3T CSFn synthesis from 7T WMn volumes

Matthew Radovan
Stanford University
mradovan@stanford.edu

1 Preprocessing

CSFn volumes were registered to WMn volumes using the mutual information metric, resulting in 120 pairs of volumes of dimension 256x440x256. The CSFn → WMn registration direction was chosen because the WMn volumes used as input during usage will be unregistered as well.

2 Architectures

2.1 Generator

The generator architecture was a U-net-style encoder-decoder according to Umapathy et al. [1]. All convolution layers had filters of size (3, 3, 3) and were zero-padded to retain the image shape. Two max pooling layers and two transposed convolution layers were used for a final output of size (256, 256, 5).

In the case of the single slice output, a single convolution layer with a (1, 1, 5) kernel and without padding was inserted prior to the 1x1x1 output convolution layer for a final output of size (256, 256).

2.2 Discriminator

The discriminator architecture was inspired by the encoder portion of the generator. Because having the output shape match the input shape was no longer necessary, zero-padding was no longer used in convolution layers. Additionally, everything after the dropout layer of the encoder was removed. And, instead, one max pooling operation followed by two convolution-batchnorm-ReLU blocks were added. Then, the output of this final ReLU layer was flattened, passed through a dense layer of size 256, and then finally passed to a single output with sigmoid activation.

2.2.1 Motivation

Perceptual loss as used in Dar er al. [2] uses VGG16, which was trained on ImageNet and thus expects a 3-channel RGB image as input. However, because our output image contains 5 slices that are not easily analogous to channels, we are presented with a difficulty in using VGG16's features. One possibility, discussed in further detail in a later section, is using features from each slice separately. However, this still does not guarantee that VGG16's features are fully relevant to the relevant features in a brain MRI. So, the motivation for the discriminator came from the idea that it may be possible to create a more useful perceptual loss by training a model to learn brain MRI features directly. This discriminator may also prove useful when using a GAN architecture.

3 Training

Given 120 paired (WMn, CSFn) volumes, 100 volumes were chosen to be the training set and 20 to be the test set. Because 5-slice slabs were input to all models and volumes were of size (256, 440, 256), there were $(440 - 5 + 1) * 100 = 43600$ unique inputs in the training set. However, no model was trained for an entire epoch due to time constraints. It is highly likely that drastically reducing the training set will not significantly affect the results based on qualitative experiments with training sets comprised of only 10 volumes.

3.1 Discriminator

The initial training data set had unaltered CSFn slabs as positive inputs (labeled 1) and corrupted CSFn slabs as negative inputs (labeled 0). The corruption was performed by taking a 16x16x3 patch within the slab (0.234375% of the full slab) and randomly scrambling the pixels within that patch. However, although training approached an accuracy of 1, the discriminator predicted positive labels for WMn inputs as well as uncorrupted CSFn labels.

So, unaltered WMn slabs were added to the training dataset as negative examples (label 0). The model's training accuracy after adding these examples still was able to reach 1.

3.2 Generator

As a baseline, only voxelwise mean absolute error (MAE) was used to train the generator. This produced fairly faithful, though fuzzy, images.

VGG16 perceptual loss was added to the voxelwise MAE and used similarly to [2]. Because VGG16 expects 3-channel RGB images but the generator outputs 5 slices that are not analogous to channels, each slice was separated and stacked to get 5 256x256x3 images for VGG16 to preprocess. Then, feature maps were extracted, the Gram matrices were calculated, and the VGG16 perceptual loss was defined as the mean square error (MSE) between the Gram matrices. Weighting between the perceptual loss and the voxelwise MAE was experimented with, but no firm conclusions were drawn.

However, the inclusion of the VGG16 perceptual loss tended to cause poor qualitative performance. So, another approach was attempted by extracting the outputs of pretrained discriminator's second-to-last layer (i.e. the first fully-connected layer) and calculating the MSE between the values of this layer when the actual and predicted slabs were input to the network. This MSE was added to the voxelwise MAE. Because the discriminator was trained on slabs rather than single slices, this approach has not yet been attempted on the single slice output.

4 Evaluation

Because most iterations of the model generated five-slice slabs, two approaches for full-volume conversion were considered: concatenating non-overlapping slabs and averaging all possible slabs. Concatenating non-overlapping slabs tended to create discontinuities at the boundary between slabs; however, averaging slabs may contribute to the fuzziness observed in the MAE-only approach.

