# Diagnosing Kidney Fibrosis in Mice Using Transfer Learning

Leonardo Daou[1], Manel R. Benabid[1]

[1]*Lebanese American University, Department Of Computer Science and Mathematics*

Abstract

Kidney fibrosis is a type of damage that affects the renal arteries. Early detection of kidney fibrosis is important to detect the progression of our kidney diseases. Transfer Learning has been widely used in detecting sharp features in medical images and has given promising results in making diagnoses especially since samples are not abundant, hence pretrained models are a solution. However, image quality still affects the outcome whether it be through experts' classification or machines' classifications. This is why we propose a method that combined transfer learning using VGGNet and a modification to its architecture. With our model, tests show that there is potential in this endeavour and can lead to better classifications.

**Keywords**

Image Classification, Machine Learning, Transfer Learning, Kidney Fibrosis

## 1. Introduction

Kidney fibrosis, or renal fibrosis, is estimated to predict disease progression. A pathological matrix deposition in the glomerular capillaries' walls causes this phenomenon [1]. The cells in which this disease occurs are important innate cells for immune surveillance that are responsible for regulating the inflammatory process. Fibrosis is known as the "dark horse" of kidney diseases [2] and is found that it could predict disease progression. The most common way so far to diagnose kidney fibrosis is through a kidney biopsy. Fibrosis biomarkers have gained importance in literature since is it the unifying feature in the progression of several renal diseases. The process of predicting kidney fibrosis through biopsy can carry many risks and is invasive hence it is often performed. In the last decade or so, Artificial Intelligence (AI) has been widely used in all different fields and has proven very efficient in the biomedical field at predicting the presence of a disease or the presence of potentially cancerous cells in the genome. Machine Learning (ML) and Deep Learning (DL) were found to be very useful when dealing with different types of data, and artificial neural networks (ANN) have been shown to be one of the best models for classification data, especially image data. The Biomedical field uses a wide range of scans such as X-rays, ultrasound, MRI, CT scans, Mammography, radiography, breast MRI, 3D ultrasound, and many more. The literature is full of research journals that leverage the power of ML and ANNs for the prediction and diagnosis of many diseases using different types of medical images with promising results which gives hope to diagnoses of other diseases that

leonardo.daou@lau.edu (L. Daou); manelroumaissa.benabid@lau.edu (M. R. Benabid)

could be spotted through imaging. As stated before, kidney fibrosis is important to identify other kidney disorders. It could also occur during the transplant of a donated kidney to a new host. When studied in mice, it is shown that it could be caused manually by causing Ischemia Reperfusion Injury (IRI) which can be done surgically by clamping the renal artery for a precise amount of time, then releasing the clamp and closing the abdomen of the mouse. The severity of the fibrosis increases the longer the renal artery is blocked. We seek to develop an algorithm to classify the degree of kidney fibrosis present in the kidney using ultrasound scans of the kidneys of mice.

