Detection and Severity Classification on Knee Osteoarthritis X-ray Images

Jean Janssen
Department of Advanced Computing Sciences
Maastricht University
Maastricht, The Netherlands

Abstract

This paper addresses the challenges in diagnosing and classifying knee osteoarthritis (KOA) severity using traditional methods. Using machine learning and Convolutional Neural Networks (CNNs), specifically MobileNetV2 and a simple CNN structure, consisting of the concatenation of convolutional and pooling layers called *SimpleCNN*, the study utilizes a three-stage approach involving data splitting, model training, and classification. The proposed methods demonstrate a maximum F1 score of 0.8468 for the use of a pretrained MobileNetV2 in comparison to a MobileNetV2 without pretrained weights and SimpleCNN where an F1 score of 0.7516 was yielded. The research conducts the potential of CNNs without pretrained weights, for automated KOA diagnosis and severity classification, offering valuable insights for further exploration in this domain.

1 Introduction

Knee osteoarthritis (OA), most common in elderly, is the progressive loss of articular cartilage in the knee area. OA is a disease that is typically progressive. The intensity of the symptoms may vary, but gradually get worse over time and may eventually lead to a disability (Hsu & Siwiec, 2022). In 2019, about 528 million people worldwide were living with osteoarthritis ("Osteoarthritis", 2023). In the medical field, the Kellgren-Lawrence (KL) score is used to classify the severity of knee osteoarthritis (KOA) from radio-graphs (Macri, Runhaar, Damen, Oei, & Bierma-Zeinstra, 2022). The KL grading system, splits the severity of this disease into five different severities that are labeled as follows:

- 0 (Healthy)
- 1 (Doubtful)
- 2 (Minimal)
- 3 (Moderate)
- 4 (Ssevere)

The accuracy of the severity grading diagnosis is dependant on a physician's perception and experience. Therefore, it is very difficult to pinpoint the exact reasoning as to why a case of KOA received a high/low grade. Furthermore, KOA is a disease that is very hard to detect within the earlier stages of the disease. This causes the differences between a grade 0 case and a grade 1 case very hard to be distinct. Due to these imperfections in diagnosis, automated and more advanced techniques have been introduced. Machine Learning (ML) and Deep Learning (DL) play a pivotal role in advancing the fields of classification and detection, which can be useful for healthcare, particularly in the detection and severity grading of knee osteoarthritis (KOA). Convolutional Neural Networks (CNNs) stand out as powerful models in this context, showcasing success in analyzing X-rays. Examples of well-known CNN architectures for medical image analysis include VGGNet, ResNet, MobileNet and DenseNet. These models are often implemented with pretrained weights as they have been seen various use cases for detection. To benchmark these models large datasets such as ImageNet was used. We can use these models even with training on data from ImageNet in medical diagnosis. This can be done through image classification enabling them to capture features and patterns associated to KOA. With the capabilities of CNNs in automated learning, not only can we enhance diagnostic accuracy, but also enable a more efficient and scalable approach to screening and monitoring these conditions. The application of CNNs in KOA detection is useful, but furthermore, the models can be used after training to predict severity thus making the of physicians more automated.

1.1 Research Aims

Within this research, my aim is to extend the severity classification with the use different models than those used in the state-of-the-art that use less parameters in an attempt to match the performance of previous. I aim to achieve the following through this research:

- Use binary classification for diagnosis purposes in addition to severity classification.
- To compare the impact of using pretrained weights for classification and severity classification.
- To match a built from the ground up neural network's performance with a state-of-the-art convolutional neural network used for severity classification.

2 Related Work

Let us discuss some of previous work performed on the classification of KOA.

The authors of (Chen, Gao, Shi, Allen, & Yang, 2019) used data obtained from the osteoarthritis initiative (OAI), where X-ray images of the whole knee area were used. Therefore, preprocessing was needed before the OA quantification of the knee could take place. For this purpose, a custom YOLOv2 network for knee joint detection was used. After this, variants of neural networks of different variants were able to classify the knee X-ray images into KL graded severities. Their version of VGG-19 came out on top where the model achieved a 69.7% accuracy at detecting the severity of KOA. In addition, in (Mohammed, Hasanaath, Latif, & Bashar, 2023), the model of detecting the severity of KOA was revised. This was done by splitting the model which would classify the severity of KOA into a model that would be trained first for the purpose of binary classification. The binary classification would take place on cases of a healthy versus an unhealthy knee joint. Then, using this trained model, severity classification would be performed on the classes of grades 2 (minimal), 3 (moderate) and 4 (severe). The data used within this study was retrieved from kaggle (Tiwari, n.d.) where images were divided into these severity classes. Their best model yielded a performance of 89% on those three classes using the ResNet101 model after training this on binary detection. Within this study, multiple network architectures were considered. Amongst these was, MobileNetV2, which achieved the highest F1-score of 0.67 (Mohammed et al., 2023) on a five class dataset.

