International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 06 Issue: 12 | December - 2022 Impact Factor: 7.185 ISSN: 2582-3930

APPLICATION OF CARDIOPATHIE PREDICTION

B.HITESH REDDY, B.PRAMOD REDDY, K.VAMSHIDHAR REDDY
Dr. T. VIJAYA SARADHI (Guide)
Sreenidhi Institute of Science and Technology
Hyderabad

ABSTRACT:

Using a coronary artery calcium score is one of the most used ways to diagnose coronary artery disease. However, because the radiologist must personally review each CT picture and the precise range, the present diagnosis procedure employing the coronary artery calcium score takes a lot of time. Three CNN models are used in this work to describe the 1200 normal cardiovascular system. We carry out the experimental test by dividing the CT image data into three categories: the original coronary artery calcium score CT images, which include the entire rib cage; the cardiac segmented images, which exclude all but the heart region; and the cardiac cropped images, which are produced using the cardiac images that have been divided into smaller parts and enlarged. As a result of the experimental test to identify calcium utilizing Inception Resnet V2, VGG, and Resnet 50 models in a certain CT picture.

INTRODUCTION

The most susceptible issues brought on by aging or hereditary disorders are heart ailments. It became essential to recognize it in advance to save countless lives. Increased or decreased calcium levels in the coronary arteries might result in heart disease. We can calculate the amount of calcium present in the arteries using coronary artery imaging. Heart attacks, congestive heart failure, blood clots, and a reduction in the heart's ability to receive oxygen are all caused by calcium levels that are higher than the typical amount. In this experiment, we are estimating the quantity of calcium that has been deposited on arteries using CT images. With the aid of this initiative, we can assist physicians in providing effective patient care.

To forecast a person's likelihood of developing heart-related issues, we built a machine learning model for this project using their CT scan as the input. First, the model in this project has to separate the heart from the image that has been given to it. Additionally, it determines the calcium score using the information given to it. The final result indicates whether or not a person may have heart-related problems. We have utilized the CNN, or Convolutional Neural Network, a technique to segment from the input picture in order to locate the heart location.

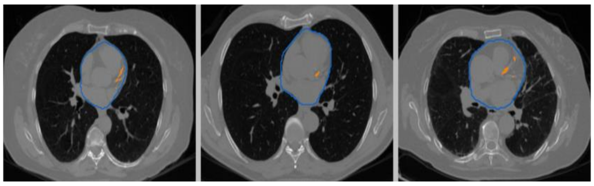

Fig 1: Ct scan of a patient.

WHAT IS CNN?

Convolutional Neural Networks, often known as CNNs, is a type of neural network designed for deep learning algorithms that are particularly useful for tasks involving pixel data and image recognition.

CNN is the most effective neural network for detecting and recognizing the item from the input. We employed it in our project due to its correctness.

Inside Of CNN:

The input must pass through each of CNN's three layers in order to produce its output. The CNN's three tiers are

- Convolutional layer
- Pooling Layer
- Fully Connected Layer

Convolutional Layer: In this layer, most computing takes place. A kernel or filter inside this layer moves over the image's receptive fields during the convolution process to determine if a feature is present.The kernel traverses the entire picture over several numbers of iterations. A dot product between the input pixels and the filter is calculated at the end of each cycle. A feature map or convolved feature is the result of the dots being connected in a certain pattern. In this layer, the picture is ultimately transformed into numerical values that the CNN can understand and extract pertinent patterns from. In this layer, the picture is ultimately transformed into numerical values that the CNN can understand and extract pertinent patterns from

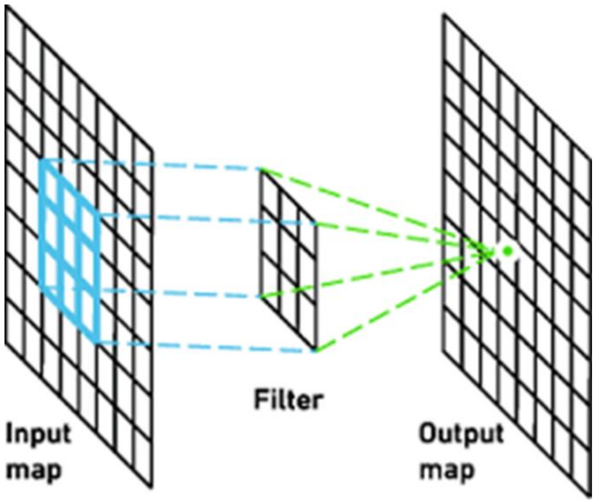

Fig2: Convolutional Layer

Pooling Layer: The pooling layer similar to the convolutional layer sweeps a kernel or filter over the input image. Contrary to the convolutional layer, the pooling layer has fewer input parameters but also causes some information to be lost. Positively, this layer simplifies the CNN and increases its effectiveness.