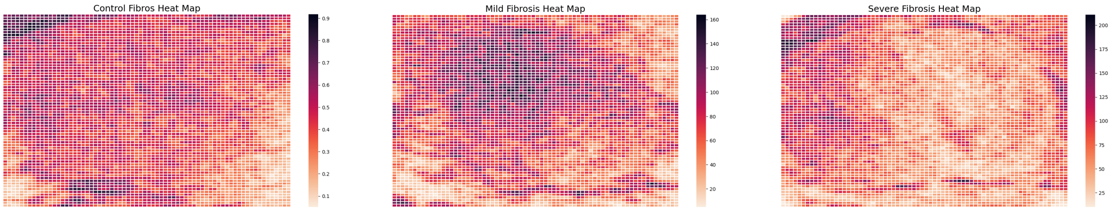
## 2. Literature Review

The literature indicates that ANNs are the most powerful at image classification due to their high capacity at identifying features in images. There are several types of ANNs including Feed Forward Neural Networks (FFNN), Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN), Long Short Term Networks (LSTM), and others. CNNs have been proven to be the best at handling image classification due to their tailored architecture to identify deep features in the images. CNNs are comprised of convolution layers, pooling layers, and Fully Connected (FC) layers as a classifier. Features are extracted in the convolution layers by applying kernels (filters) to the image. The features extracted by convolutions may not seem to make sense to humans but they do to the neural network. CNNs have different architectures, one of the state-of-the-art architectures are LeNet (citation), VGG-net (citation), ResNet (citation), and AlexNet (citation). These architectures are commonly used due to their high capacity at detecting the most important features to distinguish differences in classes. Although these state-of-the-art do a pretty good job at classifying images, medical images on the other hand each have different features and requires different types of pre-processing in order to make the algorithm achieve the absolute best way of distinguishing between features of different classes. Image pre-processing is a crucial part of the classification of medical data, one reason is that medical images are not of high resolution as the data we may obtain online or using a camera. Hence, image resolution enhancement is an important step before extracting features from these images. In addition, medical images are not abundant, either due to copyright, privacy, or other reasons. Thus, making our training a bit difficult since not much data is available. This means that new data must be generated from the already existing one. This step is called data augmentation. The latter is performed by rotating the image to a certain degree, flipping the image horizontally and vertically, pixel shifting, etc. Another important step that is usually implemented when dealing with medical images is extracting the regions of interest (ROIs). The aforementioned are the regions that fall under certain criteria and follow a certain shape of features that the algorithm should look for for the detection of the disease. ROIs are usually defined by medical experts and require domain knowledge. Another very important technique that researchers have been using in the past few years is Transfer Learning. Transfer Learning, as the name suggests, is transferring the knowledge learned obtained from a model and using it for other purposes of the same category. In manuscripts, transfer learning is basically using a pre-trained model (from the ones stated earlier or others) to obtain features from our data and classify them. Another notion is using an ensemble, first step is using the feature extraction part of the CNN model and using different classification models such as Support Vector Machines (SVM), Random Forests (RF), Decision Trees, and other classifiers. All of these techniques work together for one goal which is to improve the classification accuracy and tailor the process to our data to find the optimal model for classifying the disease (or other problems as well). Ul Haq et al.(2021)[3] proposed a model to predict breast tumors from mammograms by pre-processing the images using image enhancement (shrinkage masking) and removing noise to increase the quality of the image. Furthermore, they extracted ROIs and fed the images to a CNN model for feature extraction from the ROIs, they implemented a method called feature fusion and an ensemble of SVM, RF, and two FC layers with a sigmoid classifier. Finally, they decided on the class of the image on a majority vote from the classifiers. They obtained an accuracy of 0.994. Gupta et al.(2022) [4] designed a Spotted Hyena Aquila (SHyAq) optimization-tuned deep CNN classification model to classify porn images from websites. They obtained an accuracy of 0.9646 and outperformed other state-of-the-art techniques. Murthy et al.(2021)[5] designed an adaptive fussy deformable fusion and an optimized CNN model with ensemble classification for brain tumor diagnosis. Their model merged the concept of fuzzy c-means clustering and Snake deformable approach. They also used Deer Hunting optimization. The authors used contrast enhancement approaches for tumor segmentation and used traditional image augmentation techniques. The best accuracy obtained by the model is 0.9486. Xue et al.(2020)[6] classified cervical histopathology images using a technique empowered by transfer learning and ensemble learning. They developed four transfer learning structures: Inception-V3, Resnet-50, VGG-16, and Xception. They obtained an accuracy of 0.97 on the majority vote of different classifiers of the ensemble. Snider et al.(2022)[7] designed a CNN classification model to detect shrapnel in ultrasound images. They obtained a ROC of 0.95 and an accuracy of 0.9 for each class.