MobileNetV2 (Sandler, Howard, Zhu, Zhmoginov, & Chen, 2018) is network architecture that balance computational efficiency and accuracy. It was introduced as an improvement of its predecessor, MobileNetV1. It incorporates depth-wise separable convolutions, linear bottlenecks and inverted residuals. The depth-wise separable convolutions allow the model to split convolution into two layers. The first layer performs lightweight filtering and is called the depth-wise convolution. The second layer is responsible for building new features and is called the point-wise convolution. This reduces the computational cost to be eight to nine times smaller only at a small reduction in accuracy (Howard et al., 2017). In addition, a linear bottleneck is introduced to reduce non-linear activation after each convolutional layer. Furthermore, inverted residuals are used to use shortcuts directly between the bottlenecks with the intuition that bottlenecks actually contain all necessary information (Sandler et al., 2018). Because of this, MobileNetV2 is well-suited for this research as it allows for running the model on the hardware used for the project, while there is no significant loss of accuracy within this research.

3 Methodology

Within this section. Let us elaborate on the proposed approach for KOA detection and classification. This approach is consistent of three main stages. The first stage is the data acquisition, the second stage is the model training and lastly the classification. Firstly, the dataset is a collection of KOA X-ray images of the knee joint which was obtained from the Osteoarthritis Initiative (OAI) and was retrieved from kaggle (Tiwari, n.d.). These images are cropped versions of knee X-ray images that focus on the knee-joint area. The dataset is divided in a separate train set, test set and validation set and within these sets there were 5 different classes. These classes were, as previously mentioned, 0 (healthy), 1 (doubtful), 2 (minimal), 3 (moderate), and 4 (severe). A more detailed description of the dataset will be provided within section 3.1. For the purpose of performing two types of KOA classification, the dataset had to be rearranged into three separate datasets. We will name these datasets: Dataset 1, Dataset 2 and Dataset 3. Dataset 1 is the original dataset containing all 5 severity classes as defined previously. Dataset 2 splits the positive and negative classes into 2 separate classes 0 (negative) and 1 (positive). The negative class 0 consists of the combination of class 0 (healthy) and 1 (doubtful) from Dataset 1. The positive class 1 consists of the classes 2 (minimal), 3 (moderate), and 4 (severe) from Dataset 1 combined. Dataset 2 was used for the binary classification (diagnosis of KOA). Dataset 3 consists of three classes which were the original untouched classes 2, 3 and 4 of Dataset 1. This split of the data was first introduced by the authors of (Mohammed et al., 2023). For a visualisation of this split of the data, see table 1.

Dataset	**Classes**
Dataset 1	0 (healthy), 1 (doubtful), 2 (minimal), 3 (moderate), 4 (severe)
Dataset 2 (Diagnosis)	0 (healthy, doubtful) → (negative), 1 (minimal, moderate, severe) → (positive)
Dataset 3 (Severity)	2 (minimal), 3 (moderate), 4 (severe)

Table 1: Proposed split of the data for KOA detection and classification.

In order to perform the KOA detection and classification, CNN models were used on the data. For setting a baseline the CNN models will be trained on Dataset 1. The same CNN models will be then used to, first, be trained on Dataset 2. After training on Dataset 2, this model is stored and used for training on Dataset 3. Lastly, a comparison will be performed between these models. For this study, the CNN models MobileNetV2 and a simple CNN structure was chosen, details of this simple CNN structure will be further explained in section 3.2. In order to make a fair comparison, MobileNetV2 was both used with and without pre-trained weights. The choice of this networks was based on availability, accuracy and computational complexity. As mentioned in section 2, MobileNetV2 is less computationally complex while not sacrificing a lot of accuracy. The computational resources that were used for this training was a local machine of which the specifications are mentioned in section 4.1. The following subsections will go into further detail about each aspect of the proposed methods.

3.1 Data

In this study, the knee X-ray images used for training the models in this research was taken form kaggle (Tiwari, n.d.) and were made available by the OAI. A total of 8260 knee images were used, which are divided into 5 severity levels based on the Kellgren-Lawrence (KL) (Macri et al., 2022) grading system. These 5 levels are: 0 (healthy), 1 (doubtful), 2 (minimal), 3 (moderate), and 4 (severe). The images all have a resolution of 224×224 pixels. A summary of the division of the images amongst the severity classes of the OAI dataset, is provided in Table 2. In addition, some sample images of the AOI dataset are provided in figure 1.