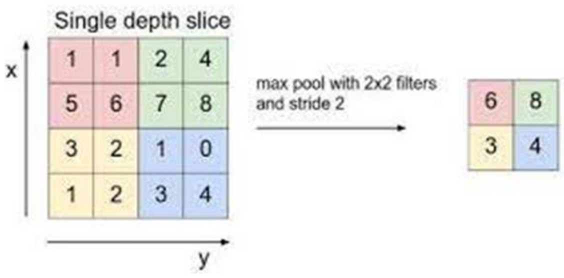

Fig 3: Pooling Layer

Fully Connected Layer: Based on the characteristics gathered in the preceding layers, picture categorization in the CNN takes place in the FC layer. Fully lined in this context indicates that every activation unit or node of the subsequent layer is connected to every input or node from the preceding layer.

WORKING OF CNN

Multiple layers of a CNN are possible, and each layer trains the CNN to recognize the many aspects of an input picture. Each picture is given a filter or kernel to create an output that grows better and more detailed with each layer. The filters may begin as basic characteristics in the bottom levels.

To examine and identify characteristics that specifically reflect the input item, the complexity of the filters increases with each additional layer. As a result, after each layer, the output of each convolved picture becomes the input for the following layer. The CNN recognizes the picture or objects it represents in the final layer, which is an FC layer.

The input image is processed via a number of different filters during convolution. Each filter performs its function by turning on specific aspects of the image, after which it sends its output to the filter in the subsequent layer. The procedures are repeated for dozens, hundreds, or even thousands of layers as each layer learns to recognize various characteristics. Finally, the CNN is able to recognize the full object thanks to the picture data flowing via its numerous layers.

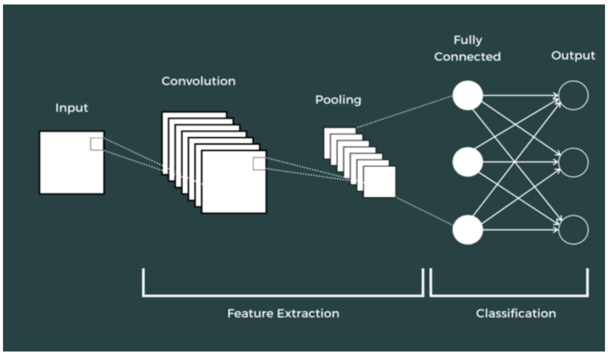

Fig4: CNN layers

ADVANTAGE OF CNN

CNN is particularly renowned for its results' great accuracy despite the volume of data. As CNN advances, it removes all the extraneous data from the input provided in order to improve the accuracy of the outputs. This is because CNN has learned from its previous phase. When compared to single-layered neural networks, many layers significantly boost the output accuracy. Any device can easily implement CNN construct models; mobile phones can even run them. CNN is employed in many different applications, including automotive and face recognition.

STEPS FOR IMPLEMENTATION

- Preparing Data Set
- Importing Libraries
- Heart localization
- Heart segmentation
- Calcium Segmentation, Calculating Calcium score

PREPARING DATASET

We are utilizing NRRD-extended data that is stored in the form of medical photographs. These conflicting pictures show both healthy and unhealthy hearts in persons with heart conditions.

A 3D image of the patient's heart, lungs, and ribcage may be found in each NRRD file.

The Simple ITK package in Python, which was created primarily to study or edit medical data, may be used to import this NRRD format file into our program.

A total of 1636 computed tomography (CT) images from the Framingham Heart Study (FHS)-CT1 were used to train and fine-tune the deep learning framework. A coronary calcium risk score was generated completely automatically in four phases. Twenty thousand eight hundred-four CT images from four separate clinical groups were independently tested.

Check for duplicate slices: In some cases, two (almost) identical CTs are stored in one folder. In this case, each CT slice is duplicated and one of them has to be removed.

Conversion to NRRD: The manual heart segmentation masks for all our cohorts were drawn using the open-source software 3D Slicer and saved in the NRRD format. Therefore, in order to be processed by the pipeline, DICOM images need to be converted to such a format.

Fig5: DataSet of this Project

IMPORT LIBRARIES

In this project, we're utilizing a variety of libraries, including Simple ITK, Tensorflow, Matplotlib, NumPy, glob, and multiprocessing. Simple ITK is used to alter medical pictures and transform them into numerical matrices.

