Department of Electrical Engineering
Indian Institute of Technology Bombay

Automated Detection of Prosthetic Knee Implants in X-rays: A Multiresolution Analysis and Machine Learning Approach

As Part of Dual Degree Thesis

Aditya Anavkar

19D070004

May 2024

under the guidance of

Prof. Vikram Gadre

Contents

Abstract

This report details the development of a Convolutional Neural Network (CNN) and other image processing techniques utilized to automatically detect knee replacements (KRs) in X-ray images, a crucial step before revision surgery for various orthopedic issues. The dataset comprises 800 augmented knee X-rays, partitioned randomly into training, validation, and testing sets with a 60:20:20 split. A custom CNN model was trained, and the best-performing model was chosen based on validation outcomes. Additionally, multi-resolution analysis (MRA) and scattering transforms were employed to enhance the model's feature extraction capabilities. MRA enabled the decomposition of images into multiple scales, distinct details, while the scattering transform provided a robust, invariant representation of image features and making the model less computationally expenseive. Finally, the selected model underwent testing using a separate hold-out test dataset to assess its performance which gives 98% accuracy on test data.

Future investigations would gain from expansive datasets encompassing a broader range of implant variations, enhancing the algorithm's robustness. It's important for such an algorithm to possess the capability to identify implants beyond its trained parameters to ensure safe deployment in clinical settings. Addressing these concerns and constraints holds promise for reducing radiologists' workload and instances of missed pre-operative implant identification, thus streamlining re-operative procedures and enhancing overall clinical results.

1 Introduction

Accurate identification of implants before revision surgery is crucial for effective pre-operative planning and equipment preparation [1]. However, this task can be challenging, especially when the implant type is unknown or primary operative records are unavailable. Leveraging machine learning and artificial intelligence algorithms for automated detection of orthopedic implants on X-ray images holds promise for supporting clinicians in making real-time decisions. Osteoarthritis, a prevalent condition causing joint pain and deformity, often necessitates joint replacement surgery [2], which can significantly improve function and alleviate pain. Successful joint replacement procedures can last for decades, highlighting the importance of reliable implant detection methods [3,4]. Deep learning has shown diagnostic abilities on par with human experts in imaging diagnoses of medical conditions like diabetic retinopathy, malignant melanoma, and tuberculosis. It has proven similarly effective for diagnosing fractures and assessing pediatric bone age.

Convolutional Neural Networks have become one of the popular modes of computer vision, they utilize convolutional kernels and pooling layers to extract features, and then forward it to a fully connected layer prior to classification [5]. CNNs are now used in all aspects from classification, to image segmentation and even in automated car systems without human intervention. Multiresolution analysis (MRA) is [6] a mathematical method applied in signal processing and data analysis to decompose data into various levels of detail, ranging from coarse to fine granularity. MRA accomplishes this by splitting data into finer and coarser perspectives through a process termed "pyramid decomposition" or "scale-space decomposition." The core concept behind MRA is that it enables the examination of data at multiple levels of detail, facilitating the extraction of pertinent information and the execution of tasks such as denoising, compression, and feature extraction. By dividing data into coarser and finer perspectives at multiple scales, it becomes possible to capture both global and local characteristics. In this research, machine learning algorithms are employed in conjunction with Multi-Resolution Analysis of data obtained from the MWR for classification and detection of presence of knee replacement implants.

There are some concerns regarding the use of complex mathematical models in clinical decision making, major issue being with Neural Networks not being explainable. Although the basic concepts and maths behind CNNs are straightforward, the networks themselves are becoming increasingly complex. In this study, we have addressed this criticism by adding a visualization method to determine the weights given to each pixel/ part of the image. We shall be using a technique known as saliency maps or heatmaps to pinpoint which areas of the image are most crucial for a given prediction [7].

In this study, we aim to develop and evaluate the performance of a deep learning system for the automated radiographic detection of Knee Arthroplasty.

