3D Convolutional Neural Networks and MRI for Meniscus and ACL Injury

Jack Weatherbe[301466197], Yvonne Wang[301391109], Daniel Sloseris[301398785], Edoardo Rinaldi[301469176], and Thomas Lin[301404392]

Simon Fraser University {djw23,ynw2,dsloseri,edoardorinaldi,hla250}@sfu.ca

Abstract. Anterior cruciate ligament (ACL) and meniscus tears are among the most common knee injuries, often occurring simultaneously, especially among athletes. They both play a critical role in ensuring adequate stability and controlling movement in the knee, so the accurate diagnosis of any abnormality is crucial. In this paper, we propose the implementation of a 3D Convolutional Neural Network (CNN) model to analyze MRI data obtained from knee scans across the axial, sagittal, and coronal planes. The main objective of the model is to detect any abnormalities associated with the meniscus and/or the ACL. To do so, we developed a multi-label classification model integrating a 'sigmoid' activation function within the output layer and employing compiled binary cross-entropy. Using a dataset of 1370 knee scans, the 3D CNN model achieved an accuracy of XXXXXXXX, which was then compared against the average AUC values obtained by the 2019 finalists of the Stanford MRnet challenge, a competition aimed to develop models for automated interpretation of knee MRI's. In addition, we explored the possibility of reformatting the 3D data images into an extended 2D image collage framework to determine if it was possible to achieve a better training time and overall model performance.

Keywords: Knee injury classification · 3D MRI · 3D convolutional neural network.

1 Introduction

The anterior cruciate ligament (ACL), positioned centrally within the knee joint, is one of the four major knee ligaments and it facilitates rotational as well as forward movements of the shin bone by connecting the femur to the tibia.

Composed of dense connective tissue, it resists anterior tibial translation and rotational forces within the knee joint [1]. ACL injuries result primarily from excessive stretching or complete tearing and are the most common type of knee ligament injuries, specifically among athletes [2].

The meniscus, on the other hand, is a cartilaginous structure in the knee responsible for shock absorption, as well support against tension and torsion [3]. Meniscus injuries are often diagnosed alongside ACL injuries and they are typically caused by abrupt movements, or twisting motions, hence very common among athletes as well. Meniscus injuries can result from trauma or age-related degeneration [4].

Machine learning can assist with diagnosis for effective treatment and Convolutional Neural Networks (CNNs) are particularly adept in image processing compared to other neural networks. In this course project, we built a 3D CNN to predict, classify, and diagnose ACL injuries, meniscus tears, abnormalities, and any combinations of the above diagnoses. Our model 3D model was implemented in TensorFlow, and takes inputs of 3D knee MRI GIFs as training dataset. The dataset was split into training, validation, and testing sets. As comparison, we also attempted the "collage" CNN approach (Gassy paper related work), which has much lower computational resource requirements.

2 Materials

2.1 Dataset

The dataset used for the study was accessed from the Stanford Machine Learning Group, which contains 1370 knee MRI exam results gathered at the Stanford University Medical Centre.

There data contains 1104 (80.6%) abnormal exams with 319 (23.3%) ACL tears, 508 (37.1%) meniscal tears, and 194 (38.2%) with ACL tears coincidental with meniscal. Due to the large proportion of abnormal exams, the dataset is imbalanced with respect to healthy knees, which in the context of our study refers to knees showing no abnormalities, a point which will be mentioned later in the Results section. Each knee MR exam contains three GIFs in the NumPy file format, each along three planes: coronal, axial, and sagittal. The exams were separated into training set (1130), and validation set (120) by Stanford. In addition to the GIFs, the data also contained CSV files for each exam, storing a binary value indicating the existence of injuries (ACL, meniscal, abnormal) as corresponding labels, where labels are 1 if injuries are diagnosed, zero otherwise.

