**Chest X-Ray Analysis using Deep Learning**

Project Synopsis submitted by

**Seema Rani - 10900218030**

**Mansi Kumari - 10900218051**

**Anindita Mitra Neogi – 10900219006**

of 4th year, Information Technology, session 2021-22

Under the guidance of

**Prof. Suparna** **Mitra**

Project Guide

&

**Prof. Sourav Mitra**

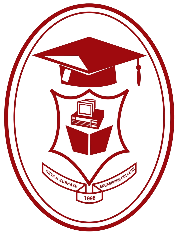
Project Co-Guide

A Project Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Technology in Information Technology under M.A.K.A.U.T.

Department of Information Technology

Netaji Subhas Engineering College

Garia, Kolkata – 700152



**Certificate**

Seema rani

Mansi Kumari

Anindita Mitra Neogi

has/have worked under my/our guidance and supervision for the project work titled **Chest X-Ray Analysis using Deep Learning** of 4th year, Information Technology, session 2021-22 and submitted the report.

--------------------------------------

**Prof. Sourav Mitra(Project Co-Guide)**

Faculty of Information Technology Department

Netaji Subhas Engineering College.

Garia, Kolkata - 700152

--------------------------------------

**Prof. Suparna Mitra(Project Guide)**

Faculty of Information Technology Department

Netaji Subhas Engineering College

Garia, Kolkata - 70015

**Acknowledgement :**

A lot of inputs from various sources have gone into the making of this report and the analysis that has been carried out. We wish to take this opportunity to thank them all.

First and foremost to **Chandan Banerjee, Head of The Department of Information Technology, Netaji Subhas Engineering College** for providing us with the opportunity to carry out our project work.

We take this opportunity to express our deep sense of gratitude and indebtedness to **Mr. Sourav Mitra(Project Co-Guide) and Mrs. Suparna Mitra(Project Guide), faculty** of Information Technology Department, Netaji Subhas Engineering College fortheir constant guidance and supervision. We would also like to thank them for providing us their valuable timeand helpful suggestions.

We are highly indebted to **Maulana Abul Kalam Azad University of Technology (formerly known as West Bengal University of Technology)** for introducing this program in our course of Bachelors of Technology.

Last but not the least, our special thanks to **all our teachers, friends, seniors and juniors** who helped us directly in completing out project successfully.

**Student’s Name: Signature: Date:**

1.Seema Rani **------------------------- -----------------**

2.Mansi Kumari **------------------------- -----------------**

3.Anindita Mitra Neogi **------------------------- -----------------**

**CONTENTS**

**Sl.No. TOPIC PAGE NO.**

1. **Abstract 5**
2. **Introduction 6-7**

2.1 X-Ray Radiography

2.2 UNet Architecture

2.3 Semantic Segmentation

1. **Related Works 8-14**

3.1 CNN

3.2 Transfer Learning

3.3 Image Level Prediction Networks

3.4 Segmentation Networks

3.5 Localization Networks

3.6 Image generation Networks

3.7 Domain Adaptation Networks

**4. Proposed Work 15-16**

**5. Objectives 17**

**6. System Design 18-27**

6.1 Flow Chart

6.2 Dataset

6.3 CNN Model

6.3.1 Convolution operation

6.3.2 Max pooling operation

6.3.3 Need for up sampling

6.3.4 Transposed Convolution

6.4 Semantic Segmentation

6.5 UNet Architecture

**7. Test Case Design 28**

**8. Implementation Details 29-31**

**9. Cost Estimation 32**

**10. Future Work 33**

**11. Conclusion 34**

**12. References 35-36**

**1.ABSTRACT:**

Recent advances in deep learning have led to a promising performance in many medical image analysis tasks. As the most commonly performed radiological exam, chest radiographs are a particularly important modality for which a variety of applications have been researched. The release of multiple, large, publicly available chest X-ray datasets in recent years has encouraged research interest and boosted the number of publications. In this project, we review all studies using deep learning on chest radiographs, categorizing works by task: image-level prediction (classification and regression), segmentation, localization, image generation and domain adaptation. And then selected a model where segmentation along with U-Net architecture of convolutional neural network is used. There are many applications of segmented images. In this paper, we tried to find the chances of diseases on the basis of counting the number of pixels.