2 Methods

2.1 Dataset

In the collected dataset we had 40 images corresponding to anteroposterior knee replacement Implants being present and 40 corresponding to normal knee X-rays. I would like to express thanks to Dr Anirudh Nene for the dataset and some more images were procured from publicly available dataset at the National Institutes of Health. All the images were resized to 150x150 for passing on to the neural network.

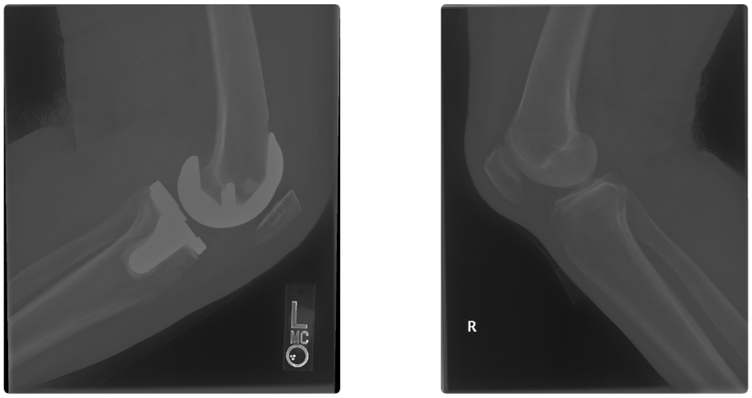

Figure 1: Left: Implant X-ray, Right: No prosthesis X-ray

The images were then split into train, validation and test, in the ratio 60:20:20. All the images were then augmented using standard techniques such as cropping, flipping, zooming, and rotations (max rotation of 30° with 50 percent probability). In total, we had 800 images, out of which 480 were used for training the mode, 160 for validation, and 160 for testing.

Class	All	Training	Validation	Testing
Implant X-ray	400	240	80	80
Normal X-ray	400	240	80	80

Table 1: Distribution of images for deep learning model development

Sample images of anteroposterior X-rays of the knee are shown in the Figure 1. On the left we have X-ray with an implant present while on the right we have a normal knee X-ray

2.2 Model

In the Machine Learning realm, we often come across the problem of finding the best-fit model for our task. A large model gives better performance but is computationally expensive and prone to overfitting, whereas a smaller model may fail to generalize on the given data. Thus after a bit of experimentation, we have come up with a custom architecture that accepts 150 x 150 x 3 input images which finds a balance between the two. It has 3 convolutional layers followed by max pooling layers and is then flattened into a fully connected layer with a single dense output (0 denoting normal X-ray, 1 denoting Implant X-ray). We have used 'Relu' activation function in all the convolutional layers and sigmoid for the final layer. The initially presented model has 409 K parameters, with its structure as shown in Figure 2. The training hyperparameters were set as follows, 50 epochs, batch size 32, ADAM optimizer with a learning rate of 0.001 and momentum of 0.85. To identify the distinguishing features in each X-ray that the network uses for classification, we generated heatmaps using class activation [8]. This technique visually emphasizes the different areas in an image that the considers important for its classification decision.

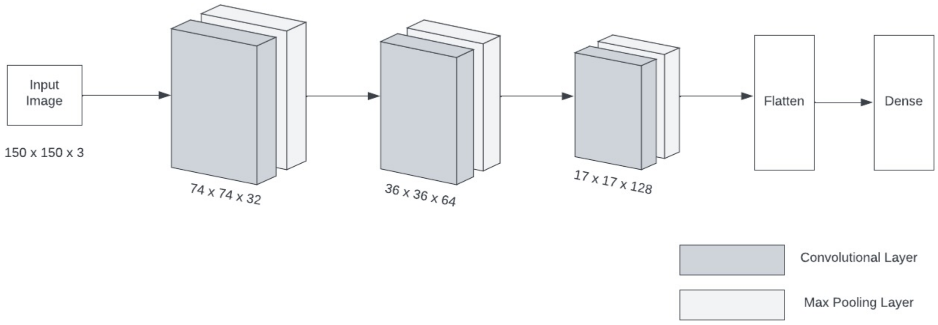

Figure 2: Deep Learning Model Architecture

2.3 Multi Resolution Analysis

Multi-resolution analysis (MRA) is a pivotal technique in image processing. It involves decomposing images into multiple scales or resolutions, each revealing different levels of detail. This decomposition is often achieved through methods such as wavelet transforms or pyramid representations. Wavelet transforms, for instance, break down images into approximation and detail coefficients across various scales, while pyramid structures like Gaussian or Laplacian pyramids organize images into hierarchical layers of decreasing resolution.