Class	**Amount of Images**
0 (Healthy)	3253
1 (Doubtful)	1495
2 (Minimal)	2175
3 (Moderate)	1656
4 (Severe)	251

Set	**Amount of Images**
Train set	5778
Test set	1656
Validation set	826

Table 2: The division of the images over the classes as well as sets of the OAI dataset.

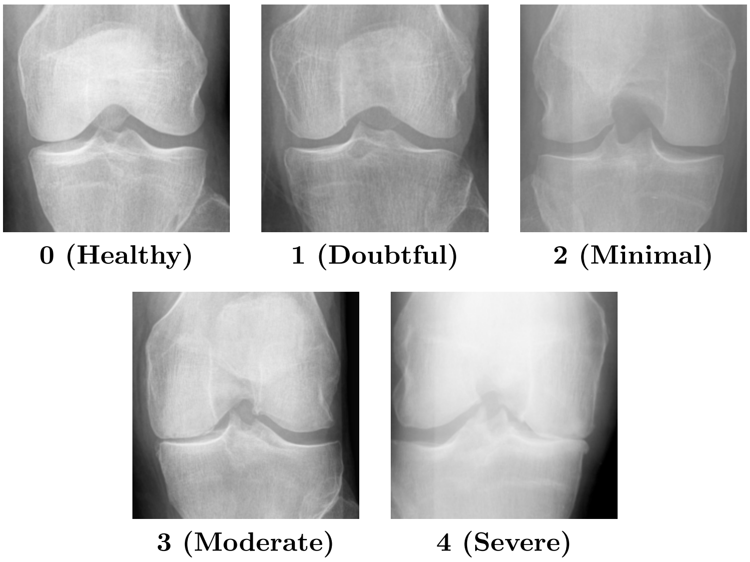

Figure 1: Sample images of each class of the OAI dataset.

The images were not pre-processed in any form. So, for this study, a total of 8260 X-ray knee joint images were used.

In previous studies (Mohammed et al., 2023), the images were preprocessed by cropping the images from the top as well as the bottom. In addition, the images's histograms were equalized. With these additions to the data, worse results were achieved in the experimental phase within this study. Therefore, I decided to use the original images of the OAI dataset.

3.2 Convolutional Neural Networks

Within the field of Deep Learning (DL), convolutional neural networks (CNNs) represent an algorithm that enables a computer to find patterns and features within images the way the human brain would. A CNN takes an image as an input and learns patterns within this image by assigning weights to various features of the image such that the image is distinguishable from other images that are also being processed by the CNN.

What makes a CNN different from a regular Neural Network is that a CNN uses convolutional operations to extract high-level features from an image. A CNN is made up of layers and we call this layer where the convolutional operations happen, the convolutional layer. A CNN transforms the image in order to pass it through many filters while maintaining the important features. Another important layer within a CNN is the pooling layer. This pooling layer reduces the dimensionality of the image. This is done in order to reduce the computational complexity of processing the image. A convolutional layer is followed up by a pooling layer. This is the fundamentally, the structure of the *SimpleCNN*.

In addition to the use of the *SimpleCNN*, an architecture that has performed well on benchmarks and has been proven to be successful was picked. This architecture is MobileNetV2 (MNetV2) (Sandler et al., 2018). Models like VGGNet, MobileNet, ResNet etc. have been highly effective within the field of transfer learning. Transfer learning is the practice of using these models that have a huge knowledge of features over a variety of images due to their training. We can use MobileNetV2 with a set of weights called ImageNet. These weights are determined on a large set of images and serves as a benchmark for the performance of MobileNetV2. Since the SimpleCNN is trained from the ground up, the MobileNetV2 is also trained on the data without the use of pretrained weight to perform a fair comparison with the SimpleCNN. For simplicity, the model with pretrained weights from ImageNet will be referred to as simply "MobileNetV2" and the model without pretrained weights will be reffered to as "MNetV2 no weights".

3.3 Challenges

As mentioned previously, for this study, the data was not pre-processed in any shape or form. This was the case because unstable learning would occur when training state-of-the-art models on the datasets as defined within section 3.1. Unstable learning refers to fluctuations in a model's performance metrics such that the model cannot converge reliably. This could be the cause of the choice of hyper-parameters, insufficient data or the presence of noise within the training dataset. In general, from studies like those performed in (Mohammed et al., 2023), overfitting is an apparent issue. This can be concluded from the results, given that for MobileNetV2 the training loss was low, at 0.2319, however, the validation loss was high at 2.7274. As an example, the following figure showcases a case of unstable learning using images that are cropped 60 pixels from the top and bottom in addition to being histogram equalized.