Moreover, we partitioned the Stanford training dataset and labels into our own training set (70%), validation set (20%), and testing set (10), as listed in Table [**?**]. The deep learning model is trained using the training set and its performance during training is assessed with the validation in order to adjust specific parameters. Upon completion of training and validation, the testing set is employed to evaluate the model's performance.

Table 1: Partitioned dataset distribution.

Partition	Percentage	Size
Training	*70*	791 exams
Validation	*20*	226 exams
Testing	*10*	113 exams
Total	*100*	1130 exams

Table 2: Stanford dataset distribution.

Pathology	Percentage	Size
Abnormality	*80.8*	913 exams
ACL tear	*18.4*	208 exams
Meniscal tear	*35.1*	397 exams
ACL and meniscal tear	*11.1*	125 exams
Total	*100*	1130 exams

3 Methods

3.1 Overview

Utilizing a series of 2D images across different planes like axial, sagittal, and coronal, knee MRI's can provide a detailed 3D view of the knee structure.

By implementing a 3D Convolutional Neural Network (CNN) we can leverage the spatial dimensionality in MRI's to analyze the entire image sequence, especially through the use of 3D kernels. In fact, unlike in 2D CNN, the use of 3D kernels allows the linking of information from neighbouring slices (ie. scans) [5] across the various planes, enabling the detection of more complex 3D patterns and characteristics within the knee.

In our research, we developed an ML model to analyze sequences of 3D MRI knee scans to identify injuries in the ACL and/or meniscus, as well as general knee irregularities. More specifically, the model predicts three categories: ACL injuries, meniscus injuries, and overall knee abnormalities. Through the use of 'sigmoid' activation function for binary cross-entropy in the final layer, the model is able to detect and categorize ACL and meniscus anomalies, providing valuable insights into the patient's knee structure.

3.2 Preprocessing

To ensure a consistent slice count across the axial, sagittal, and coronal scans we applied padding, which involves adding extra slices to have a uniform size across the data. More specifically, we determined that 50 slices yielded the best results, therefore we padded the data containing fewer than 50 slices and cropped those over 50 to achieve a uniform 50-slice count across all three planes.

Moreover, to represent all scans by a common scale, we normalized the pixel values within 0 and 1. More specifically, for memory management purposes, we performed batch-wise normalization for batches of 10 where the minimum and maximum pixel values for each batch were used as references for the normalization process. The rationale is because when working with deep learning models, ideally the numbers should be as small as possible, so scaling is a common practise to help models converge more efficiently while training.

In addition, given that the objective of the model was to recognize patterns within the region of the knee containing the ACL and meniscus, we cropped the input scans to 160×160, in order to focus on the knee subregions of interest.

3.3 Data Augmentation

The augmentation process for the MRI scan slices included two main transformations: random rotation and horizontal flipping.

Random rotation reorients the slices by multiples of 90 degrees, ensuring that the model is exposed to various image orientations of the knee scans and, as a result, improving the model's robustness.

Horizontal flipping, on the other hand, mirrors scan slices along the width axis, further improving the model's adaptability to different spatial orientations of the scans. Given that ACL and meniscus tears are characterized by specific characteristics, the augmentation process did not include shearing, to avoid distorting the scans and negatively affect the integrity of the data.

3.4 3D CNN Model - INCLUDE A FLOW CHART???????

To conduct the model's training, we used a computer with an NVIDIA GeForce RTX 2060 GPU. The data was analyzed in batches of 10 by implementing the model with the library TensorFlow using Python (version 3.8.6). Early stopping criteria based on validation loss were implemented to prevent the model from overfitting and set to a limit of 5 epochs.

The architectural design of the 3D Convolutional Neural Network (CNN) in this study was largely influenced by Guida et al. [5]. Initially, a convolutional block was created, incorporating 32 kernels of size $7 \times 7 \times 7$ and implementing batch normalization to streamline data, alongside Rectified Linear Unit (ReLU) activation for better data interpretation. Later steps included applying a MaxPooling layer for data compression and integrating two sets of residual blocks using 3D convolutions. Additionally, a dropout layer set at a 50% rate was used to prevent the model from fixating on specific details.