In the realm of 2D image classification, MRA proves indispensable by enabling feature extraction across multiple scales. This capability is crucial because different features in images, such as textures and edges, may manifest more distinctly at specific resolutions. By incorporating information from various scales, classifiers can derive a more comprehensive and nuanced understanding of the image content, thereby enhancing classification accuracy. Another significant advantage of MRA lies in its ability to enhance the robustness of classification models against scale variations. In practical applications, images often vary in scale due to factors like distance or object size variations. MRA equips classifiers with the capability to adapt to these variations by incorporating features that are invariant or robust across different scales, thereby improving generalization to unseen data. We will utilize the scattering transform to extract features, leveraging its capability to achieve invariance to geometric transformations. This approach ensures robust feature representation, improving the accuracy and generalization of our models in detecting patterns and structures within the data.

2.4 Scattering Transform

The scattering transform is primarily a technique used for analyzing and processing two-dimensional signals, such as images. It extends concepts from wavelet analysis but operates differently in terms of its theoretical framework and computational approach.

The scattering transform is a nonlinear method for signal representation that achieves invariance to geometric transformations while preserving distinguishable features. It effectively removes variations due to translations, rotations (in the case of 2D or 3D signals), frequency shifts (for 1D signals), and changes in scale that are often irrelevant to many classification and regression tasks. By transforming signals into their scattering representation, unnecessary variability is reduced while essential structural information for specific tasks is retained. This simplifies model development, especially beneficial when working with limited training data. The scattering transform uses fixed wavelet filters and applies a complex modulus as its nonlinearity. Each network layer functions as a wavelet transform, separating signal scales. The wavelet transform's contractive nature, along with the modulus operation, ensures that the entire network remains contractive. This results in reduced variance and increased stability against additive noise. The scale separation provided by wavelets also enhances stability against signal deformations. These characteristics make the scattering transform particularly suitable for representing structured signals such as natural images, textures, audio recordings, biomedical signals, and molecular density functions.

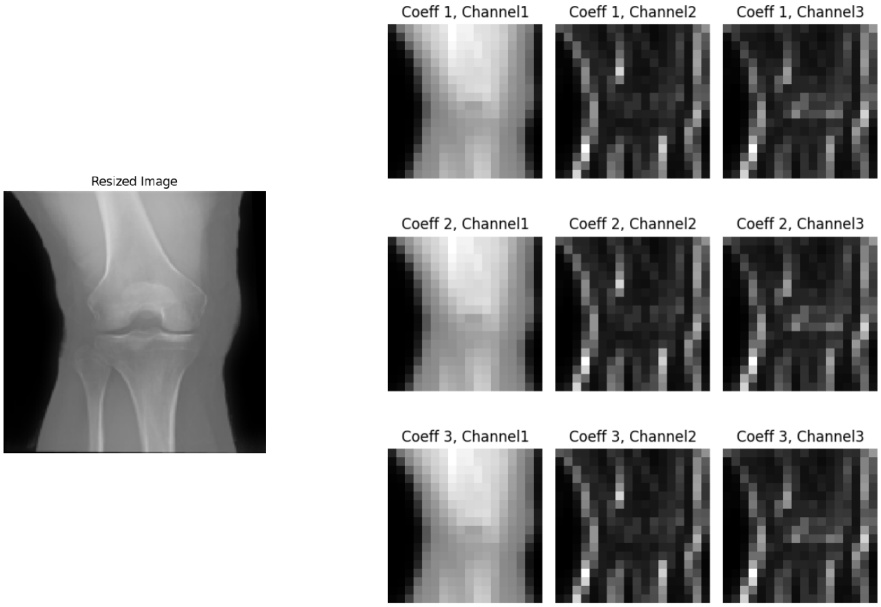

Figure 3: Normal Knee X-ray (left) and its Scattering Transform (right)