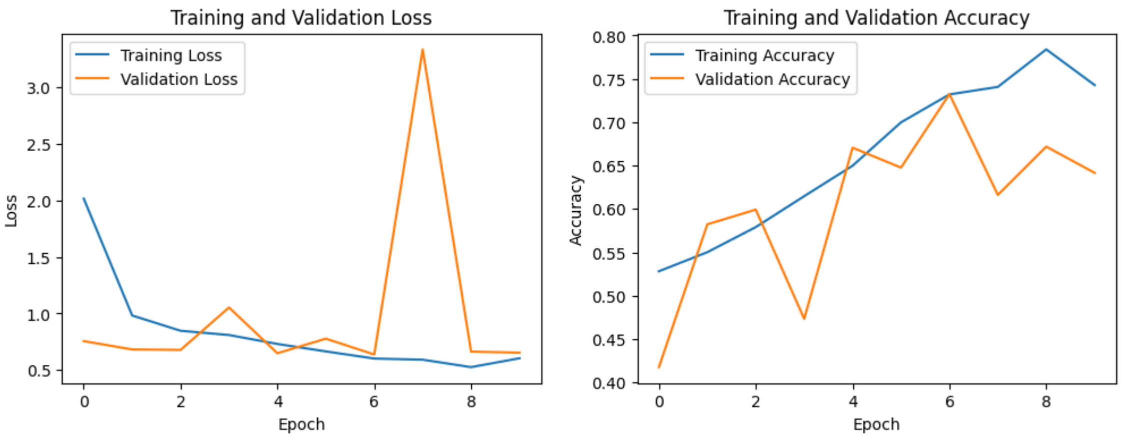

Figure 2: Binary classification on preprocessed dataset using MobileNetV2

Because of this, this research was used performed the original images without any preprocessing of the dataset made available by the OAI.

3.4 SimpleCNN

Within this section, Let us explain the architecture of the SimpleCNN.

As explained previously, a CNN model consists of convolutional layers followed by pooling layers. The activation function for calculation of the activation of each neuron within this network, is the ReLU activation function as denoted in equation 1 (Vallés-Pérez et al., 2023).

$$f(x) = max(0, x) \tag{1}$$

Where x is the input of the neuron.

After activation, batch normalization is applied. This is used to enhance stable learning and accelerate convergence. Since there are so many possible features to learn it makes sense to introduce batch normalization. In addition, batch normalization also assists in correcting instability when learning. The SimpleCNN model consists of two convolutional layers, for which the activation is calculated and then batch normalization is applied, followed by two pooling layers. The kernels in the convolutional layers have a size of 3 and the kernels in the pooling layers have a size of 2. The kernels scan and extract spatial features from the input data and depending or their size they cover a larger/smaller area of the images while systematically going over the image pixel by pixel. The pooling layers within this network are both MaxPooling layers. After this, a flatten layer is used to flatten the output from the previous layers. This is followed by a fully connected layer which is batch normalized, again, reducing internal co-variate shift. Then, ReLU activation is used to learn the complex patterns within the data. A dropout layer is included to randomly drop 20% (in our case this was 20%, but this parameter can be set to a different value) of the neurons during training, acting as a regularization technique to reduce over-fitting. Finally, another fully connected layer represents the output layer with the specified number of classes that we want the neural to classify for.

3.5 Evaluation Metrics

To measure the performance of our classifies, we use different metrics other than the classification accuracy. This is because the classification accuracy is only telling the ratio of total correct predictions from the total number of samples in the dataset. We cannot take anything meaningful out of this unless the data is perfectly balanced. For this study, this is not the case. Therefore, the performance of the classifiers are measured by use of the F1 score. The F1 score is the mean of precision and recall. The precision is the ratio of true positives (TP), and thus correctly classified images, to the total number of positives. The recall is the ratio between the TPs and the total number of positive samples in the dataset. The formulae are shown below in the equations 2 until 5.

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{2}$$

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{5}$$

Where TP is true positive, TN is true negative, FP is false positive, and FN is false negative.

