Detection and Localization of calcium deposits in coronary arteries from CT scans using Deep Learning models

Sahana Srihari
ssrihar2@jh.edu

Sanika Phatak
sphatak3@jhu.edu

Abstract

As cardiovascular disease is one of the frequent cause of death around the world, it is imperative to accurately detect the presence of calcium deposits in the artery to prevent the occurrence of heart attacks. The proposed method being explored is the detection and localization of calcium deposits in the coronary arteries present in chest CT scans. The major problem in detecting the calcium deposits in the CT scans is that they smaller as compared to other highlighted pixels which increases the difficulty in detection. We hypothesise that deep networks can learn the calcium deposits as features as we go deeper into the networks. We have used Inception and Xception models to prove this and analyzed the Class activation maps across the layers of these models to show the learning. Inception model trained from scratch attains the best performance with a high accuracy score of 97% during testing.

1. Introduction

A cardiac CT scan for coronary calcium is a non-invasive way of obtaining information about the presence, location and extent of calcified plaque in the coronary arteries—the vessels that supply oxygen-containing blood to the heart muscle. Calcified plaque results when there is a build-up of fat and other substances under the inner layer of the artery. This material can calcify which signals the presence of atherosclerosis, a disease of the vessel wall, also called coronary artery disease (CAD). People with this disease have an increased risk for heart attacks. In addition, over time, progression of plaque buildup (CAD) can narrow the arteries or even close off blood flow to the heart. The result may be chest pain, sometimes called ”angina,” or a heart attack. [3]

Calcium is a marker of CAD, and the amount of calcium detected on a cardiac CT scan is a helpful prognostic tool. The findings on cardiac CT are expressed as a calcium score. CorE 64 and Core320 trials investigate the use of CT as a diagnostic tool for detecting cardiovascular diseases and disorders, as compared to cardiac catheterization. Data collected from these studies indicated CT scans as a diagnostic alternative to cardiac catheterization and showed the potential of CT scans for changing the delivery of healthcare.

One major issue that we face is the appropriate detection of the pixels corresponding to the calcium deposits relative to that of the remaining portions present in the CT scans. The pixel intensities corresponding to the calcium deposit in each slice of the CT scan results in just a handful of pixel values. Secondly, the volume of data for analysis is extremely large - the dataset acquired are volumetric datasets and grayscale, typically matrices of size 512x512x64 and the calcium deposits are generally of the order of a few voxels (10-200 voxels per volume) when present.

In this pipeline, we are using a subset of 3000 images from the original experiment. This dataset is split into training, validation and testing after appropriate augmentations. Given the nature of the problem at hand, we identify it to be a task for Deep Learning since we would need to sift through many image slices of large volumes for identification of small calcium deposits. The field of Deep Learning has rapidly progressed over the last decade with immense strides of improvement to tackle all adversity faced for computer vision and classification tasks. One of the main contributions of deep learning models has been witness during the ILSVRC [2] challenge. One such model that arose a winner is the Inception model [5] by Google and it's modification - Inception V3[6] and Xception. The Xception model Xception[1] - Extreme version of Inception, have depth-wise separable convolution. Inception V3 contained all the modification seen in the Inception v2 model along with using RMSProp Optimizer, factorized 7x7 convolutions, BatchNorm in the Auxillary Classifiers and Label smoothing. We test out dataset against these models with the pre-trained weights from the ImageNet data and also fine-tune and run the models to better fit our data.

1.1. Dataset Details

The input dataset are chest C.T. scans for which the images are collected have high dimension (512X512) in grayscale. The training for data with such high resolution takes

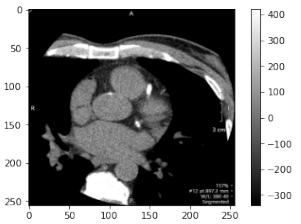

Figure 1. Original positive CT scan

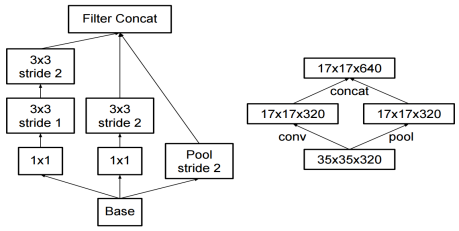

Figure 2. Inception V3 block

increases training time, therefore the images used are downsampled to (256X256). The images are CT scan slices of the chest from every patient and the slices with calcium deposits are labelled as 1(positive) and the ones without calcium deposits are labelled as 0(negative). The data initially available is largely unbalanced with only 523 positive samples among 3000 images. Data augmentation was carried out to balance the dataset. Each image was flipped vertically and horizontally along with random change in brightness and angle of rotation. Every image from the positive samples had 4 replicas and total images used for the experiment were 5086 with balanced classes. The data was split into training set, validation set and testing set with 3560, 915 and 611 samples respectively. Figure - 1 shows an original CT scan with positive calcium deposit seen as a white highlight in the center of the scan. We can see that these calcium deposits are very small compared to the whole scan and often bones which are more highlighted in the scan can be misinterpreted as calcium deposits. Hence this is the job for deep learning to learn calcium deposits as smaller more intricate features.