In essence, the scattering transform computes a cascade of wavelet transforms and modulus operations across different scales and orientations. This process helps capture invariant features in the input signal, making it robust to various transformations such as translations and deformations. These features are aggregated into a multi-scale representation that can be used for tasks like image classification, texture analysis, or image synthesis.

In the context of the scattering transform, J and L refer to parameters that define the architecture and behavior of the transform:

J: represents the number of scales considered in the scattering transform. It determines how many times the original signal is transformed using wavelets, each time capturing features at a different scale. Increasing J allows the transform to capture more intricate details at finer scales but also increases computational complexity.

L: denotes the number of layers in the scattering transform network. Each layer consists of a wavelet transform followed by a modulus operation. These operations iteratively extract and combine features across different scales. Increasing L leads to a deeper representation of the signal, potentially capturing more complex relationships and hierarchies in the data.

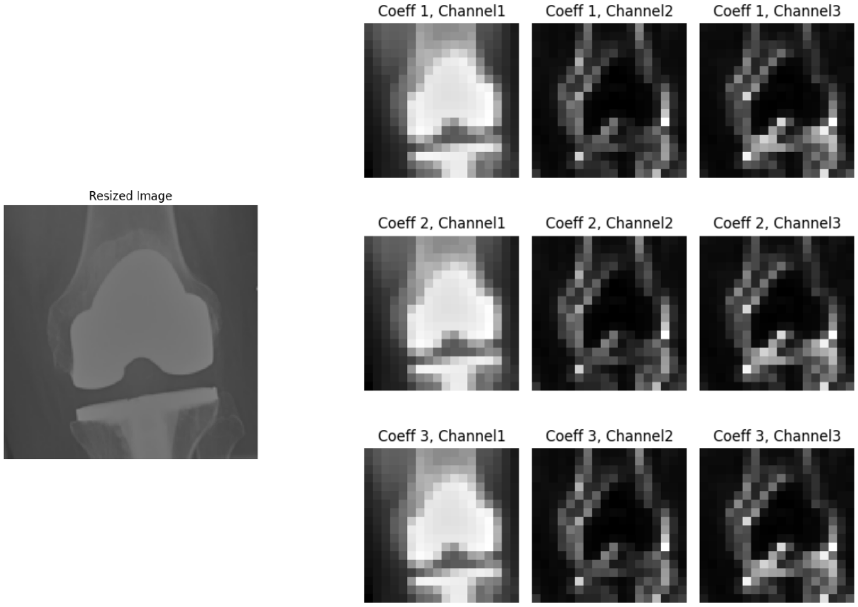

Figure 4: Implant Knee X-ray (left) and its Scattering Transform (right)

Let us assume that x is a tensor of size (C, N1, N2). Then the output Sx via a Scattering Transform with scale J and L angles and m order 2 will have size:

$$\left(C, 1 + LJ + \frac{L^2 J(J-1)}{2}, \frac{N_1}{2^J}, \frac{N_2}{2^J}\right)$$

Thus when applied to our input image of size 150 x 150 x 3 reshpaed to (3,150,150) with J = 2, we get output tensor of the shape (3, 217, 18, 18). We can utilize some of these channels, for our purpose of classification, we shall be using 3 channels thus reducing the dimensions to 18 x 18 x 3. The scattering transform of a normal knee X-ray is visualized in Figure 3 and scattering transform an implant X-ray is visualised in Figure 4. As we can see in Figure 4, the implant is still largely discernible and its key features such as shape, contrast from the background are still preserved. This preservation of key features in the scattering transform is crucial for the model's ability to accurately identify implants.