When considering the model that generated the single slice, every possible slab was input into the model and the output slices were combined. The model trained only using voxelwise MAE created clear striations both in sagittal and coronal views (axial slices were generated). However, using the VGG feature loss helped remedy this issue (although not completely).

Due to the project's goal of generating segmentable CSFn volumes from WMn volumes, samseg was used to evaluate the model's output by qualitatively evaluating the ability to segment the synthesized CSFn

volumes. Samseg was run on the synthesized CSFn volumes and the ground truth (registered to WMn) CSFn volumes. The unregistered CSFn volumes were not used as comparison because registration may change structure volumes, and will make Dice scores irrelevant.

Recall that the model was trained on CSFn volumes registered to WMn volumes as the ground truth. To evaluate the output of the model, we applied the appropriate inverse warp to the model's output volume in order for it to be in the same space as the original, unregistered CSFn volume. Then, we ran samseg on both the input volume and the inverse-registered model output.

As a comparison baseline, samseg was also run on the original WMn volumes and the CSFn volumes that had been registered to their corresponding WMn volumes.

Dice scores and percentage volume change were calculated. For the percentage volume change, the original CSFn volumes and the WMn volumes were used as the original volume for each experiment respectively.

5 Possible directions

It may be more effective to train a custom discriminator for the single slice output model rather than using the current VGG16 perceptual loss.

Because the current discriminator architecture is known to work on WMn vs corrupted CSFn vs uncorrupted CSFn volumes, it may be possible to use a GAN architecture. Because we have so much paired data, we can train the GAN in N steps:

1. Use the generator to synthesize CSFn volumes
2. Train discriminator on real CSFn volumes
3. Train discriminator on the synthesized CSFn volumes
4. Train the full model (where the discriminator is frozen) on input WMn images and discriminator labels of 1.

Because the existing discriminator model rejects WMn and corrupted CSFn image, using it as a start for the GAN discriminator may decrease training time. This is not significantly different from the custom perceptual loss at first glance, as it uses the same discriminator; however, in this case, the discriminator is trainable to force the generator to generate more faithful CSFn images.

6 Summary of approaches

* denotes likely best approach based on qualitative results

1. Model with 5-slice slab output
   (a) Voxelwise MAE
   (b) Voxelwise MAE + slicewise VGG16 perceptual loss
   (c) * Voxelwise MAE + discriminator feature loss
2. * Model with single slice output
   (a) Voxelwise MAE
   (b) * Voxelwise MAE + VGG16 perceptual loss

7 Summary of next approaches

1. Model with 5-slice slab output
   (a) Voxelwise MAE + discriminator as adversarial network
   (b) Voxelwise MAE + discriminator feature loss + discriminator as adversarial network
2. * Model with single slice output
   (a) Voxelwise MAE + discriminator feature loss
   (b) Voxelwise MAE + new discriminator as adversarial network
   (c) * Voxelwise MAE + discriminator feature loss + new discriminator as adversarial network

* If it is not feasible to run experiments for every potential approach listed, I believe the most promising approach is 2c, which can be tested in portions via 2a and 2b separately. Using single slices as outputs circumvents the striations between slabs observed in the 5-slice slab output model.

8 Tangential possibilities

I'm curious to see if we can train a network to directly predict label probabilities for a slice (or thin slab of slices). This may not be infeasible because the model's architecture would have to be minimally changed and given samseg's performance on the 5-slice model trained only using voxelwise MAE. Note that samseg does produce volumes of the probability of a given structure at every voxel, for any given structure. WMn volumes may be able to be segmented using this strategy as well, given that we have corresponding CSFn volumes. May be interesting to have a WMn/CSFn agnostic segmentation model.

References

[1] L. Umapathy, M. B. Keerthivasan, N. M. Zahr, A. Bilgin, and M. Saranathan, "A contrast synthesized thalamic nuclei segmentation scheme using convolutional neural networks," 2020.

[2] S. U. Dar, M. Yurt, L. Karacan, A. Erdem, E. Erdem, and T. Çukur, "Image synthesis in multi-contrast mri with conditional generative adversarial networks," *IEEE Transactions on Medical Imaging*, vol. 38, no. 10, pp. 2375–2388, 2019.