## 3. Materials and Methods

### 3.1. Data

In order to solve this problem, data was collected from 50 mice which were then divided into 3 groups depending on their kidney fibrosis status which are control, mild and severe. The number of mice for each group is respectively 15, 15 and 20. Ultrasound images were taken for each mice with 100 frames for every one. However, since the kidney volume does not fully show up in all frames, frames 30 to 90 were initially taken to then be reduced to only 11 per mice. However, some mice were sacrificed during the experimental stage which left 31 mice. This gives 88 images for control, 99 for mild, and 154 for severe. In order to understand if we have any ROI or useful insights towards the data-set, heat-maps were plotted over images from all 3 cases. They were first normalized and turned into gray-scaled in order to understand some concepts from the intensity of the pixels. As we can see in figures 1 a to c, we can see that depending on the intensity of the pixels, we can gleam that there is a clear difference between two groups which are control/mild and severe fibrosis. The light spots correspond to dark parts of the image which informs the onlooker that there is a point of interest in that region. However, the difference between the two other stages which are control and mild do not have a high difference which can be seen from the noise affecting figure 1.a in the center. Due to this fact, we tried a couple of preprocessing steps to check whether they can be enhanced. We



(a) Control (b) Mild (c) Severe

**Figure 1:** Heatmaps representing the pixel intensity of the three stages of kidney fibrosis.

tried denoising the data using a denoising convolutional neural network (DnCNN) which had been pretrained), sharpening the images and a combination of both. Data augmentation was also used in order to have more training data. The images were flipped both vertically and horizontally, and they were zoomed in by 0.1 to reach a total of 1023 images among which 264 were for control, 297 were for mild, and 462 were for severe. The data was also first normalized to values between 0 and 1, whether we are testing on the enhanced images or on the original ones. They were then standardized. The data was also split between 70% that will be used for training and 30% that will be used for testing our model later on.

### 3.2. Methodology

Since we do not have a big amount of data, we proposed to use transfer learning since the weights will have some idea of the type of data used which are images. This is why convolutional neural networks (CNN) since they excel at recognizing features contained in images. It is a neural network comprised of convolutional layers which will work as feature extractors and give us new transformed matrices which have the features discovered by the convolutions done in that layer. It will also have some pooling layers which will help reduce the number of pixels in an image while keeping the features needed. This is done to reduce computation time since each pixel is normally one feature and considering the number of pixels per image, a machine learning model would take a lot of time to be trained. After all convolutions are done, fully connected layers are then added to be able to interpret the values obtained from the convolutions. This is the reason why transfer learning was used. It already has the weights of the convolutional layers trained to be able to grasp some features. Even if the images differ and the content is different, some concepts like edges will not differ. This will also save the model time since these feature extractors can recognize basic features. Some pretrained models like VGGnet already have seen millions of images on ImageNet, which means we can be sure that they will be able to recognize basic features like edges.

## 4. Proposed Model

For the purpose of classifying these images into their correct label, we proposed an approach inspired by [8]. This was brought about by the fact that the image quality is bad, there is a lot of noise detected and there are not many features to be selected, thus a smaller architecture was needed. This is why VGG11 was used for the transfer learning which was pretrained on the ImageNet dataset. Following figure 2, we can see that it is composed of 8 convolution layers with 64, 128, 256 and 512 filters with the same kernel size and ReLU activation function. Max pooling was applied to the output of the convolutions. The part that we replaced is the fully connected layers to use two hidden layers with 500 and 200 neurons. They both had sigmoid as an activation function, with the first layer being followed by a ReLU activation function layer to combat the vanishing gradient problem, and had a dropout of 0.6 each. [9] figured that for this type of network when we have a severe overfitting problem, that a dropout of 0.5 helped reduce it and increase the accuracy. We elevated the dropout rate to 0.6.