2.5 Scattering Convolutional Neural Network

Here we have proposed a second model bringing scattering transform and CNN together called Scattering CNN (SCNN). It involves first decomposing the input images via scattering transform and then passing on 3 channels for classification task into the CNN. The scattering transform has fixed filter thus it does not have any additional trainable parameters. Thus due to the reduced input dimensions we are able to bring the total number of parameters down to 108 K which is around one-fourth of the original model size. The SCNN architecture is as shown in Figure 5.

It is important to note that while being useful for feature extraction scattering transform is computationally expensive due to its complex multi-scale, multi-orientation wavelet decompositions. Each input signal undergoes multiple levels of wavelet transforms, capturing features at various scales and orientations,

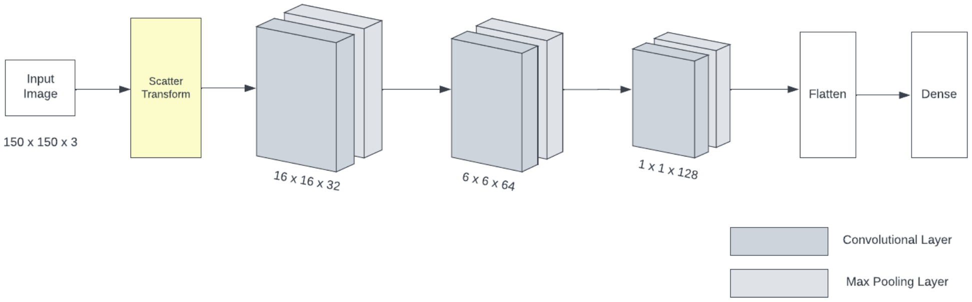

Figure 5: Scattering Convolutional Neural Network (SCNN) Architecture

which requires substantial computational resources. The comparison between the two models is shown in results section.

2.6 Saliency mapping

One significant critique of CNNs and neural networks is the opacity of their decision-making processes. It is often challenging, if not impossible, to fully understand how a deep, intricate network with numerous weights arrives at its classification conclusions. Saliency maps address this issue by highlighting the regions of an input image that the CNN focuses on when making decisions. These maps emphasize the most critical areas, providing a visual summary of the system's reasoning that is readily interpretable by humans.

To visualize class activation maps for debugging deep neural networks we have implemented Grad-CAM using Keras and TensorFlow.

Algorithm 1 Grad-CAM for Visualizing Class Activation Maps

```
procedure GRADCAM($model$, $input\_image$, $conv\_layer\_name$, $pred\_index$ = None)
    Make Predictions
        with tf.GradientTape() as tape do
            $conv\_layer\_output$, $predictions \leftarrow grad\_model(input\_image)$
            if $pred\_index$ is None then
                $pred\_index \leftarrow \arg\max(predictions)$
            end if
            $class\_output \leftarrow predictions[:, pred\_index]$
        end with

    Compute and Average Gradients
        $grads \leftarrow tape.gradient(class\_output, conv\_layer\_output)$
        $pooled\_grads \leftarrow$ tf.reduce_mean($grads$, axis=(0, 1, 2))

    Weight Channels by Importance
        $conv\_layer\_output \leftarrow conv\_layer\_output[0]$
        for $i = 1$ to len($pooled\_grads$) do
            $conv\_layer\_output[:, :, i] \leftarrow conv\_layer\_output[:, :, i] * pooled\_grads[i]$
        end for
        $heatmap \leftarrow$ tf.reduce_sum($conv\_layer\_output$, axis=-1)

    return $heatmap$
end procedure =0
```

The implemented function generates a heatmap to visualize the areas of an input image that a Convolutional Neural Network focuses on when making a prediction, helping to understand the model's decision-making process. Initially, a new model is created, mapping the input image to both the output of the last convolutional layer and the final predictions of the original model. This is done using to extract the necessary outputs. The function then computes the gradient of the top predicted class with respect to the activations of the last convolutional layer. If no specific prediction index is provided, the function selects the index of the highest predicted class. The output corresponding to this class is isolated, and the gradients are calculated to show how changes in the activations affect the selected class's score.