4 Experiments

In this section, let us define the experiments run. The experiments were run on three datasets, namely, Dataset 1, Dataset 2 and Dataset 3. Dataset 2 is a binary dataset used for the classification of KOA where classes 0 and 1 were combined to represent a negative class and classes 2, 3 and 4 were combined to represent a positive class. Dataset 3 was used for further severity classification of KOA and was created by leaving out classes 0 and 1 of the OAI dataset.

Each model which was mentioned in the previous section (section 3.2), was used to build a multi-step diagnosis system. First each model was trained on Dataset 2 for detecting KOA, then this model was saved and used to be trained on Dataset 3 for severity diagnosis.

Each model was run for 25 epochs with callbacks and early stopping to prevent overfitting. The results shown in the upcoming section is the result from the last epoch generated. For this same epoch the model was saved.

4.1 Hardware & Software

We performed the experiments on a Windows 11 computer with 16GB of RAM, an AMD Ryzen 3600 CPU, and an NVIDIA GeForce RTX 3060 Ti GPU. For all experiments CUDA was used. The models for these experiments were run in the Python programming language in Jupyter Notebooks using the PyTorch framework.

5 Results

In general, MobileNetV2 with pretrained weigths performs best on all datasets. This is followed by MNetV2 without weights and then lastly followed by SimpleCNN. When performing binary classification, MNetV2 without weights and SimpleCNN perform similarly, but SimpleCNN starts to fall behind when severity classification is performed on Dataset 3. The highest achieved F1 score was yielded by MobileNetV2 with the use of weights from ImageNet. The highest yielded F1 score was 0.8468.

Model	Training Acc.	Training Loss	Validation Acc.	Validation Loss	Testing Acc.	Precision	Recall	F1 Score
SimpleCNN	0.6762	0.7798	0.4734	1.3119	0.4873	0.4541	0.4873	0.4653
MNetV2 no weights	0.7269	0.6531	0.5375	1.1952	0.5707	0.5241	0.5707	0.5299
MobileNetV2	0.7489	0.5850	0.5714	1.2951	0.5791	0.6004	0.5791	0.5509

Table 3: Results of different classifiers on Dataset 1 containing 5 classes

Model	Training Acc.	Training Loss	Validation Acc.	Validation Loss	Testing Acc.	Precision	Recall	F1 Score
SimpleCNN	0.8744	0.2851	0.7034	0.6631	0.7463	0.7500	0.7463	0.7465
MNetV2 no weights	0.8896	0.2597	0.7651	0.5716	0.7257	0.7503	0.7257	0.7224
MobileNetV2	0.9564	0.1118	0.8123	0.6563	0.8037	0.8453	0.8037	0.8001

Table 4: Results of different classifiers on Dataset 2 containing 2 classes

Model	Training Acc.	Training Loss	Validation Acc.	Validation Loss	Testing Acc.	Precision	Recall	F1 Score
SimpleCNN	0.8246	0.4272	0.6957	0.8865	0.6699	0.6508	0.6699	0.6451
MNetV2 no weights	0.9203	0.2113	0.7536	0.8245	0.7406	0.7885	0.7406	0.7516
MobileNetV2	0.9747	0.0776	0.8348	0.6601	0.8474	0.8502	0.8474	0.8468

