data.

6 Conclusion

A pipeline is developed for unsupervised brain MRI segmentation, including data preprocessing, feature extraction, clustering based segmentation, model evaluation, and segmentation analysis. Dice score of 0.8 is achieved for segmentation of grey matter and white matter, and our results also demonstrate that including features from CNNs could help improve the classification accuracy. More advanced methods related to extracting and combining features may be explored for the future work.

Acknowledgements We would like to thank Dr. Qiyuan Tian, Chuanbo Pan, and Prof. Brian Hargreaves for helpful discussions over the course of this project.

References

[1] Bruce Fischl, David Salat, Evelina Busa, Marilyn Albert, Megan Dieterich, Christian Haselgrove, Andre Kouwe, Ron Killiany, David Kennedy, Shuna Klaveness, Albert Montillo, Nikos Makris, Bruce Rosen, and Anders Dale. Whole brain segmentation: Automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33:341–55, 02 2002.

[2] Jinwon An and Sungzoon Cho. Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, 2(1), 2015.

[3] Michał Januszewsk. Segmentation-Enhanced CycleGAN. V(19):1–11, 2019.

[4] Takayasu Moriya, Holger R. Roth, Shota Nakamura, Hirohisa Oda, Kai Nagara, Masahiro Oda, and Kensaku Mori. Unsupervised segmentation of 3d medical images based on clustering and deep representation learning. *Medical Imaging 2018: Biomedical Applications in Molecular, Structural, and Functional Imaging*, Mar 2018.

[5] Takayasu Moriya, Holger R. Roth, Shota Nakamura, Hirohisa Oda, Kai Nagara, Masahiro Oda, and Kensaku Mori. Unsupervised pathology image segmentation using representation learning with spherical k-means. *Medical Imaging 2018: Digital Pathology*, Mar 2018.

[6] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, March 1982.

[7] Hervé Abdi and Lynne J. Williams. Principal component analysis. *WIREs Comput. Stat.*, 2(4):433–459, July 2010.

[8] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[9] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, June 2009.

[10] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[12] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2016.

[13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[14] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[15] Ivana Despotović, Bart Goossens, and Wilfried Philips. Mri segmentation of the human brain: challenges, methods, and applications. *Computational and mathematical methods in medicine*, 2015, 2015.

[16] William R Crum, Oscar Camara, and Derek LG Hill. Generalized overlap measures for evaluation and validation in medical image analysis. *IEEE transactions on medical imaging*, 25(11):1451–1461, 2006.

[17] https://surfer.nmr.mgh.harvard.edu/fswiki/fstutorial/anatomicalroi/freesurfercolorlut, 2017.

[18] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[19] Dorin Comaniciu and Peter Meer. Mean shift analysis and applications. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1197–1203. IEEE, 1999.

[20] Xu Ji, João F. Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation, 2018.

A Supplemental Material

A.1 Code

The code is hosted on GitHub at https://github.com/ysx001/cs221-project

A.2 Data

The processed dataset is located on a lab server. An example of the data for one subject is at https://drive.google.com/drive/folders/1nbeTn4zNrVG4dRlF3fLDZNhjUA2UCBRh?usp=sharing

A.3 CodaLab

The CodaLab worksheet can be found at https://worksheets.codalab.org/worksheets/0x041a17a82b8249a0bd7c128d51e8bafa

B Next steps

We have done modifications on the IIC code to utilize it for the MRI data, and trained network for the newly adapted datasets. However, there are detailed image transformations and evaluations that we need to modify further to have IIC better adapted for the MRI segmentation purpose, whether it's combining the two or using IIC as one of the features for k-means. We are exploring and experimenting with other feature extraction methods instead of manual extraction or pre-trained VGG for the k-means algorithm. Furthermore, we need to investigate the effects of different weighting on different features and how to choose and weight the extracted features. We have mentioned some methods to measure our segmentation results quantitatively. In the next step, we are also focusing on comparing and evaluating different methods using the metrics proposed.

B.1 IIC

Recently, Ji et al. proposed a method called IIC, which achieved state-of-the-art results for unsupervised image classification and segmentation in a wide variety of image domains. [20] The main contribution of the paper is the IIC objective, which aims to maximize the mutual information between two pairs of image features $\Phi(\mathbf{x}), \Phi(g\mathbf{x})$. In the equation, $\mathbf{x}$ is some input image/patch, g is some perturbation matrix. Φ is a CNN whose output is a segmentation map over the input image.

$$\max_{\Phi} I(\Phi(\mathbf{x}), \Phi(g\mathbf{x})) \tag{2}$$

The intuition behind maximizing mutual information is that the conditional distributions on cluster assignments for $\mathbf{x}, g\mathbf{x}$ should be dependent.

In the original paper, IIC has shown great results on natural images (COCO-Stuff) and satellite imagery (Postdam-3). We intend on adapting IIC for brain MRIs.