MRI Segmentation of the Knee using U-Net

Sophia Kim-Wang

DeFrate Lab

BME 590 – Machine Learning

29 April 2019

Introduction

Machine learning is a relatively new computational method that can be applied to any area of research. It is being used to classify objects in self-driving cars, identify numbers on a handwritten check, predict personalized health outcomes in the medical setting, and more. The applications of machine learning are far reaching. For the application in knee biomechanics and cartilage health, many musculoskeletal labs may benefit from the automated segmentation of biomedical imaging. However, deep learning has not been applied to segmentation of musculoskeletal images nearly as much as neurological imaging. Specifically, magnetic resonance imaging (MRI) is of particular interest due to its ability to make high resolution images of soft tissues, such as cartilage, ligaments, tendons, and muscle.

The DeFrate Lab is especially interested in cartilage deformations and structural properties after physical activities, during dynamic motions, and in individuals with knee injury [1, 2, 5, 6]. Current work in the lab involves manual segmentation by research personnel to segment the bones of the knee (tibia and femur) and their respective articular cartilage and ACL footprints (**Figure 1**). From these segmentations, we can make three dimensional models to then calculate cartilage strains, ACL strains, and specific MR relaxation times (T1rho and T2). Unfortunately, current segmentation programs available on the market require additional manual adjustments, taking as much time as manual segmentation if not more. Thus, our lab has remained using manual methods, as we are also able to obtain sub-pixel resolution. With the recent successes in machine learning and image segmentation, it would be beneficial to test the implementation of this method to our MR images.

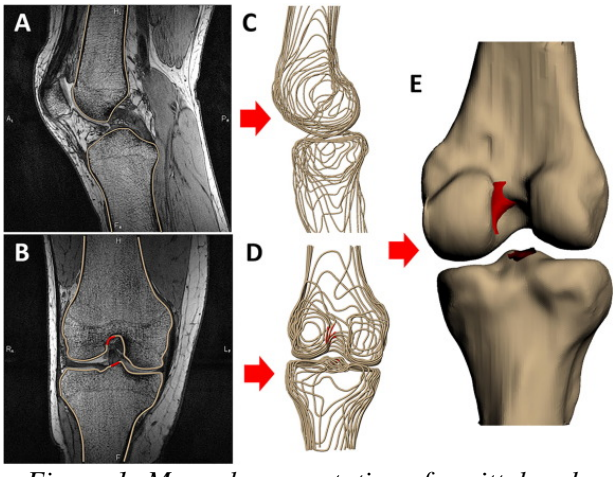

Figure 1: Manual segmentation of sagittal and coronal knee MRI scans to create a 3D rendered model [2].

Related Work

Currently, there are some automated segmentation algorithms available. For example, Koo et al. [3] developed a patented method of extracting features (voxel values) from MRI images with two different contrast mechanisms. These features are then separated into clusters of musculoskeletal tissue type (i.e. cartilage, bone, etc), and a support vector machine (SVM) method is applied for

training. Many of these prior models require large training sets to establish a prior knowledge. Zhang et al. tried to mitigate this and to enhance the SVM by incorporating discriminative random fields which enables contextual information to be included regarding the neighboring voxels in the MRI images [7]. More recent work by groups have started using convolutional neural networks (CNN) to automatically segment the bones and cartilage in the knee [4]. This study used a specific encoder-decoder architecture, SegNet, for its CNN and also incorporated a three-dimensional deformable model, which helps to maintain smooth boundaries and its overall shape.

Methods

Training Data Set:

Fortunately, our lab has years of training data (approximately 500 human subjects' knee MRI scans). Each subject's scan consists of 125 slices, where about 80 of the slices have segmentations for the tibia, femur, and articular cartilage. Unfortunately, we are limited by the computing power of our lab's machines and were not able to use the Google Cloud Platform's GPU instances as we must follow HIPAA guidelines.

Using a desktop computer with 10 GB of memory, I was able to successfully use a dataset of 517 images from 7 subjects scans and their respective masks. 10% (52) of these images were used for testing, resulting in 465 images for training. The following MRI specifications were used for the dataset used in this study: 512x512 pixels, slice thickness 1mm, double-echo steady-state sequence (flip angle: 25 degrees, repetition time: 17 ms, echo time: 6ms). All images were acquired on the same scanner, a 3T Trio Tim (Siemens) [5].

With regard to the segmentation masks (ground truth), the original manual segmentation is completed on 3D modeling software (Rhinoceros) where all images are stacked and separated by 1mm such that the outlines can be compiled to make a 3D wireframe model (**Figure 1**). In order to convert these outlines to masks that can be used for machine learning, the outlines were converted to a point cloud. Then, the point clouds of the tibia were converted to masks in MATLAB. The pixel values in these masks were 0 for non-tibia regions and 1 for tibia regions (Figure 2). This project addresses only the segmentation of the tibia, which enabled the images and masks to be cropped to 256x256. Addressing only the tibia and cropping the images in half significantly increased the optimization time, which was necessary given the desktop computer's constraints.