After computing the gradients, we average them across the width and height dimensions to produce a vector that indicates the importance of each feature map channel for the predicted class. Each channel in the last convolutional layer's output is multiplied by its corresponding average gradient value, weighting the channels by their importance to the prediction. These weighted channels are then summed to produce the initial heatmap. For better visualization, the heatmap is normalized to a range of 0 to 1 by setting negative values to 0 and dividing by the maximum value. The final heatmap highlights the areas of the image most influential in the model's decision, providing a visual interpretation of the model's focus and decision-making process.

3 Results

For the initially proposed CNN model, the training, validation loss and accuracy plots are shown in Figure 6. This graph shows the training and validation accuracy over 50 epochs, where the training accuracy quickly reaches nearly 100%, demonstrating effective learning on the training set. The validation accuracy, despite some fluctuations, stabilizes around 80-90%, indicating good generalization and potential for further optimization.

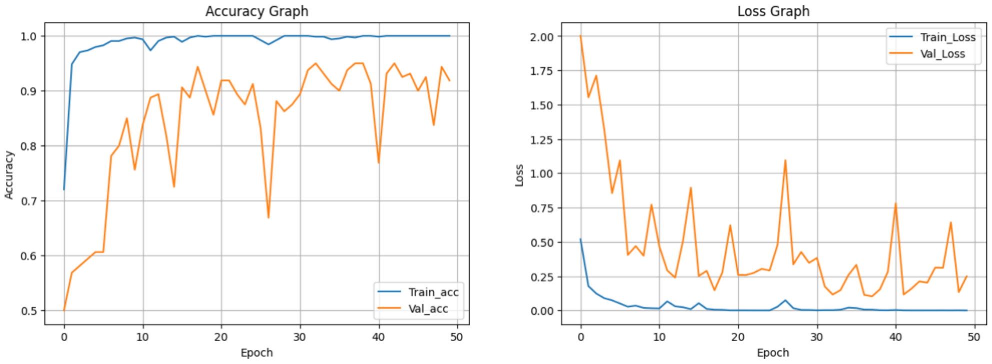

Figure 6: CNN Model Training accuracy and Loss across epochs.

In assessing our current model's performance, it's evident from the graph that while increasing model complexity might initially seem beneficial, it exacerbates overfitting. The graph depicting the CNN model reveals a divergence between training and validation losses, indicating that the model learns to fit the training data too closely, resulting in poor generalization to unseen data. To address this, scaling back model complexity becomes crucial. This approach aims to strike a balance where the model is powerful enough to capture underlying patterns in the data without excessively tailoring its parameters to noise or specific features of the training set.

Comparatively, the SCNN model depicted in Figure 7 demonstrates a more favorable trend. Here, both training and validation losses converge and saturate at a lower value after approximately 25 epochs. This convergence suggests that the SCNN model achieves better generalization, maintaining lower losses on both training and validation datasets. By simplifying the architecture and employing techniques like feature aggregation through scattering transforms, the SCNN effectively manages to extract relevant features while mitigating the risk of overfitting. This outcome underscores the importance of model selection and architecture design tailored to the specific complexities and nuances of the dataset, aiming to enhance both training efficiency and generalization capability.