Table 5: Results of different classifiers on Dataset 3 containing 3 classes

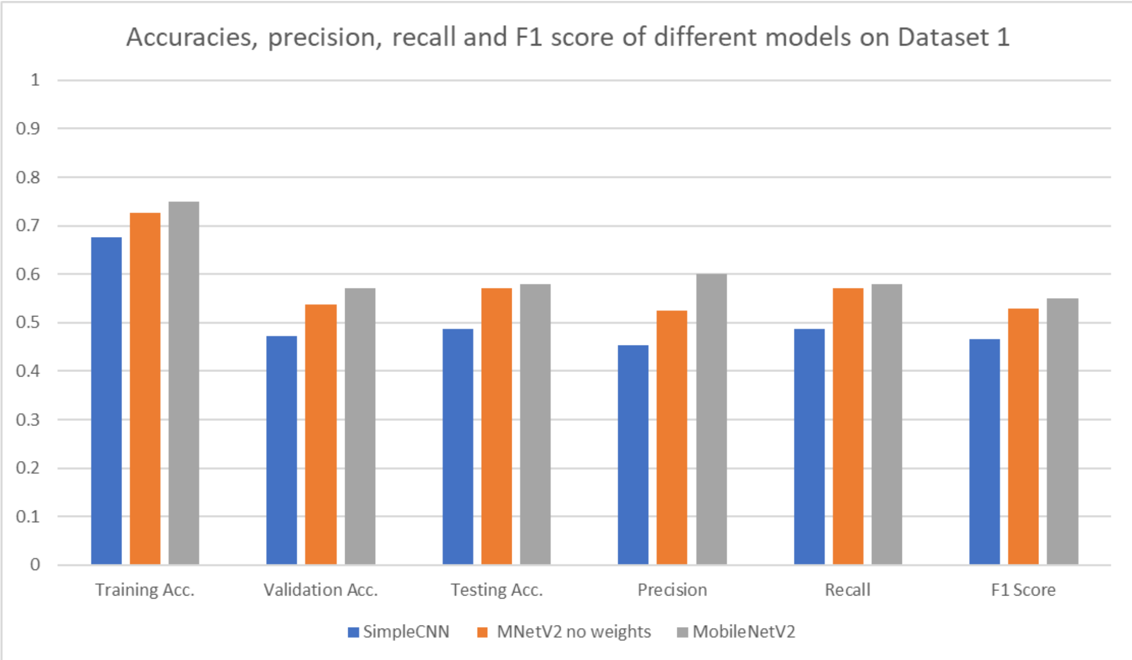

Figure 3: Accuracies, precision, recall and F1 score of different classifiers on Dataset 1 containing 5 classes

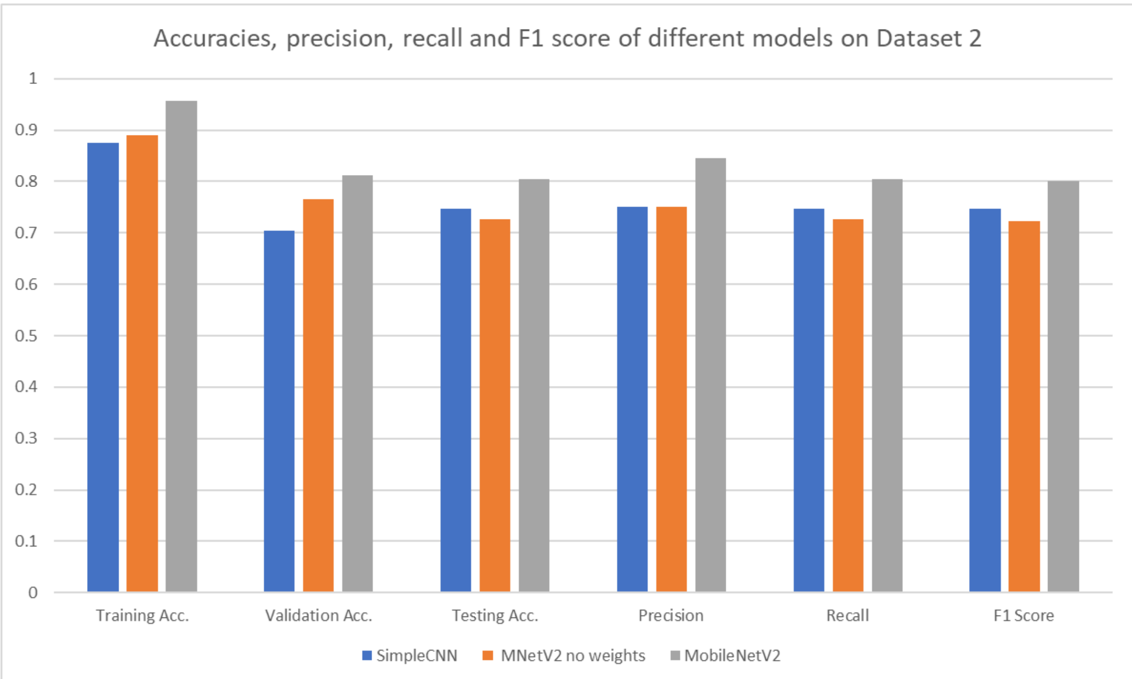

Figure 4: Accuracies, precision, recall and F1 score of different classifiers on Dataset 2 containing 2 classes