**KEYWORDS:**

Deep Learning, chest radiograph, pixel calculation, chest X-ray analysis, Segmentation.

**2.INTRODUCTION:**

**2.1 X-ray (Radiography) – Chest:**

Chest x-ray uses a very small dose of ionizing radiation to produce pictures of the inside of the chest. It is used to evaluate the lungs, heart and chest wall and may be used to help diagnose shortness of breath, persistent cough, fever, chest pain or injury. It also may be used to help diagnose and monitor treatment for a variety of lung conditions such as pneumonia, emphysema and cancer. Because chest x-ray is fast and easy, it is particularly useful in emergency diagnosis and treatment.

In clinical practice, the CXR images are typically interpreted by radiologists and this process is time-consuming and prone to subjective assessment errors . Hence, it is always desirable to have a reliable computer tool that can efficiently and accurately aid in the diagnosis and detection of the diseases depicted on CXR images.

As an emerging technology in machine learning, deep learning has demonstrated remarkable strength in a variety tasks of medical image analyses, such as disease classification, segmentation, and registration.

With the recent availability of the Chest X-ray 14 Dataset , many deep learning approaches have been proposed to automatically diagnose the thoracic diseases in CXR images. Most of the approaches used the entire CXR images for training purpose, but this strategy may suffer from two drawbacks. First, the potential artifacts in the entire images, such as image misalignment and irrelevant objects (e.g. medical devices) as shown in Figure 1, may cause unnecessary noise and thus limit the network performance. Second, training deep neural networks typically requires resizing original images to a smaller resolution to allow a higher computational efficiency. However, this resizing process significantly reduces the image resolution and unavoidably leads to the loss of image details, which may be crucial for detecting the pathologies with small lesions. Realizing these drawbacks, some recent studies have proposed to train the networks with higher resolution images.

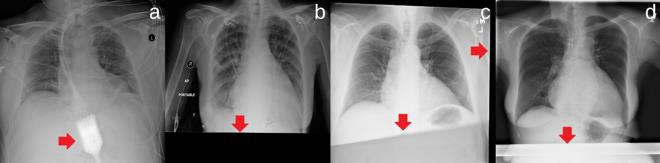


Fig. 1. Examples illustrating the entire CXR images in the Chest X-ray 14 Dataset, where (a) has an irrelevant object and (b)-(d) have misalignment as indicated by the arrows.

In this projects, we used UNet architecture of the convolutional neural network along with semantic segmentation.

**2.2. UNet:**

UNet, evolved from the traditional convolutional neural network, was first designed and applied in 2015 to process biomedical images. As a general convolutional neural network focuses its task on image classification, where input is an image and output is one label, but in biomedical cases, it requires us not only to distinguish whether there is a disease, but also to localise the area of abnormality.

UNet is dedicated to solving this problem. The reason it is able to localise and distinguish borders is by doing classification on every pixel, so the input and output share the same size.

**2.3. Semantic Segmentation:**

There are various levels of granularity in which the computers can gain an understanding of images. For each of these levels there is a problem defined in the Computer Vision domain. One of them is semantic segmentation.

The goal of semantic image segmentation is to label each pixel of an image with a corresponding class of what is being represented. Because we’re predicting for every pixel in the image, this task is commonly referred to as dense prediction.

The output itself is a high resolution image (typically of the same size as input image) in which each pixel is classified to a particular class. Thus it is a pixel level image classification.

**3.RELATED WORK:**

Deep learning is a branch of machine learning, which is a general term describing learning algorithms. The algorithm underpinning all deep learning methods is the neural network, in this case, constructed with many hidden layers (‘deep’). These

networks may be constructed in many ways with different types of layers included and the overall construction of a network is referred to as its ‘architecture’.

Following are the commonly used architectures categorized by types of application

in the CXR literature.

**3.1. Convolutional Neural Networks:**

In the 1980s, networks using convolutional layers were first introduced for image analysis (Fukushima and Miyake, 1982), and the idea was formalized over the following years (LeCun and Bengio, 1998). These convolutional layers now form the basis for all deep learning image analysis tasks, almost without exception.

Convolutional layers use neurons that connect only to a small ‘receptive field’ from the previous layer. These neurons are applied to different regions of the previous layer, operating as a sliding window over all regions, and effectively detecting the same local pattern in each location. In this way, spatial information is preserved and the learned weights are shared.