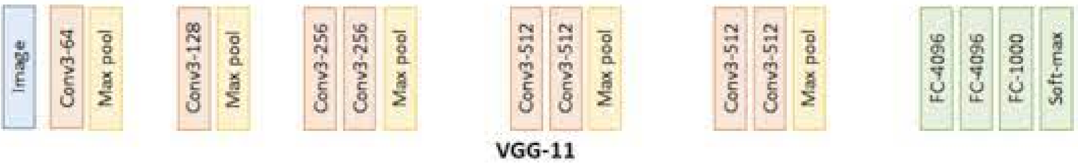


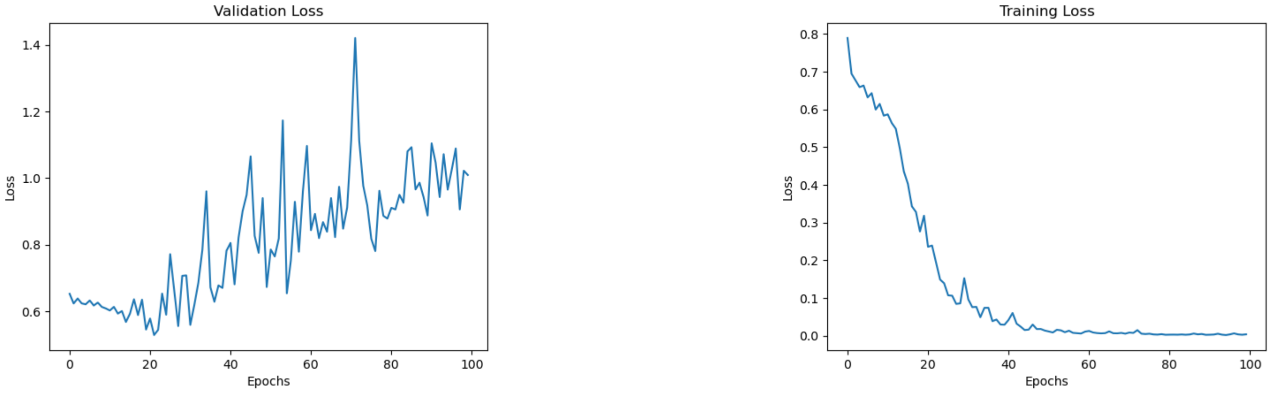
**Figure 2:** VGG11 Architecture

All of the network had the same optimizer which was SGD and a momentum of 0.9. However, the learning rate was changed between the convolutional part and the fully connected layers. The first part mentioned had a learning rate of 0.001 and decayed to 0.0001 through training since we are fine-tuning the convolution layers. The fully connected layers had a higher learning rate of 0.01. The model had a binary corssentropy as the loss function. The simpler model was preferred since VGG16 and VGG19 did not perform as well as VGG11 and often caused vanishing gradients which is the case when we have more layers than needed and, in our case, we did not have that many features to extract. This is why less number of convolutions were better.

## 5. Experiment and Results

As mentioned before, the data was split into 70% for training and 30% for testing. The augmented data was then passed to the model for training. The problem in this case with the data as was seen in figure 1 is that it is noisy and will likely give rise to overfitting. No amount of preprocessing was able to increase the accuracy that we reached and instead deteriorated it since some features were deleted by the enhancement process since the noise was very close to them in spatial coordinates in the image. In figure 3 and figure 4, we can observe the overfitting that happened over the number of epochs. The training loss has a very smooth curve whether it be for accuracy or loss. However, when we examine the validation (test) loss, we can see that the loss is increasing while the accuracy is also increasing but unstably which tells us that overfitting is occurring. However, this is expected since the state of the data is not good and we cannot clean it without removing crucial features. This further impacted the accuracy values per epoch where we can see that initially both the training and the testing had similar values but the training accuracy quickly overtook it to reach a difference of 0.3.

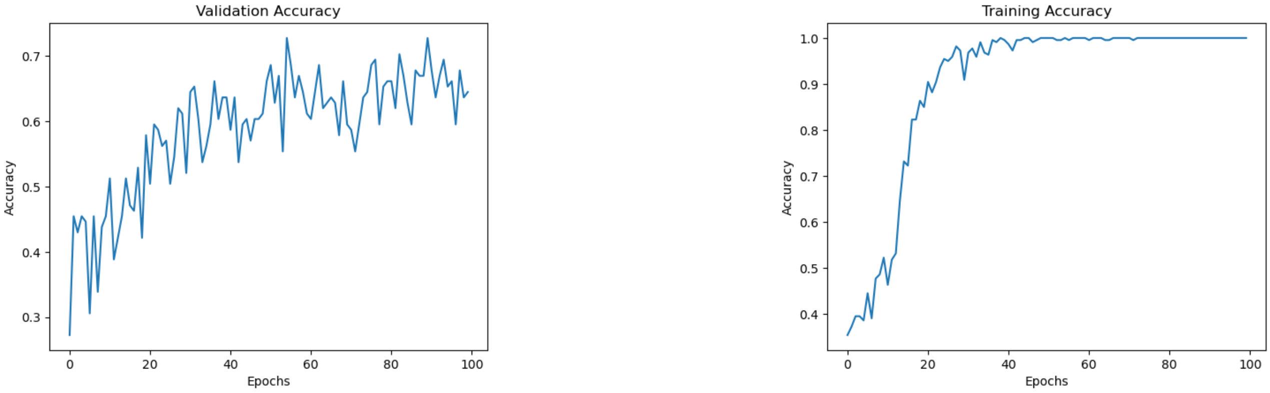
Regarding the results obtained from the three trials on the testing set which can be seen on