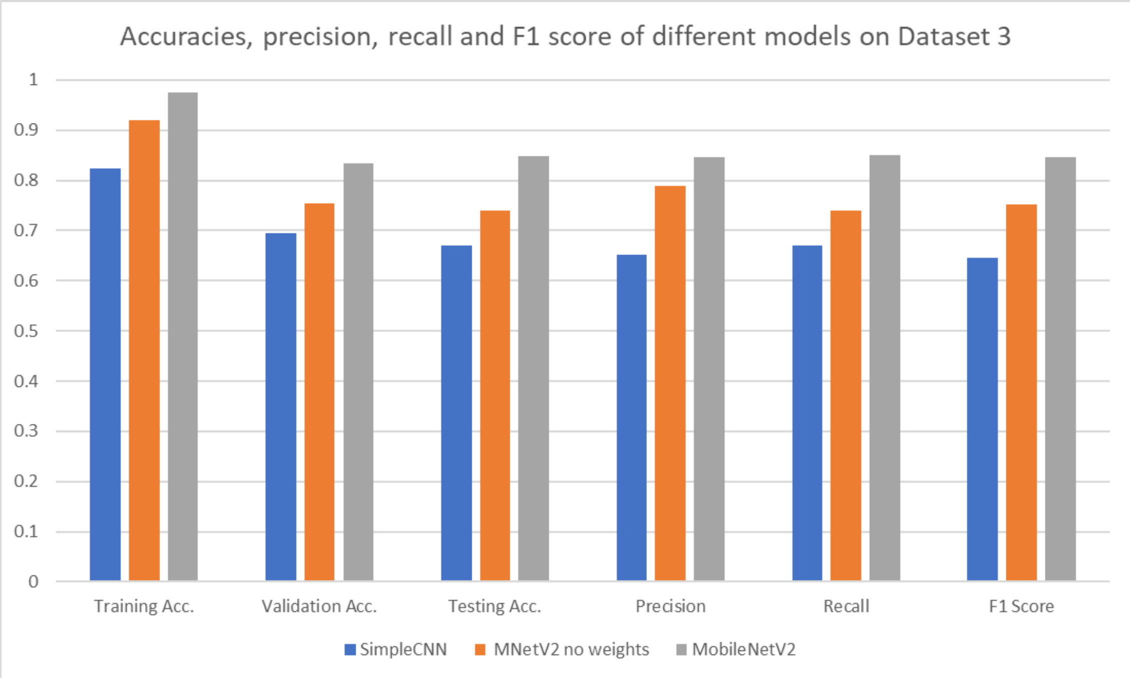

Figure 5: Accuracies, precision, recall and F1 score of different classifiers on Dataset 3 containing 3 classes

6 Discussion

In this research, we explored the use of CNN models for diagnosis of KOA using image classification. Furthermore, we explored the use of these models to classify the severity of KOA using knee joint X-ray images. The results that this classification without weights goes rather well, but there is still much to be desired comparing F1 scores.

One of our research aims was to develop a simple neural network that could match the state-of-the-art architectures used in deep learning today. We can say that for severity grading with use of Dataset 1, SimpleCNN falls behind significantly with only an F1 score of 0.46 in comparison to MobileNetV2 (with an F1 score of 0.52 for MNetV2 no weights and 0.55 for MobileNetV2). When performing binary classification, SimpleCNN and MNetV2 no weights, perform similarly. With equal precisions of 0.75. SimpleCNN slightly outperforms MNetV2 no weights here when comparing the f1 scores of 0.74 and 0.72 for SimpleCNN and MNetV2 no weights respectively, but this could be due to the overfitting that might occur sooner in MNetV2 no weights as this network has more parameters than SimpleCNN. This difference in trainable parameters can make a significant difference during training as the dataset these models were trained on, is relatively small. However, for binary classification, MobileNetV2 performs much better during diagnosis with an F1 score of 0.80. When we use these models for the next step, the severity classification. We can see that SimpleCNN starts to fall behind again, like it did during classification on Dataset 1. The SimpleCNN performs worst, followed by the MNetV2 no weights, followed by MobileNetV2. However, the lowest F1 score (which is for SimpleCNN) is still 0.64, which shows that this model does still classify most severities correctly.

Secondly, let us zoom in on the performance difference between the models with and without weights. What we can see is that MobileNetV2 is outperforming the other models in terms of Precision, Recall and F1 score. This suggests that the pretrained weights are still very useful for correctly training a KOA diagnosis model and a KOA classification model. Reasons for this could be, that it is rather difficult for each model to extract good features out of the model as there is not really any absolute cases and severities lay very close to each other. This lack of clean classes makes correct classification difficult. In addition, the dataset is rather unbalanced, with class 0 having 3252 images and class 4 having 251 images. This makes classifying the severities of classes 2, 3 and 4 to be more difficult for the models without pretrained weights. Furthermore, the dataset is rather small with around 5000 training images. This means that for training not a lot of parameters have to be used. Pretrained models train relatively few parameters, thus reducing the risk of overfitting. Overfitting occurred in all occasions, but for MobileNetV2, this occurred at a state where the losses were much lower than for the other models. The performance of MobileNetV2 on these datasets without any preprocessing on the data show that there might be features that are left out when cropping the images and performing histogram equalization on them. When comparing performance of the model to the performance of (Mohammed et al., 2023) of an accuracy of 84.7% compared to 89%, this is not much separation. Further exploration of the data is necessary.