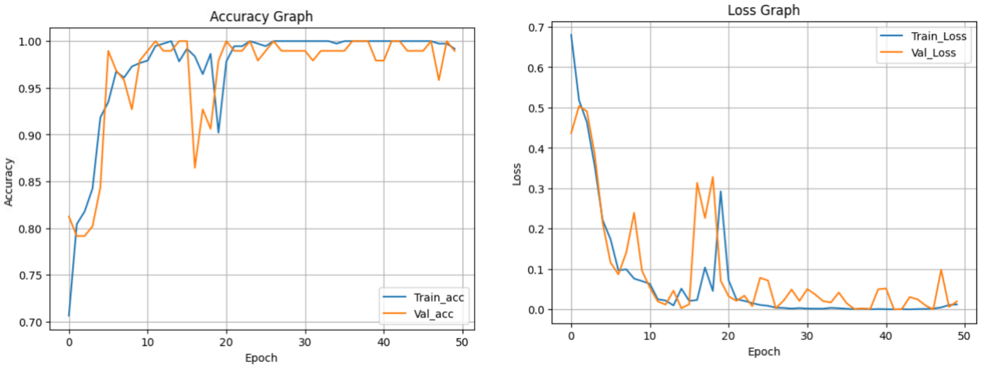

Figure 7: SCNN Model Training accuracy and Loss across epochs.

The top-left heatmap in Figure 8 emphasizes regions with higher probabilities of implant presence, particularly around the actual implant area in the adjacent X-ray. The concentration of high-intensity values (shown in yellow and green) directly corresponds to the implant's location, demonstrating the model's ability to prioritize and accurately identify key features associated with implants. This focused attention indicates that the model assigns greater importance to specific patterns and structures indicative of an implant, effectively distinguishing it from surrounding anatomical features.

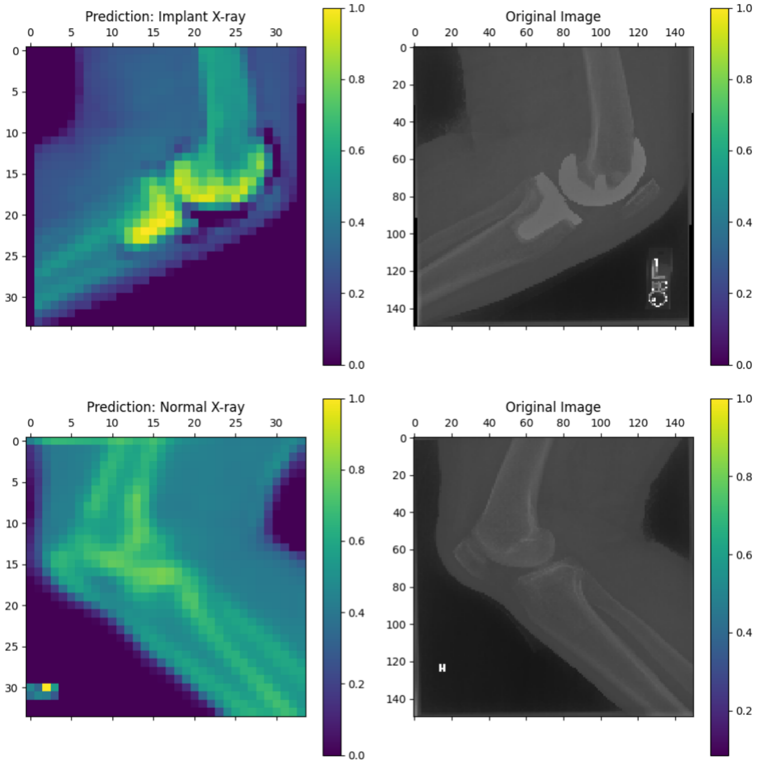

Figure 8: CNN Model Training accuracy and Loss across epochs.

Figure 9 showcases sample predictions from the SCNN model on the hold-out test dataset, illustrating its good performance. The predictions align closely with the ground truth labels, indicating accurate classification of knee replacements in both normal and implant X-ray images. The model identifies the presence of implants with high precision, as well as normal knee X-rays, demonstrating robustness across different anatomical views and varying image qualities. This performance highlights the SCNN model's dependability and potential for clinical application in automated knee implant detection and classification.