Tensorflow: It is a Machine Learning and AI library with a wide range of algorithms. The main focus of this library is to train models and interface with deep neural networks.

Matplotlib: It is a visualization library that is used to present the results in more understandable ways i.e., in graphs and plots.

Glob: It is used to return all file paths that match a specific pattern. We can use glob to search for a specific file pattern, or perhaps more usefully, search for files where even though a certain pattern by using wild card characters.

HEART LOCALIZATION

In Heart Localization, we will use a CNN convolutional layer to find the heart in a specific scan.

As is common knowledge, images are stored in computers as data matrices. The convolutional layer uses a filter matrix to remove the unnecessary portions of images. For example, we use this filter layer to locate the heart in a particular CT scan. The pooling layer in CNN then removes any additional unwanted data or portions of images. In our instance, pictures other than the heart, such as lungs and rib cages.

Fig6: Heart Localization

HEART SEGMENTATION AND CALCIUM SEGMENTATION

To predict cardiovascular risk, we propose a deep learning system that can automatically derive a calcium score from a given CT scan. The system consists of four sequential steps: heart localization, heart segmentation, coronary calcium segmentation, and coronary calcium segmentation. For the first three phases, we trained a different fully convolutional neural network using the U-Net50 architecture. To get around the need for a very large cohort for deep learning network training, the U-Net architecture was initially created for biomedical picture segmentation.

1636 CT images were used to create the cohort for training and fine-tuning the three deep-learning models, of which 623 were from patients with coronary calcium and 1013 were from patients without it. We elected not to include several hundred more CT scans from participants who had no coronary calcium despite the fact that there was a modest imbalance between those subjects and those who did. Random choices were made for the excluded subjects. If present, coronary calcium was manually segregated by knowledgeable readers across all topics. A further sample of 129 randomly chosen training cohort individuals had their hearts mechanically segmented. Twenty thousand eight hundred-four participants from four clinical studies and trials, including cardiac-specific CT scans, lung screening CT scans, health outcomes, and follow-up data, made up our testing cohort. For 5521 participants, experienced readers' manually computed calcium ratings were accessible, as well as 895 patients' manually segmented hearts. All CT scans were padded, cropped, and resampled to the same resolution of 0.7 0.7 2.5 mm/px, with a size of 512 512 512 pixels (px). The Supplementary provides a thorough explanation of the training, tuning, and testing cohorts and how they are used.

Our system's initial network was taught to locate the heart in a particular 3D CT image. This step was important because, depending on the cohort, scanner utilized, and place where the scan was taken, CT scans might vary in size, resolution, area collected, or field of vision, for example. For training and tweaking, the training cohort was divided at 70:30, and all scans were downscaled to a size of 112112112px to fit inside the GPU memory. Standard U-Net with four downsampling steps running for 1200 epochs was used as the training model. For the purposes of heart localization and heart segmentation, data augmentation was utilized by applying a rotation of 4 degrees around the sagittal, transversal, and longitudinal axes. Furthermore, For heart localization and heart segmentation, we used translation within 10px and 20px in the axial plane, respectively. Upsampling the network's output to the size of the original CT image produced a very crude cardiac segmentation that we utilized to position a bounding box for the following stages.

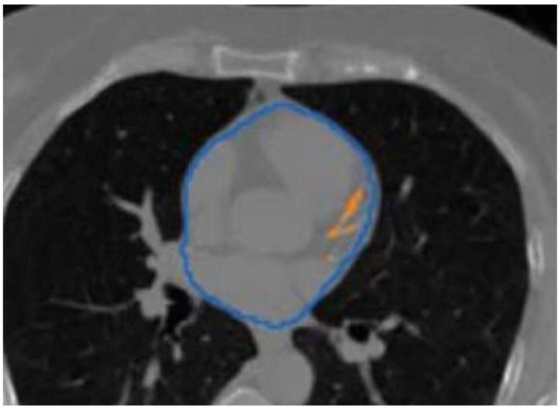

Fig7: Heart Segmentation

The deep learning system's second network was trained to divide the heart into several segments. The input scans were downsampled to 128 128 80px after being initially cropped to 384 384 80px cubes around the heart's core. The training cohort was once more divided 70/30 for training and tuning, and only modest ranges of rotations and translations were used to supplement the data. The four downsampling phases in the training model ran for 1000 epochs, using the same architecture as in the preceding step. The final model was trained to integrate the training and tuning cohorts for improved performance once the model parameters were determined to perform well on the tuning cohort. Precise segmentation of the heart was achieved by upsampling the network's output to the size of the original CT scan. Although the inaccuracy of the heart segmentation was minimal, as this step was primarily intended to limit the area for the following calcium segmentation stage, we added a rim of 11 pixels to the projected heart segmentation to guarantee the whole heart was recorded.