Convolutional Neural Network: U-Net

The CNN used for this image segmentation was a U-Net model adapted from a pre-existing UNet model. This model has 23 convolutional layers that included a repeated unit of two convolutional layers (varying kernel size, ReLU activation, and padding included) followed by a 2x2 max pooling layer (stride of 2). In the encoding/downsampling steps, the number of feature channels is doubled, starting at a feature channel of 8. In the decoding/upsampling steps, there are equivalent convolutions as the encoding steps but it is an "up-convolution" and the features

are halved. Rather than maxpooling, concatenation steps take place. All CNN work and image augmentation (cropping, contrast enhancement) were completed on Jupyter Notebook.

Physical Layer: Blur Kernels

The physical layer applied to this image segmentation model was changing the resolution of the images by changing the blur kernel size. For this project, five different kernel sizes were used: 3x3, 5x5, 8x8, 10x10, and 12x12. The weights of the kernels were optimized for each kernel size. The same datasets were applied to each kernel size. However, the datasets were randomly shuffled and split into training and test batches. After establishing the best performing model for each kernel size, the model and weights were applied to the test dataset. To ensure binary values (tibia=1, not tibia=0), a pixel threshold value of 0.5 was applied to the resulting images.

Metrics

The following metrics were extracted after running the model for all 5 kernel sizes: accuracy (training and validation), binary cross entropy loss (training and validation), and time per epoch.

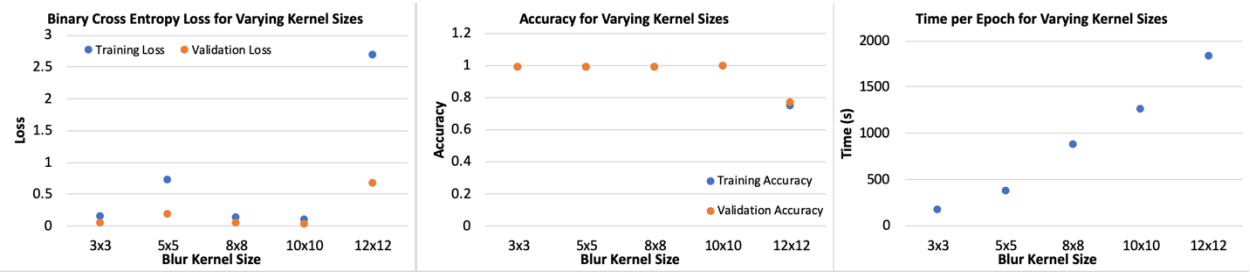

Figure 2: Binary cross entropy loss, accuracy, and time per epoch for varying kernel sizes.

Results

The binary cross entropy loss, accuracy, and time per epoch for each kernel size is reported in the graphs (**Figure 2**). For the loss, it is evident that training loss was always higher than the validation loss and that the losses are very low for all kernel sizes except 12x12. With regard to accuracy, all accuracies were close to 1 with the exception of the 12x12 kernel. This suggests that 12x12 is too large of a kernel size to be used for this dataset. The time per epoch plot also justifies that the larger the kernel size, the longer each epoch takes. Thus, to be efficient but still accurate, it is reasonable to choose the smallest kernel size (3x3).

Upon viewing the representative predicted masks in **Figure 3**, it is clear that the 8x8 and 10x10 predicted masks are slightly overestimating the

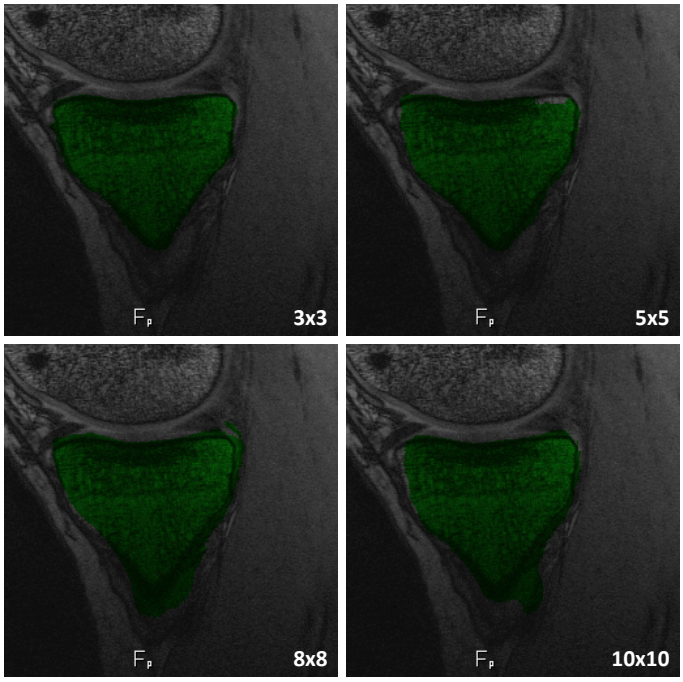

Figure 3: Representative predicted masks (green) produced from the same MRI image using U-Net models with varying kernel size (denoted on the bottom right corners).