2. Models

2.1. Inception V3:

The main motivation behind Inception models was identifying the right kernel size for convolution operations especially for deeper networks as the passing the gradients through the network gets increasingly more difficult. Hence the solution followed concatenating multiple sized filters that makes the network wider and allows the best features to be learnt by the network. Following this network modification were brought forward to increase the accuracy and reduce the computational complexity of the model as Inception V2 and Inception V3. Inception V3 addressed the issue of representational bottleneck using smart factorization-which was factorizing the 5x5 filters into a stacked 3x3 filter representation.Also the model factorizes convolutions of filter size nxn to a combination of 1xn and nx1 convolutions. Inception V3 focused on the performance of the auxillary classifiers and their contributions to the network. It was noted that they acted more as regularizers with the presence of BatchNorm and Dropout. Therefore, with "The aim of factorizing Convolutions is to reduce the number of connections/parameters without decreasing the network efficiency" along with incorporating RMSProp Optimizer,Factorized 7x7 convolutions, BatchNorm in the Auxillary Classifiers and Label Smoothing Inception V3 was created.

2.2. Xception:

The revision of Inception V3 modle gave rise to the creation of Xception model which joined the Inception family of models. The main contribution in the introduction of depth wise separable convolutions which can be translated as an inception module with maximally large number of towers where in-between regular convolutions there is a depth-wise convolution(DSC) which are a spatial convolutions performed independently across the channels. Then this is followed by a point-wise convolution which are 1x1 convolutions. The point-wise convolution allows for projects the channel output after depth-wise convolution into a different channel space.

The architecture is technically a linear stack DSC with residual connections which contains 36 convolutional layers for the feature extraction base of the model which can be structured into 14 modules having residual connections. The modified depth-wise separable convolution have no intermediate ReLU non-linearity. The mapping of cross-correlation and spatial correlations in the feature maps are decoupled. Xception has the same number of parameters with performance gains attributed to the more efficient use of model parameters.

3. Experiment details

The main aim of the experiment was to understand how deep networks learn the features and if they are able to learn the intricate calcium deposit features. Class Activation Mapping (CAM) [7] technique is used to understand the discriminative image regions used by a CNN to identify a specific class in the image. CAMs are obtained at intermediate and end CNN layers of the deep models to understand at which stage what features of the scans are learnt. CAMs are generated by extracting the features at a particular CNN

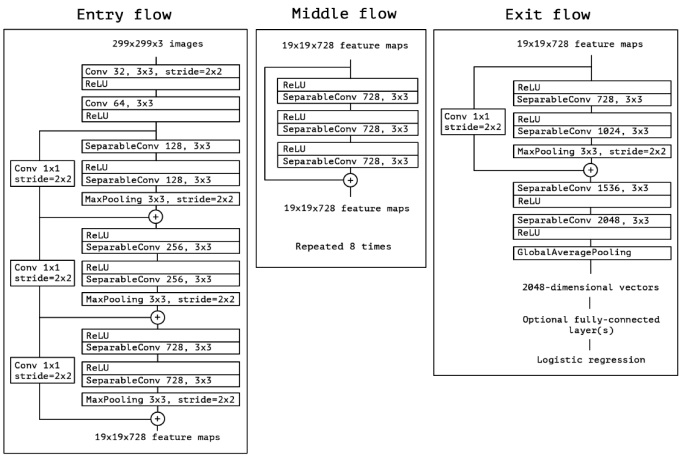

Figure 3. Xception Architecture

layer and passing them through a global average pooling layer followed by a fully connected layer with softmax activation which gives excellent localization of the features learnt. The fully connected layer in CAM, essentially uses the final weighted sum of feature map weights to give the heatmap of a particular class. For Inception model, the experiment is done for pretrained Imagenet weights with finetuned last year and the whole model trained from scratch. For Xception model, the experiment is carried out for pretrained image net weights with finetuned last layer and finetuned last 4 layers which includes training 2 CNN layers. Trained models are trained for 15 epochs for all the models. CAMs are generated for intermediate and end layers for trained models. Classification report and training accuracy and loss plots are compared for all models along with the confusion matrix.