**3.2. Transfer Learning:**

Transfer learning investigates how to transfer knowledge extracted

from one domain (source domain) to another (target) domain. One of the most commonly used transfer learning approaches in CXR analysis is the use of pre-training.

With the pre-training approach, the network architecture is first trained on a large dataset for a different task, and the trained weights are then used as an initialization for the subsequent task for fine-tuning **(Yosinski et al., 2014).** Depending on data availability from the target domain, all layers can be re-trained, or only the final (fully connected) layer can be re-trained.

This approach allows neural networks to be trained for new tasks using relatively smaller datasets since useful low-level features are learned from the source domain data. It has been shown that pre-training on the ImageNet dataset (for classification of natural images) **(Baltruschat et al., 2019b)** is beneficial for chest radiography analysis and this type of transfer learning is prominently used in the research surveyed in this work. ImageNet pre-trained versions of many architectures are publicly available as part of popular deep learning frameworks.

The pretrained architectures may also be used as feature extractors, in combination with more traditional methods, such as support vector machines or random forests. Domain adaptation is another subfield of transfer learning.

**3.3.Image-level Prediction Networks**

In this work we use the term ‘image-level prediction’ to refer to tasks where prediction of a category label (classification) or continuous value (regression) is implemented by analysis of an entire CXR image. These methods are distinct from

those which make predictions regarding small patches or segmented regions of an image.

Classification and regression tasks are grouped together in this work since they typically use the same types of architecture, differing only in the final output

layer. One of the early successful deep convolutional architectures for image-level prediction was AlexNet **(Krizhevsky et al., 2012)**, which consists of 5 convolutional layers followed by 3 fully connected layers.

AlexNet became extremely influential in the literature when it beat all other competitors in the ILSVRC (ImageNet) challenge **(Deng et al., 2009)** by a large margin in 2012. Since then many deep convolutional neural network architectures have been proposed.

The VGG family of models **(Simonyan and Zisserman, 2014)** use 8 to 19 convolutional layers followed by 3 fully-connected layers. The Inception

architecture was first introduced in 2015 **(Szegedy et al., 2015)** using multiple convolutional filter sizes within layered blocks known as Inception modules.

In 2016, the ResNet family of models **(He et al., 2016)** began to gain popularity and improve upon previous benchmarks. These models define residual blocks consisting of multiple convolution operations, with skip connections which typically improve model performance. After the success of ResNet, skip connections were widely adopted in many architectures. DenseNet models **(Huang et al., 2017),** introduced in 2017, also use skip connections between blocks,

but connect all layers to each other within blocks. A later version of the Inception architecture also added skip connections (Inception-Resnet) **(Szegedy et al., 2017).**

The Xception network architecture **(Chollet, 2017)** builds upon the Inception architecture but separates the convolutions performed in the 2D image space from those performed across channels. This was demonstrated to improve performance compared to Inception V3.

**3.4.Segmentation Networks:**

Segmentation is a task where pixels are assigned a category label, and can also be considered as a pixel classification. In natural image analysis, this task is often referred to as ‘semantic segmentation’ and frequently requires every pixel in the image to have a specified category.

In the medical imaging domain these labels typically correspond to anatomical features (e.g., heart, lungs, ribs), abnormalities (e.g., tumor, opacity) or foreign objects (e.g., tubes, catheters). It is typical in the medical imaging literature to segment just one object of interest, essentially assigning the category ‘other’ to all remaining pixels.

Early approaches to segmentation using deep learning used standard convolutional architectures designed for classification tasks **(Chen et al., 2018b).** These were employed to classify each pixel in a patch using a sliding window approach. The

main drawback to this approach is that neighboring patches have huge overlap in pixels, resulting in inefficiency caused by repeating the same convolutions many times. It additionally treats each pixel separately which results in the method being

computationally expensive and only applicable to small images or patches from an image.

To address these drawbacks, fully convolutional networks (FCNs) were proposed, replacing fully connected layers with convolutional layers **(Shelhamer et al., 2017)**.This results in a network which can take larger images as input and produces a likelihood map output instead of an output for a single pixel. In 2015, a fully convolutional architecture known as the U-Net was proposed **(Ronneberger et al., 2015)** and this work has become the most cited paper in the history of medical image analysis.