The model ended with a GlobalMaxPooling3D layer to extract relevant data features, transitioning into a densely connected layer with 1024 units for further data combination, reinforced by an extra dropout layer. Its output layer included three nodes using the sigmoid activation function to calculate probabilities of the three categories: ACL injury, meniscus injury, knee abnormality. Moreover, the model's training used the Adam optimizer with a value of 0.001 to minimize binary cross-entropy loss.

4 Results

We fitted our model with 14 epochs, which is number of times the model iterates through the entire training dataset. We also passed in our validation dataset, and saved our training records inside an object declared "history". To gauge our model's performance, we evaluated it with four TensorFlow metrics: Precision, Recall, Accuracy, and AUC.

In the context of our project, Precision is a measure of the model's accuracy in correctly identifying meniscus tears, ACL tears, and abnormal knees among the predicted positive scans and it is calculated as follows:

$$Precision = \frac{TP}{TP + FP}. \tag{1}$$

where TP = True Positives, FP = False Positives.

Recall, on the other hand, measures the model's effectiveness in detecting meniscus tears, ACL tears, and abnormal knees correctly, hence the fraction of actual injuries correctly diagnosed and it is measured as:

$$Recall = \frac{TP}{TP + FN}, \tag{2}$$

where FN = False Negatives. It is worth mentioning that for medical diagnoses, Recall is a more important metric than Precision and there is trade-off between the two metrics. The reason for this consideration is that detecting (and potentially curing) false positives is usually less detrimental than not correctly identifying true positives.

Accuracy is the ratio between the total number of correct predictions and the total number of predictions, measured as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \tag{3}$$

which is the fraction of how many diagnoses our model correctly predicted, including "healthy".

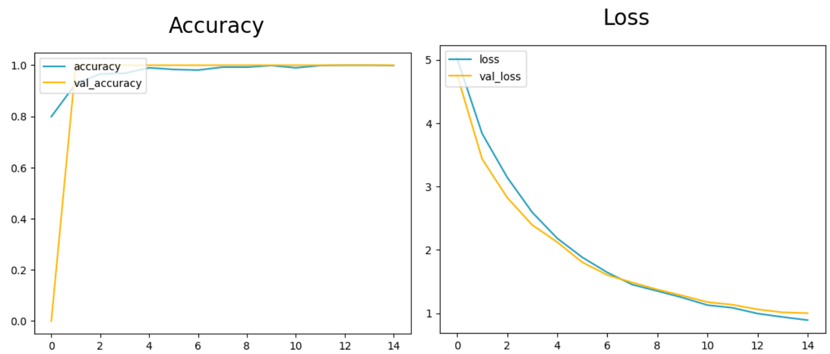
Lastly, AUC (Area Under Curve) measures the model's ability to distinguish healthy knees and those with a meniscus injury, ACL injury or general abnormalities across varying threshold values. A higher AUC value indicates that our model can successfully distinguish a healthy knee and an injured one within the three classes, ACL, meniscus or abnormal.

The summary of our metric values are included in the Table below.

Table 3: Evaluation results.

Precision	Recall	Accuracy	AUC
0.6968	0.7397	0.7493	0.8221

From the classification metrics, the model's overall predictions were 74.93% (Accuracy) correct, with 69.68% (Precision) of the model's positive predictions correct. In the figures below, we also plotted our overall training loss and validation loss, as well as our training and validation accuracy saved in our model history.