Coronary calcium segmentation was taught to the third network. The previously segmented heart was split up into smaller cubes for this phase, each measuring 48 x 48 x 32 pixels. Numerous trials with various cube sizes revealed that the selected size was the most effective because larger cubes lengthened training times while maintaining the same accuracy. For the rainfall, we utilized cubes that overlapped all but one pixel, however during the testing, the cubes did not overlap.

RESULTS

By calculating the calcium percentage, we will get the CAC score of patients. Now we will compare these scores with the scores predicted by expert doctors. We are using a Confusion matrix to compare the results. We can see that predicted values match 80 percent of expected values.

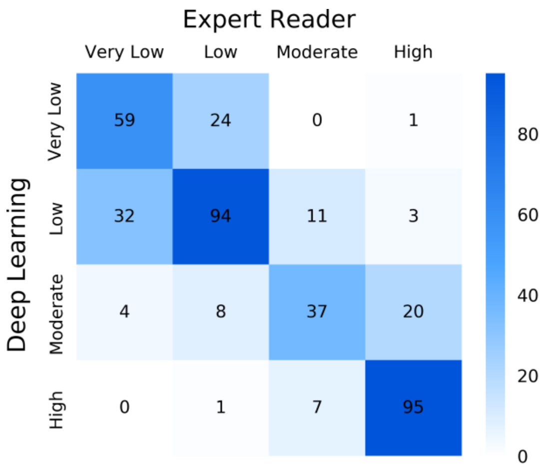

Fig9: Confusion matrix between predicted and expected scores.

Finally, after rigorous training we achieved an accuracy of 92%, which we can deploy in hospitals

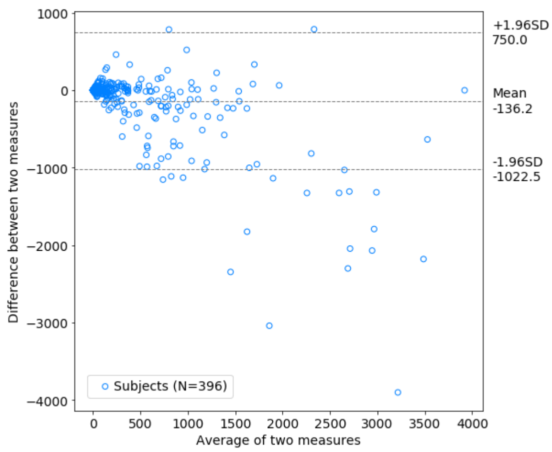

Fig10: Mean between 2 measures and difference between them

MERITS

When it comes to merits, the higher accuracy of our project makes it more reliable for decision-making. It makes the doctor's job a bit easier in predicting the stability of a patient's heart. As the accuracy is high we can install this in a CT scan machine, by this it can directly provide us with the results of reports, instead of just reports.

LIMITATIONS

The process of this project is highly reliable but it requires high-processing computers. Normal computers cannot handle the processing rate of this code. It requires a high graphic processing rate to get a higher accuracy of the data. The training of this model is comparably high. CT scan is a must as input, it cannot predict without a proper scan.

SOFTWARE AND HARDWARE REQUIREMENTS

In this project we have used Windows 10, Intel Core i7-9700 CPU of 3.00 GHz, 32.0 GB RAM, and NVIDIA GeForce RTX 2080 graphics card.

CONCLUSION

In recent times we have come across many heart-related problems and it can be a lifelong health issue for an individual. There are many reasons for a person suffering from heart-related problems. One such problem is increased or decreased calcium content in the heart. Calcium in the heart plays an important role in cardiac muscle contraction and metabolism. It is important to check calcium levels through the CT scan. If the calcium levels are low, then it may lead to heart failure. If calcium levels are high, then it may lead to irregularity in the heart pulse.

So, with this project, we have built a model which takes the CT scan reports as input and gives the result as chances of a person getting heart-related problems. It is really a good application that may make the job of doctors easier in predicting the patient's heart status. In this, the major step is to image segmentation of the heart and calcium content from the heart, for image segmentation we have used the CNN algorithm for better accuracy. Further, from the training data.

ACKNOWLEDGMENTS

We would like to express our special thanks to our mentor Dr. Vijaya Saradhi who gave us a golden opportunity to do this wonderful project on this topic which also helped us in doing a lot of research and we came to know about so many new things. We are really thankful to them. Secondly, We would also like to thank my friends who helped us a lot in finalizing this project within the limited time frame.