4. Results

4.1. Inception model:

For the Inception model with pretrained Imagenet weights finetuned for only the last layer, the highest accuracy obtained is 66% with the precision score of 0.64 for the positive scans. Figure - 4 shows the training and validation loss and accuracy curves for the finetuned Inception model. The highest training accuracy obtained is 95% and highest validation accuracy obtained is 70% which starts decreasing after 10th epoch. Figure - 5 shows the truth table for the finetuned Inception model.

For the Inception model trained from scratch, the highest accuracy obtained was 97% with precision score of 0.96 for the calcium positive scans which shows significant improvement from the finetuned results which is due to the convolutional layers learning specifically for the training dataset used here and hence we can see the features learnt

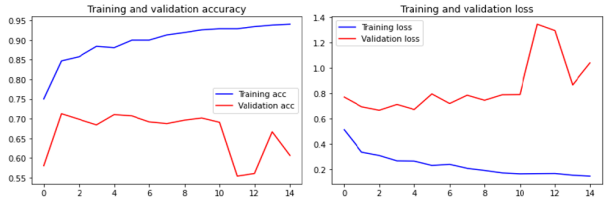

Figure 4. Training/Validation Accuracy and Loss - Inception finetuned

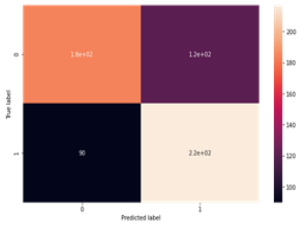

Figure 5. Truth Table - Inception finetuned

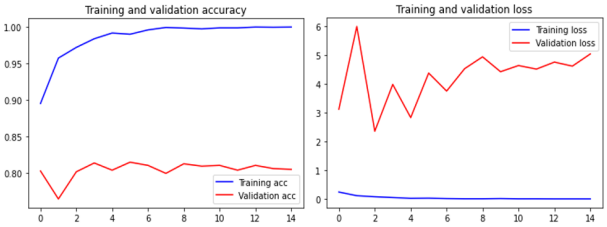

Figure 6. Training/Validation Accuracy and Loss - Inception trained

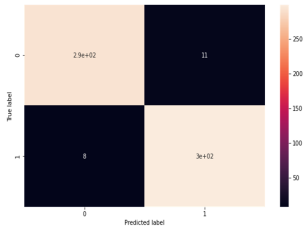

Figure 7. Truth Table - Inception trained

at intermediate layers in this model. Figure - 6 shows the training and validation loss and accuracy curves for the inception model trained from scratch. The highest training accuracy obtained is 100% and highest validation accuracy obtained is 82% which is obtained at 3rd epoch and remains almost constant post that. Figure - 7 shows the truth table for the Inception model trained from scratch.

4.2. Xception model:

For the Xception model with pretrained Imagenet weights finetuned for only the last layer, the highest accu-

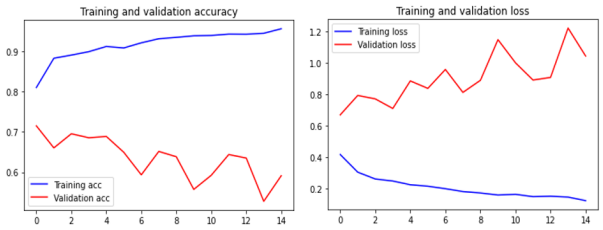

Figure 8. Training/Validation Accuracy and Loss - Xception fine-tuned

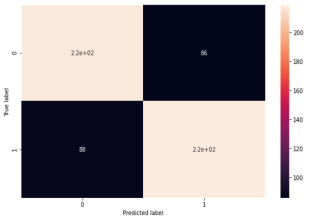

Figure 9. Truth Table - Xception finetuned

racy obtained is 72% with the precision score of 0.72 for the positive scans. Figure - 8 shows the training and validation loss and accuracy curves for the finetuned Xception model. The highest training accuracy obtained is 97% and highest validation accuracy obtained is 70% which starts decreasing after 4th epoch. Figure - 9 shows the truth table for the finetuned Xception model. The finetuned Xception model performs better than the finetuned inception model.

As mentioned before, the second Xception model was not trained from scratch and only the last few layers were retrained for our training dataset which included 2 convolutional layers. The second Xception model trained had a poor performance with just 64% accuracy with a precision score of 0.69 for the positive class. Figure - 10 shows the training and validation loss and accuracy curves for the second Xception model with last four layers trained. The highest training accuracy obtained is 95% and the highest validation accuracy obtained is around 70% which is pretty stable after the 3rd epoch itself. The second Xception performs worse than all other models and does not seem to learn much by training just the last four layers of the model. Figure - 11 shows the truth table for the second Xception model.