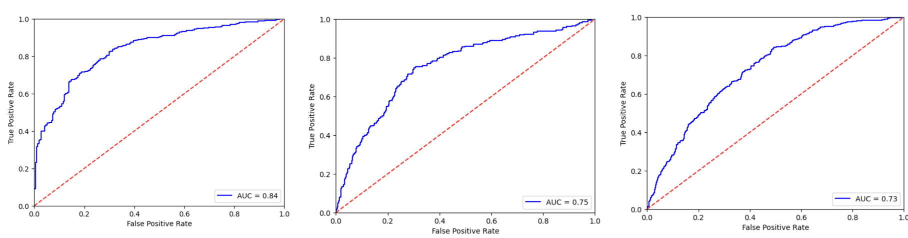


(a) Accuracy vs. Epoch. (b) Loss vs Epoch.

Fig. 1: Accuracy increase and Loss decrease over 14 epochs.

As the figure shows, both the training and validation loss decreased over time, indicating that our efforts with Regularisation, Normalization, and Dropout were successful in rectifying overfitting.

For a more detailed illustration of our model's classification performance, we plotted the individual ROC curves of our three pathologies, with their respective optimal thresholds in the Table 4 below.



(a) ROC curve for abnormal classification. (b) ROC curve for ACL injury classification. (c) ROC curve for meniscus injury classification.

Fig. 2

Table 4: Optimal threshold for maximum AUC.

	Abnormal	ACL	Meniscus
Threshold	0.6363	0.4486	0.5028

5 Accomplishments

Inspired by a research paper co-authored by our professor, Dr. Ghassan Hamanreh [6], we attempted, as a side project, to develop an image collage based deep convolutional neural network (CNN) approach to detect knee injuries. Since in the paper [6] it is mentioned that 2D collaging is less computationally intensive compared to a 3D CNN, we wanted to explore whether it would perform better than our model.

The approach involved transforming 3D GIF's volumes into 2D 4×4 grids of slices, which was an arbitrary choice. These 16 slices were specifically chosen from the middle of the GIFs to preserve the integrity of the data while reducing computational complexity. However, for future work, larger grids could be considered, as the limited size of ours might have resulted in the loss of important information regarding the knee structure.

To focus on the knee subregions of interest and reduce computational load, the slices were cropped to 128×128. The tiled grids were systematically arranged into a 1×3 array across all three axes, laying the groundwork for subsequent model development.

For model creation, TensorFlow Keras facilitated the stacking of labels (ACL, meniscus, abnormal) into a tuple, serving as the image data's labels. We tested various models, more specifically binary, multi-class and multi-label.

The binary model was an initial trial in which we tried to only distinguish meniscus tears from non-meniscus tears. However, this model consistently overfitted, possibly due to the small tear sizes within the 4×4 grids. Future work should involve deeper neural networks with smaller filters to extract information more effectively from these 1×3 image grids, as well as investigating further how to better tune automated hyperparameters. Moreover, we suggest using a more balanced dataset, as the one we used contained slightly than 20% of healthy scans.

Two other approaches involved multi-class and multi-label models, where we divided the labels into seven classes covering all the possible diagnoses combinations, as shown in Table 5.

Table 5: Labels used for the 2D collaging model.

Label Number	Label Type
1	healthy
2	ACL tear
3	Meniscus tear
4	Abnormal
5	ACL and Meniscus tear
6	Meniscus tear and abnormal
7	ACL tear and abnormal
8	ACL tear, meniscus tear and abnormal

Despite efforts to balance the dataset, overfitting persisted, leading to poor performance on the validation dataset. For the multi-class as well as the multi-label model, we think that further investigation into automated hyperparameter tuning might mitigate overfitting and yield a better model's performance.