(a) Testing Loss

(b) Training Loss

**Figure 3:** Testing Loss vs Training Loss



(a) Testing Accuracy

(b) Training Accuracy

**Figure 4:** Testing Loss vs Training Loss

Table 1, we can observe that the AUC score has an average of 0.75 with a low standard deviation of 0.199. This is almost the same for all metrics in regards to standard deviation. It did not go lower since we have different splits as well as different weight initialization with noisy data. The transfer learning part coming from the convolutional layers of the model helped reduce it since at least for them, we had the same start line. The loss is another matter since it is no scaled from 0 to 1 since we are dealing with binary crossentropy but it did fluctuate between the best run which is trial 3 and the rest. Overall, all four measures of precision, recall, accuracy and f-measure had around the same value for both average, with around 0.68, and standard deviation of around 0.028. If we look into each case of sham/control, mild and severe in Tables 2,3 and 4, we can see that regarding recall, it was the highest for mild conditions, then for severe and lastly for control cases. This is beneficial for practical medical cases since we care more about having an accurate prediction of the positive cases to be able to help them. This was also

|  | Binary Crossentropy Loss | AUC | Precision | Recall | Accuracy | F-measure |
|---|---|---|---|---|---|---|
| trial_1 | 3.441931 | 0.732828 | 0.666667 | 0.661157 | 0.661157 | 0.663900 |
| trial_2 | 3.441931 | 0.749495 | 0.666667 | 0.661157 | 0.661157 | 0.663900 |
| trial_3 | 2.932015 | 0.772475 | 0.716667 | 0.710744 | 0.710744 | 0.713693 |
| Average | 3.271959 | 0.751599 | 0.683333 | 0.677686 | 0.677686 | 0.680498 |
| Stdev | 0.294400 | 0.019907 | 0.028868 | 0.028629 | 0.028629 | 0.028748 |

**Table 1**
Metrics per trial

|  | Precision for sham | Recall for sham | Accuracy for sham | Specificity for sham | F-measure for sham |
|---|---|---|---|---|---|
| trial_1 | 0.750000 | 0.545455 | 0.826446 | 0.931818 | 0.631579 |
| trial_2 | 0.785714 | 0.666667 | 0.859504 | 0.931818 | 0.721311 |
| trial_3 | 0.681818 | 0.454545 | 0.793388 | 0.920455 | 0.545455 |
| Average | 0.739177 | 0.555556 | 0.826446 | 0.928030 | 0.632782 |
| Stdev | 0.052787 | 0.106421 | 0.033058 | 0.006561 | 0.087935 |

**Table 2**
Metrics for control cases per trial

|  | Precision for mild | Recall for mild | Accuracy for mild | Specificity for mild | F-measure for mild |
|---|---|---|---|---|---|
| trial_1 | 0.600000 | 0.636364 | 0.785124 | 0.840909 | 0.617647 |
| trial_2 | 0.571429 | 0.727273 | 0.776860 | 0.795455 | 0.640000 |
| trial_3 | 0.805556 | 0.878788 | 0.909091 | 0.920455 | 0.840580 |
| Average | 0.658995 | 0.747475 | 0.823691 | 0.852273 | 0.699409 |
| Stdev | 0.127727 | 0.122468 | 0.074073 | 0.063270 | 0.122767 |