5. Discussion

The results clearly show that the Inception model trained from scratch gives the best results as seen in the table shown in Figure - 12. Due to unavailability of better compute power, the Xception model was trained only for the last few layers of the network which performs poorly. This is because deep networks like Xception learn the intricate fea-

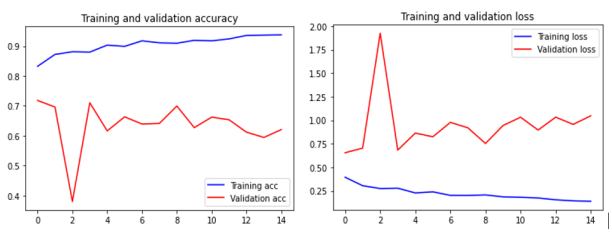

Figure 10. Training/Validation Accuracy and Loss - Xception last 4 layers trained

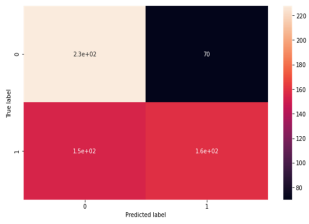

Figure 11. Truth Table - Xception last 4 layers trained

Model	Incpetion	Xception
Finetuned- last layer accuracy	62%	72%
Trained accuracy	97%	63%
	Inception	Xception
Finetuned- last layer precision	0.64	0.72
Trained precision	0.96	0.69

Figure 12. Accuracy/Precision table

tures somewhere in the intermediate layers closer to the end of the network rather than in the last two convolutional layers. We have trained only the last two convolutional layers in the second Xception model and hence it is not able to learn the required features and behaves poorly.

To understand the learning of features better, we generated the CAMs for trained Inception and Xception models and compared it to a fully trained VGG16 network [4]. Figure - 13 shows the activation maps for the trained VGG network which does not localise the calcium deposits in any of the layers well and performs poorly with a low accuracy of 67%. This proves the requirement of using deeper networks to learn the intricate calcium deposit features.

Figure - 14 shows the activation maps for the trained Inception model and we can see that as you go deeper into the network, the layers learn the calcium deposit features more prominently. In layer 86 the important portion of the image is learnt and by layer 94 of the network, the calcium deposit features are well localised hence giving a very high accuracy.

Figure - 15 shows the activation maps for the two convolutional layers of the Xception model one of which was trained on our dataset. The localisation of calcium deposits not good for the layer 128 which was trained for our dataset

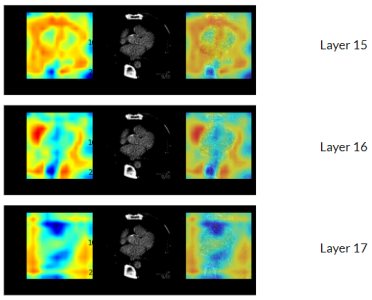

Figure 13. VGG trained model heat map

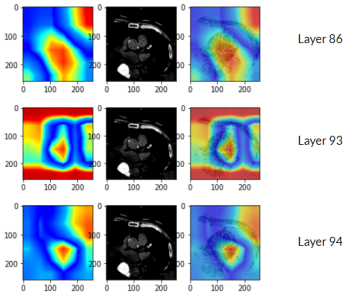

Figure 14. Inception trained model heat map

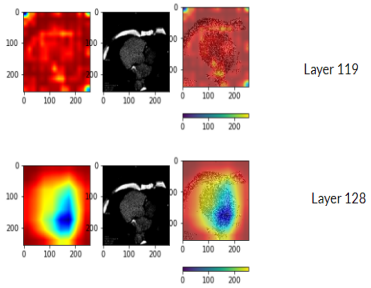

Figure 15. Xception trained model heat map

but is much better for the layer 119 with pretrained weights. This indicates that the Xception model must be training for the intricate features like calcium deposits in the layers a little higher up as compared to what we trained due to which our trained Xception model trains badly.

6. Future Work

Further, we plan to train the intermediate layers of the Xception model which might give a better performance than the currently trained model as we believe that the intermediate layer near the end of the network are more important to learn intricate features like presence of calcium deposits. We also plan to experiment with deeper networks like YOLO and Resnet to confirm weather deeper networks learn to distinguish the calcium deposit features better.

.

References

[1] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017. 1

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 1

[3] Ran Shadmi, Victoria Mazo, Orna Bregman-Amitai, and Eldad Elnekave. Fully-convolutional deep-learning based system for coronary calcium score prediction from non-contrast chest ct. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 24–28. IEEE, 2018. 1

[4] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition . In *ICLR*, 2015. 4

[5] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 1

[6] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 1

[7] B. Zhou, A. Khosla, A. Lapedriza, A. Olivia, and A. Torralba. Learning deep features for discriminative localisation. In *CVPR16*, 2016. 2