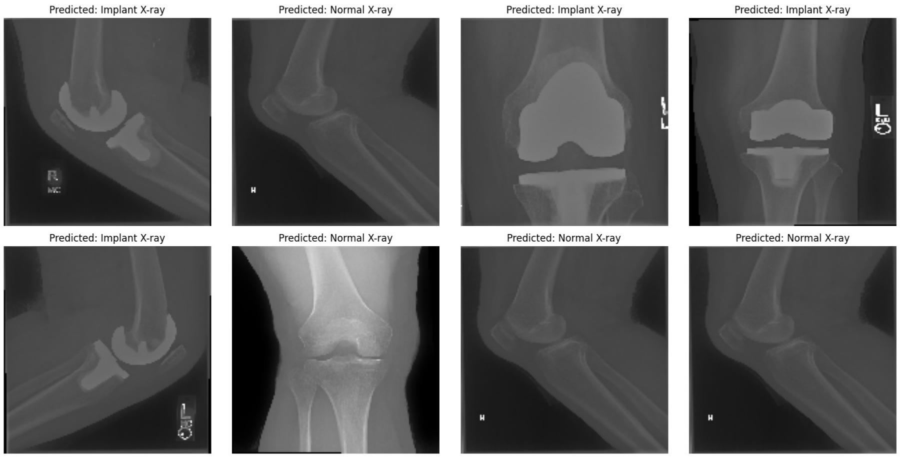

Figure 9: CNN Model Training accuracy and Loss across epochs.

Table 2 presents the accuracy of two deep learning models, CNN and SCNN, across training, validation, and testing datasets. The CNN model achieved accuracy of 91% for the hold out test data and in comparison, the SCNN model demonstrated better performance 98% in testing. These results highlight the SCNN model's better accuracy and robustness across different stages of model evaluation. The SCNN model, achieving 98% accuracy, required significantly more training time (1 hour and 56 minutes) compared to the CNN model's 28 minutes for 50 epochs. This trade-off highlights that SCNN offers higher accuracy but at the cost of increased computational time and resources.

Model	Training	Validation	Testing
CNN	100	93	91
SCNN	100	99	98

Table 2: Accuracy of CNN and SCNN models on training, validation, and testing datasets

4 Conclusion

In conclusion, our study demonstrates the effectiveness of models, particularly the SCNN which brings together scattering transform along with deep learning, for the automated detection and classification of knee implants from X-ray images. The SCNN achieved a high accuracy of 98%, surpassing the CNN's by a large margin, albeit with a longer training time. This trade-off highlights the SCNN's potential for clinical application, providing reliable and accurate implant detection that can significantly aid in pre-operative planning and decision-making. The integration of saliency mapping further enhances model interpretability, ensuring clinicians can trust and understand the automated decisions, thus paving the way for broader adoption in medical diagnostics.

References

1. Wilson NA, Jehn M, York S, Davis CM. Revision total hip and knee arthroplasty implant identification: implications for use of unique device identification 2012 AAHKS member survey results. J Arthroplasty 2014;29(2):251–5. https://doi.org/ 10.1016/j.arth.2013.06.027

2. Vina ER, Kwoh CK. Epidemiology of osteoarthritis: literature update. Curr Opin Rheumatol 2018;30(2):160–7. https://doi.org/10.1097/BOR.0000000000000479.

3. Ferket BS, Feldman Z, Zhou J, Oei EH, Bierma-Zeinstra SMA, Mazumdar M. Impact of total knee replacement practice: cost effectiveness analysis of data from the Osteoarthritis Initiative. BMJ 2017;356. https://doi.org/10.1136/bmj.j1131.

4. Kane RL, Saleh KJ, Wilt TJ, Bershadsky B. The functional outcomes of total knee arthroplasty. J. Bone Jt. Surg. - Ser. A 2005;87(8):1719–24. https://doi.org/ 10.2106/JBJS.D.02714.

5. Sakib S, Ahmed, Jawad A, Kabir J, Ahmed H. An overview of convolutional neural network: its architecture and applications. ResearchGate; 2018 [Online].

6. S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11, no. 7, pp. 674-693, July 1989, doi: 10.1109/34.192463.

7. Zintgraf, Luisa M., et al. "Visualizing deep neural network decisions: Prediction difference analysis." arXiv preprint arXiv:1702.04595 (2017).

8. B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, A. Torralba, A. Iccv, et al. Learning deep features for discriminative localization (2015), pp. 2921-2929.