The UNet consists of several convolutional layers in a contracting (downsampling) path, followed by further convolutional layers in an expanding (upsampling) path which restores the result to the input resolution. It additionally uses skip connections between the same levels on the contracting and expanding paths to recover fine details that were lost during the pooling operation. The majority of image segmentation works in this review employ a variant of the FCN or the U-Net.

**3.5.Localization Networks:**

Some model uses the term localization to refer to identification of a specific region within the image, typically indicated by a bounding box, or by a point location. As with the segmentation task, localization, in the medical domain, can be used to identify anatomical regions, abnormalities, or foreign object structures.

There are relatively few papers in the CXR literature reviewed here that deal specifically with a localization method, however, since it is an important task in medical imaging, and may be easier to achieve than a precise segmentation, we categorize these works together.

In 2014, the RCNN (Region Convolutional Neural Network) was introduced **(Girshick et al., 2014)**, identifying regions of interest in the image and using a CNN architecture to extract features of these regions. A support vector machine (SVM) was used to classify the regions based on the extracted features.This method involves several stages and is relatively slow.

It was later superseded by fast-RCNN **(Girshick, 2015)** and subsequently by faster-RCNN **(Ren et al., 2017)** which streamlined the processing pipeline, removing the need for initial region identification or SVM classification, and improving both speed and performance. In 2017, a further extension was added to faster-RCNN to additionally enable a precise segmentation of the item identified within the bounding box. This method is referred to as Mask R-CNN **(He et al., 2017).**

While this is technically a segmentation network, we mention it here as part of the RCNN family. Another architecture which has been popular in object localization is YOLO (You Only Look Once), first introduced in 2016 **(Redmon et al., 2016)** as a single-stage object detection method, and improved in subsequent versions in 2017 and 2018 **(Redmon and Farhadi, 2017, 2018)**. The original YOLO architecture, using a single CNN and an image-grid to specify outputs was significantly faster than its contemporaries but not quite as accurate.

The improved versions leveraged both classification and detection training data and introduced a number of training improvements to achieve state of the art performance while remaining faster than its competitors. A final localization network that features in medical imaging literature is **RetinaNet (Lin et al., 2017)**.

Like YOLO, this is a single stage detector, which introduces the concept of a focal loss function, forcing the network to concentrate on more difficult examples during training. Most of the localization works included in this review use one of the architectures described above.

**3.6. Image Generation Networks:**

One of the tasks deep learning has been commonly used for is the generation of new, realistic images, based on information learned from a training set. There are numerous reasons to generate images in the medical domain, including generation of more easily interpretable images (by increasing resolution, or removal of projected structures impeding analysis), generation of new images for training (data augmentation), or conversion of images to emulate appearances from a different domain (domain adaptation). Various generative schemes have also been

used to improve the performance of tasks such as abnormality detection and segmentation.

Image generation was first popularized with the introduction of the generative adversarial network (GAN) in 2014 **(Goodfellow et al., 2014).** The GAN consists of two network architectures, an image generator, and a discriminator which attempts to differentiate generated images from real ones. These two networks

are trained in an adversarial scheme, where the generator attempts to fool the discriminator by learning to generate the most realistic images possible while the discriminator reacts by progressively learning an improved differentiation between real and generated images.

The training process for GANs can be unstable with no guarantee of convergence, and numerous researchers have investigated stabilization and improvements of the basic method **(Salimans et al., 2016; Heusel et al., 2017; Karras et al., 2018; Arjovsky et al., 2017).** GANs have also been adapted to conditional data generation **(Chen et al., 2016; Odena et al., 2017**) by incorporating class labels, image-to-image translation (conditioned on an image in this case) **(Isola et al., 2017),** and unpaired image-to-image translation **(CycleGAN Zhu et al. (2017)).**

GANs have received a lot of attention in the medical imaging community and several papers were published for medical image analysis applications in recent years **(Yi et al., 2019b).** Many of the image generation works identified in this review employed GAN based architectures.

**3.7.Domain Adaptation Networks:**

In this work we use the term ‘Domain Adaptation’, which is a subfield of transfer learning, to cover methods attempting to solve the issue that architectures trained on data from a single ‘domain’ typically perform poorly when tested on data from

other domains.