Figure 4: Three representative slices from one subject. Ground truth masks (dark gray) overlaid by predicted masks (peach) from the 3x3 results.

tibia, going beyond the margin of the outer cortical bone. The 5x5 result is better but this seems to underestimate a part of the tibial plateau. It is clear, again, that the 3x3 result is optimal. Note that the 12x12 result is not depicted here as the all of its predicted masks came back as a matrix of zeros, indicating no region was predicted as bone. **Figure 4** also shows some representative ground truth masks and predicted masks using the 3x3 kernel size model. Even though the accuracy was nearly 0.985, there are still some visible discrepancies between the ground truth and predicted masks, as noted by the dark gray regions.

Discussion

The results of this project showed that segmentation of MRI images can be accurately performed using a U-Net architecture. With sufficient training data, we can create a model with optimized weights to automatically segment our hundreds of images per subject in a timely fashion. Currently, our lab is manually segmenting each scan, taking up to three days per scan. Because the training data segmentation was completed by a researcher and confirmed by a musculoskeletal radiologist with up to 30 years of clinical experience, we are confident that the optimized weights for this U-Net are reliably consistent and correct.

With regard to the accuracy across all kernel sizes, it is evident that we were able to achieve an average of 0.98 for all kernel sizes except the 12x12 kernel. This is most likely due to decrease in resolution of the images, which ultimately led to hyperparametrization. When there was hyperparametrization, it was evident as the time per epoch increased. For example, the 3x3 kernel took 165 seconds while the 12x12 required 1900 seconds per epoch. Because we were able to achieve the same accuracy with a 3x3 kernel, this kernel size is the optimal choice since the computing time per epoch was also the shortest. Based on these assessments, there is a clear tradeoff between the ratio of pixels in the entire dataset to model parameters and the computing time. The ideal ratio was approximately 62 for this dataset and model. This ratio could be maintained by increasing the training dataset and/or decreasing the number of feature channels in each convolutional layer.

Binary cross entropy loss was more variable between kernel sizes than the accuracy. With the same number of epochs (10), the lowest validation loss occurred with the 10x10 kernel size. The next lowest loss came from the 3x3 kernel size model. Based on the efficiency, metrics, available training data, and visual representations, the 3x3 kernel size appeared to be the best performer

for this scenario. As we increase the dataset and make modifications to the U-Net, it will certainly change as to which kernel size is appropriate.

Future directions for the automatic segmentation of knee MR images include increasing the dataset, elastically deforming the dataset, utilizing a computer with a GPU, and potentially segmenting the entire 512x512 image. To add an additional physical layer, an illumination variable may be added to help improve the segmentation results. A different type of encoder-decoder network may also be applied as in Zhang et al. with the SegNet. Overall, the U-Net proved to be successful in segmenting the tibia of the approximately 500 images, requiring less than 30 minutes to create the model. This has the potential to eliminate the menial labor and time from our lab's research personnel and could enhance the studies in our lab.

References

1. Carter TE, Taylor KA, Spritzer CE, et al. In vivo cartilage strain increases following medial meniscal tear and correlates with synovial fluid matrix metalloproteinase activity. *J Biomech*. 2015;48(8):1461-1468.
2. Englander ZA, Martin JT, Ganapathy PK, Garrett WE, DeFrate LE. Automatic registration of mri-based joint models to high-speed biplanar radiographs for precise quantification of in vivo anterior cruciate ligament deformation during gait. *J Biomech*. 2018;81:36-44.
3. Koo S, Hargreaves BA, Gold GE. Automatic segmentation of articular cartilage from MRI. Patent No: US 8,706,188 B2. Date of Patent: April 22, 2014.
4. Liu F, Zhou Z, Jang H, Samsonov A, Zhao G, Kijowski R. Deep convolutional neural network and 3d deformable approach for tissue segmentation in musculoskeletal magnetic resonance imaging. *Magn Reson Med*. 2018;79(4):2379-2391.
5. Paranjape CS, Cutcliffe HC, Grambow SC, et al. A new stress test for knee joint cartilage. *Sci Rep*. 2019;9(1):2283.
6. Sutter EG, Liu B, Utturkar GM, et al. Effects of anterior cruciate ligament deficiency on tibiofemoral cartilage thickness and strains in response to hopping. *Am J Sports Med*. 2019;47(1):96-103.
7. Taylor KA, Collins AT, Heckelman LN, et al. Activities of daily living influence tibial cartilage t1rho relaxation times. *J Biomech*. 2019;82:228-233.
8. Zhang K, Lu W, Marziliano P. Automatic knee cartilage segmentation from multi-contrast mr images using support vector machine classification with spatial dependencies. *Magn Reson Imaging*. 2013;31(10):1731-1743.