Lastly, the performance of SimpleCNN is close to the performance of the different versions of MobileNetV2. The usage of SimpleCNN shows that for diagnosis of KOA by means of binary classification can be done by a simple cnn structure that is just the concatenation of convolutional and pooling layers. However, in order to perform at a level similar to the state-of-the-art, further improvements have to be considered to improve performance. For future work, the SimpleCNN can be improved. This can be done through experimentation with additional layers and different kernel sizes.

7 Conclusion

In conclusion, within this research we were able to develop a convolutional neural network from scratch that was able to perform diagnosis as well as severity classification of KOA. We were also able to compare the effect of using pretrained weights on a state-of-the-art model and saw that MobileNetV2 with use of ImageNet yielded a maximum classification accuracy of 57%, 80% and 84% on Dataset 1, Dataset 2 and Dataset 3 respectively. The result of the models without pretrained weights were not up to par with the model with pretrained weights but still show that there is potential for the use of other network structures.

In addition, the contribution of our work is that MobileNetV2 yielded good performance without any preprocessing of the data when compared to similar work. Therefore, further exploration of the dataset and what features can be extracted to provide a better understanding of KOA can be interesting for further studies within this field.

8 Additional Content

My prior experience when it comes to deep learning and computer vision was up until this point rather limited. In this period of my masters in Artificial Intelligence (AI), I have gained a lot of knowledge on ML and DL through the course ACML. In the end, I was able to apply it to this course, but that did not go without struggle. I have implemented many models, not all of which I have shown here in the report. What I learned from the most is that optimizing a model takes a lot of trial and error and that you cannot strike gold with every network architecture on this type of data. Human data is also rather hard to work with in relation to other, more "clean" data. Within this project I learned how to properly set up a model and train it and how data preprocessing can play a crucial role in the classification and learning of a model. Furthermore, I have learned how important computational resources are for project like this. My computer is definitely not a low-end one, but still it took a lot of time to run the experiments and test and tweak parameters to run different tests. In addition, due to time constraint I settled on working with MobileNetV2 as this is still fully trainable given the time-frame for this project.

The resources that I used mainly came from the studies related to this work that I found online. There are more classification methods for KOA and all have different implementations. Taking inspiration from these research papers was the main starting point and using Google as well as ChatGPT to ask questions about improving hyperparameters, reducing the risk of overfitting and many other questions helped me during this project.

8.1 Link To Video

Link to the video presentation: `https://youtu.be/HaFR6HPsAnk`

References

Chen, P., Gao, L., Shi, X., Allen, K., & Yang, L. (2019). Fully automatic knee osteoarthritis severity grading using deep neural networks with a novel ordinal loss. *Computerized Medical Imaging and Graphics*, *75*, 84-92. Retrieved from https://www.sciencedirect.com/science/article/pii/S0895611118304956 doi: https://doi.org/10.1016/j.compmedimag.2019.06.002

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Hsu, H., & Siwiec, R. M. (2022). *Knee osteoarthritis*. StatPearls Publishing, Treasure Island (FL). Retrieved from http://europepmc.org/books/NBK507884

Macri, E. M., Runhaar, J., Damen, J., Oei, E. H., & Bierma-Zeinstra, S. M. (2022). Kellgren/lawrence grading in cohort studies: methodological update and implications illustrated using data from a dutch hip and knee cohort. *Arthritis Care & Research*, *74*(7), 1179–1187.

Mohammed, A. S., Hasanaath, A. A., Latif, G., & Bashar, A. (2023). Knee osteoarthritis detection and severity classification using residual neural networks on preprocessed x-ray images. *Diagnostics*, *13*(8), 1380.

Osteoarthritis. (2023).

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 4510–4520).

Tiwari, S. (n.d.). Knee osteoarthritis dataset with severity grading. Retrieved from https://www.kaggle.com/datasets/shashwatwork/knee-osteoarthritis-dataset-with-severity/data

Vallés-Pérez, I., Soria-Olivas, E., Martínez-Sober, M., Serrano-López, A. J., Vila-Francés, J., & Gómez-Sanchís, J. (2023). Empirical study of the modulus as activation function in computer vision applications. *Engineering Applications of Artificial Intelligence*, *120*, 105863. Retrieved from https://www.sciencedirect.com/science/article/pii/S0952197623000477 doi: https://doi.org/10.1016/j.engappai.2023.105863