The term ‘domain’ is weakly defined; In medical imaging it may suggest data from a specific hardware (scanner), set of acquisition parameters, reconstruction method or hospital. It could, less frequently, also refer to characteristics of the population included, for example the gender, ethnicity, age or even strain of some pathology included in the dataset.

Domain adaptation methods consider a network trained for an image analysis task on data from one domain (the source domain), and how to perform this analysis accurately on a different domain (the target domain). These methods can be categorized as supervised, unsupervised, and semi-supervised depending on the availability of labels from the target domain and they have been investigated for a variety of CXR applications from organ segmentation to multi-label abnormality classification.

There is no specific architecture that is typical for domain adaptation, but rather architectures are combined in various ways to achieve the goal of learning to analyze images from unseen domains. The approaches to this problem can be broadly divided into three classes (following the categorization of **(Wang and Deng, 2018))**; discrepancy-based, reconstruction-based and adversarial-based.

Discrepancy-based approaches aim to induce alignment between the source and target domain in some feature space by fine-tuning the image analysis network and optimizing a measurement of discrepancy between the two domains. Reconstruction-based approaches, on the other hand, use an auxiliary encoder-decoder reconstruction network that aims to learn domain invariant representation through a shared encoder. Adversarial-based approaches are based on the concept of adversarial training from GANs, and use a discriminator network which tries to distinguish between samples from the source and target domains, to encourage the use of domain-invariant features.

This category of approaches is the most commonly used in CXR analysis for domain adaptation, and consists of generative and non-generative models. Generative models transform source images to resemble target images by operating directly on pixel space whereas non generative models use the labels on the source domain and leverage adversarial training to obtain domain invariant representations.

The majority of works surveyed in this review use one or more of the model architectures discussed here with varying numbers of hidden layers.

**4. PROPOSED WORK:**

Among all the models mentioned above we found UNet architecture gives better result. In this architecture of convolutional neural network, we get the same size of image as output as we give input. As a general convolutional neural network focuses its task on image classification, where input is an image and output is one label, but in biomedical cases, it requires us not only to distinguish whether there is a disease, but also to localise the area of abnormality.

Deep Learning is a subset of artificial intelligence function that provides the system with the ability to learn from data without being programmed explicitly.This is basically a mathematical and probabilistic model which requires tons of computations. It is very trivial for humans to do those tasks, but computational machines can perform similar tasks very easily.

Consumer hardware may not be able to do extensive computations very quickly as a model may require to calculate and update millions of parameters in run-time for a single iterative model like deep neural networks.Thus, there is a scope for the hardware which works well with extensive calculation.

Deep learning is a very CPU intensive program-esque thing to be running, so be prepared to shell out a lot of money for a good enough system. **Here are some system requirements:**

**Hardware:**

A minimum of 8GB RAM can do the job, 16GB RAM and above is recommended for most deep learning tasks. When it comes to CPU, a minimum of 7th generation (Intel Core i7 processor) is recommended.

**Software:**

**1. TensorFlow :**

Website: https://www.tensorflow.org/

GitHub Repository: https://github.com/tensorflow/tensorflow

Developed By: Google Brain Team

Primary Purpose: Deep Neural Networks

**2. Numpy:**

Website: https://numpy.org/

Github Repository: https://github.com/numpy/numpy

Developed By: Community Project (originally authored by Travis Oliphant)

Primary purpose: General Purpose Array Processing

**3.Pandas**

Website: https://pandas.pydata.org/

Github Repository: https://github.com/pandas-dev/pandas

Developed By: Community Developed (Originally Authored by Wes McKinney)

Primary Purpose: Data Analysis and Manipulation

**4. Scikit-Learn:**

Website: https://scikit-learn.org/

Github Repository: https://github.com/scikit-learn/scikit-learn

Developed By: SkLearn.org

Primary Purpose: Predictive Data Analysis and Data Modeling

**5. Keras:**

Website: https://keras.io/

Github Repository: https://github.com/keras-team/keras

Developed By: various Developers, initially by Francois Chollet

Primary purpose: Focused on Neural Networks

**6. Matplotlib:**

Website: https://matplotlib.org/

Github Repository: https://github.com/matplotlib/matplotlib

Developed By: Micheal Droettboom, Community

Primary purpose: Data Visualization

**Conclusion:**

Every library has its own positives and negatives. These aspects should be taken into account before selecting a library for the purpose of deep learning and the model’s accuracy should also be checked after training and testing the models so as to select the best model in the best library to do our task.