**Table 3**
Metrics for mild cases per trial

reflected in the F-measure score since both mild and severe cases predictions had higher values than control cases. The standard deviation for the severe cases was the lowest which can be attributed to the features we saw in the heatmaps where it was clear which one was severe so its performance was more stable for the severe cases. However, the accuracy for severe was instead the lowest of the three cases with the accuracy of sham and mild cases being practically the same. If we check the results for each mice alone in the supplementary folders Results, then we can see that some images of the mice were completely correctly classified while some were the opposite and not one was correctly classified, though these are rarer. This can be attributed to the quality of the images and the noise that made it harder for the model to differentiate between the cases that we had. There is also the fact that some frames are better than the others, which correlates to the fact of the third case where some were correctly classified and some were not. The previously observed phenomena on all images pretaining to the cases and the respective metrics was also, on average, observed when we looked at each mouse.

|  | Precision for severe | Recall for severe | Accuracy for severe | Specificity for severe | F-measure for severe |
|---|---|---|---|---|---|
| trial_1 | 0.672131 | 0.745455 | 0.719008 | 0.696970 | 0.706897 |
| trial_2 | 0.680000 | 0.618182 | 0.694215 | 0.757576 | 0.647619 |
| trial_3 | 0.677419 | 0.763636 | 0.727273 | 0.696970 | 0.717949 |
| Average | 0.676517 | 0.709091 | 0.713499 | 0.717172 | 0.690821 |
| Stdev | 0.004011 | 0.079253 | 0.017204 | 0.034991 | 0.037820 |

**Table 4**
Metrics for severe cases per trial

## 6. Conclusion and Future Work

The classification of kidney fibrosis, which posed a problem for experts' classification, proved to be a challenge even for machine learning models. This is due to the fact that these images that have been used have a low quality. The model developed should at least help guide professionals in their decision as to which stage the fibrosis belongs to. The current images that we can get for diagnosis need to be enhanced so more accurate classifications can be made as well as give more features to the image. This was another challenge which did not allow for more complex methods or deeper networks. Hence, the choice to use shallower network than usual for image classification and transfer learning. This leads to potential future work which is the use of search base algorithm to directly modify the structure of the network as training is being made as well as to try an ensemble of this combination while keeping shallow networks.

## References

[1] D. Zhou, Y. Liu, Understanding the mechanisms of kidney fibrosis, Nature Reviews Nephrology 12 (2016). doi:10.1038/nrneph.2015.215.

[2] D. Zhou, Y. Liu, What can target kidney fibrosis?, Nephrology Dialysis Transplantation 32 (2017). doi:10.1093/ndt/gfw3885.

[3] I. U. Haq, H. Ali, H. Y. Wang, C. Lei, H. Ali, Feature fusion and ensemble learning-based cnn model for mammographic image classification, Journal of King Saud University – Computer and Information Sciences 34 (2022). doi:10.1016/j.jksuci.2022.03.023.

[4] J. Gupta, S. Pathak, G. Kumar, A hybrid optimization-tuned deep convolutional neural network for bare skinned image classification in websites, Multimedia Tools and Applications 81 (2022). doi:10.1007/s11042-022-12891-3.

[5] M. Y. B. Murthy, A. Koteswararao, M. S. Babu, Adaptive fuzzy deformable fusion and optimized cnn with ensemble classifcation for automated brain tumor diagnosis, Biomedical Engineering Letters 12 (2022). doi:10.1007/s13534-021-00209-5.

[6] D. XUE, X. ZHOU, C. LI, Y. YAO, An application of transfer learning and ensemble learning techniques for cervical histopathology image classification, IEEE Access 8 (2020). doi:10.1109/ACCESS.2020.2999816.

[7] E. J. Snider, S. I. Hernandez-Torres, E. N. Boice, An image classification deep-learning

algorithm for shrapnel detection from ultrasound images, Scientific Reports 8427 (2022). doi:10.1038/s41598-022-12367-2.

[8] D. Meng, L. Zhang, G. Cao, W. Cao, G. Zhang, B. Hu, Liver fibrosis classification based on transfer learning and fcnet for ultrasound images, IEEEAccess 5 (2017). doi:10.1109/ACCESS.2017.2689058.

[9] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, R. R. Salakhutdinov, Improving neural networks by preventing co-adaptation of feature detectors, Computer Science (2012). doi:10.48550/arXiv.1207.0580.