6 Contributions

- **Jack Weatherbe**: Researched potentially viable 3D CNN models to analyze our data with the rest of the team. Did extensive background research on the model we ultimately chose. Preprocessed and augmented the data to adequately train the model and implemented the key components of the model whose functions and parameters were later adjusted with the help of the team. Overviewed the drafting of the report to ensure that the model architecture and findings were accurately reported.
- **Yvonne Wang**: Developed from scratch a website to ulpoad MRI scans detectable by our model. Connected the backend to the front end so the uploaded images could be connected and analyzed by our ML model. Helped Jack train his model by suggesting a few changes (ie. change validation loss parameter. Reviewed the final report to ensure it accurately reported our findings. Routinely participated in group meetings to ensure other team members were aware of the progress in building the website.
- **Daniel Sloseris**: Extensively researched viable 2D CNN models as an alternative solution to the 3D CNN model. Developed and tested a 2D collaging model as suggested by Professor to determine if it would yield a better performance than the 3D model. Routinely briefed teammates on my progress and technical issues related to the 2D collaging project, so that they were aware of the results and they could faithfully document them. Overviewed the drafting of the 2D collaging mode sections in the report.
- **Edoardo Rinaldi**: Did extensive research on potential project topics and found the dataset we eventually used for the project. Contacted various research groups to ask for their datasets and infromation regarding their project. Worked alongside Yvonne in the website development, cleaning up the code and adjusting a few UI features. Routinely participated in group meetings to document the technical progress of every member in order to accurately document it in the final report. Worked alongside Thomas to do background research on 2D and 3D models, as well as ours specifically in order to draft the final report.
- **Thomas Lin**: Did background research alongside Edoardo on various studies in knee-related issues to determine a viable 3D CNN model to implement in our project. Routinely documented Daniel's findings, technical issues and progress to adequately report them in our final paper. Participated in group meetings to take notes on other members' progress, in order to document it in the final report.

7 Conclusion and Discussions

Suggested Future work for you (if you continue working on this, e.g. next semester) or for other students to pick up where you left off.

8 Future Work

Suggested Future work for you (if you continue working on this, e.g. next semester) or for other students to pick up where you left off.

Acknowledgements

We would like to sincerely thank our professor, Dr. Ghassan Hamarneh, for his guidance and valuable insights. His paper served as the foundational inspiration for our endeavor to construct a 2D collaging model.

Our sincere appreciation extends to our teaching assistants, Ben Cardoen and Kumar Abhishek, who patiently assisted us during the model implementation by providing resources and insights on how to resolve technical issues.

We are also indebted to the Stanford MRnet group for their generosity in providing the data which formed the basis of our project.

The support and assistance from these individuals and groups have been instrumental in the completion of our project, and we are immensely grateful for their contributions.

Appendix

You can use Appendices for listing detailed low-level information more appropriate to separate from the previous sections not to interrupt the flow of the reading.

Document where the data and source code can be found, and what, if any, steps are needed to reproduce.

References

1. Duthon VB, Barea C, Abrassart S, Fasel JH, Fritschy D, Ménétrey J. Anatomy of the anterior cruciate ligament. Knee Surg Sports Traumatol Arthrosc. 2006 Mar;14(3):204-13. doi: 10.1007/s00167-005-0679-9. Epub 2005 Oct 19. PMID: 16235056
2. Evans J, Nielson Jl. Anterior Cruciate Ligament Knee Injury. [Updated 2022 May 5]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2023 Jan-. Available from: https://www.ncbi.nlm.nih.gov/books/NBK499848/
3. (n.d.). UCSF Meniscus Preservation Center. https://meniscus.ucsf.edu/what-meniscus
4. Meniscus Tear — HealthLink BC. (n.d.). https://www.healthlinkbc.ca/illnesses-conditions/injuries/meniscus-tear

5. Guida, C.; Zhang, M.; Shan, J. Knee Osteoarthritis Classification Using 3D CNN and MRI. Appl. Sci. 2021, 11, 5196. https://doi.org/10.3390/app11115196
6. Hussain, M. A., Amir-Khalili, A., Hamarneh, G., Abugharbieh, R. (2017). Collage CNN for Renal Cell Carcinoma Detection from CT. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (pp. 000-000). DOI: 10.1007/978-3-319-67389-$9_2$7.