**5. OBJECTIVES:**

Chest x-ray (i.e., radiography) and chest CT are a more effective imaging technique for diagnosing lunge related problems. Still, a substantial chest x-ray is a lower cost process in comparison to chest CT. Deep learning is the most successful technique of machine learning, which provides useful analysis to study a large amount of chest x-ray images. The main aim of this technique helps in reducing the workload of radiologists. But the challenging part of this project is accuracy.

In this model, we have used the models that uses a multi-class configuration with three target classes and are trained and tested on the publicly available JSRT

database, consisting of 247 X-ray images the ground-truth masks for which are available in the SCR database. Our best performing model, trained with the loss function based on the Dice coefficient, reached mean Jaccard overlap scores of 95.0% for lungs, 86.8% for clavicles and 88.2% for heart. This architecture outperformed the human observer results for lungs and heart. And then we applied UNet architecture of convolutional neural network in this dataset.

To diagnose a disease in any X-ray image, one has to upload the image in .png format. The model will predict whether there is any disease(like- Pneumonia,Lung cancer, Pulmonary tuberculosis, Inflammatory mass etc.) present in chest area or not.

**6.SYSTEM DESIGN:**

**6.1Flow Chart of Project:**

X-ray images from JSRT dataset

Semantic Segmentation using mask from publicly available SCR dataset

Pre-Processing of publicly available JSRT dataset

Predicting Output

Build CNN model using UNet architecture

Testing, Validating and Visualizing data

**6.2.** **Dataset**

This model use a multi-class configuration with three target

classes and are trained and tested on the publicly available JSRT

database, consisting of 247 X-ray images the ground-truth masks

for which are available in the SCR database.

**6.3.** **CNN Model**

**6.3.1.** **Convolution operation**

There are two inputs to a convolutional operation:

i) A 3D volume (input image) of size (nin x nin x channels)

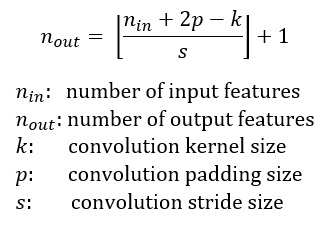
ii) A set of ‘k’ filters (also called as kernels or feature extractors) each

one of size (f x f x channels), where f is typically 3 or 5.

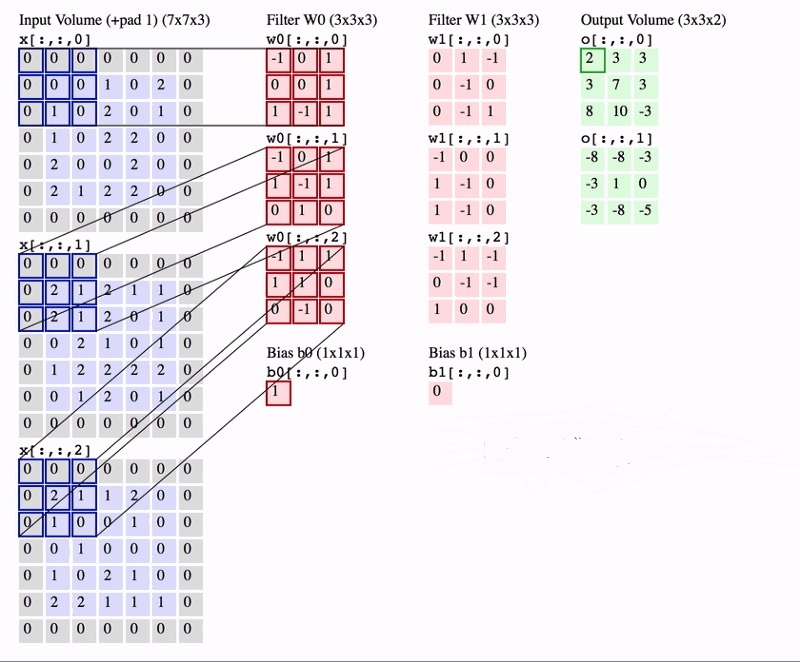
The output of a convolutional operation is also a 3D volume (also called

as output image or feature map) of size (nout x nout x k).

The relationship between nin and nout is as follows:



Convolution operation can be visualized as follows:



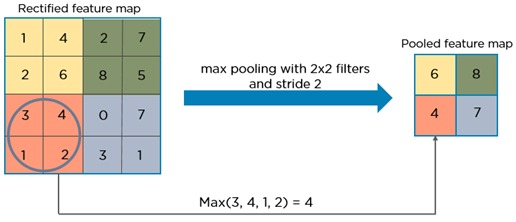
In the above Fig, we have an input volume of size 7x7x3. Two filters each of size 3x3x3. Padding =0 and Strides = 2. Hence the output volume is 3x3x2.One important term used frequently is called as the Receptive filed. This is nothing but the region in the input volume that a particular feature extractor (filter) is looking at. In the above Fig, the 3x3 blue region in the input volume that the filter covers at any given instance is the receptive field. This is also sometimes called as the context.

To put in very simple terms, receptive field (context) is the area of the input image that the filter covers at any given point of time.

**6.3.2.Max pooling operation:**

In simple words, the function of pooling is to reduce the size of the feature map so that we have fewer parameters in the network.

For example:



Basically from every 2x2 block of the input feature map, we select the maximum pixel value and thus obtain a pooled feature map. Note that the size of the filter and strides are two important hyper-parameters in the max pooling operation.

The idea is to retain only the important features (max valued pixels) from each region and throw away the information which is not important. By important, I mean that information which best describes the context of the image.

A very important point to note here is that both convolution operation and specially the pooling operation reduce the size of the image. This is called as down sampling. In the above example, the size of the image before pooling is 4x4 and after pooling is 2x2. In fact down sampling basically means converting a high resolution image to a low resolution image.

Thus before pooling, the information which was present in a 4x4 image, after pooling, (almost) the same information is now present in a 2x2 image.

Now when we apply the convolution operation again, the filters in the next layer will be able to see larger context, i.e. as we go deeper into the network, the size of the image reduces however the receptive field increases.

Notice that in a typical convolutional network, the height and width of the image gradually reduces (down sampling, because of pooling) which helps the filters in the deeper layers to focus on a larger receptive field (context). However the number of channels/depth (number of filters used) gradually increase which helps to extract more complex features from the image.

Intuitively we can make the following conclusion of the pooling operation. By down sampling, the model better understands “WHAT” is present in the image, but it loses the information of “WHERE” it is present.

**6.3.3.Need for up sampling**

As stated previously, the output of semantic segmentation is not just a class label or some bounding box parameters. In-fact the output is a complete high resolution image in which all the pixels are classified.

Thus if we use a regular convolutional network with pooling layers and dense layers, we will lose the “WHERE” information and only retain the “WHAT” information which is not what we want. In case of segmentation we need both “WHAT” as well as “WHERE” information.

Hence there is a need to up sample the image, i.e. convert a low resolution image to a high resolution image to recover the “WHERE” information.

In the literature, there are many techniques to up sample an image. Some of them are bi-linear interpolation, cubic interpolation, nearest neighbor interpolation, unpooling, transposed convolution, etc. However in most state of the art networks, transposed convolution is the preferred choice for up sampling an image.

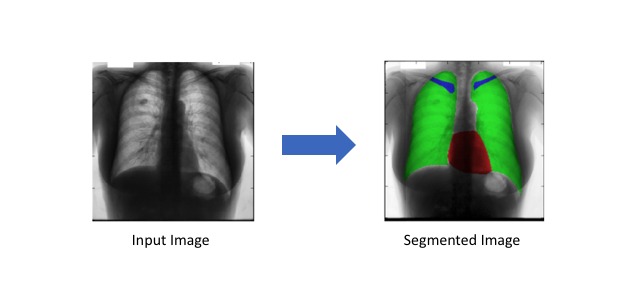
**6.3.4 Transposed Convolution**

Transposed convolution (sometimes also called as deconvolution or fractionally strided convolution) is a technique to perform up sampling of an image with learnable parameters.

However, on a high level, transposed convolution is exactly the opposite process of a normal convolution i.e., the input volume is a low resolution image and the output volume is a high resolution image. By just taking the transpose of the filter matrix, we can reverse the convolution process, hence the name transposed convolution.

**6.4. Semantic Segmentation**

There are various levels of granularity in which the computers can gain an understanding of images. For each of these levels there is a problem defined in the Computer Vision domain. One of them is semantic segmentation.



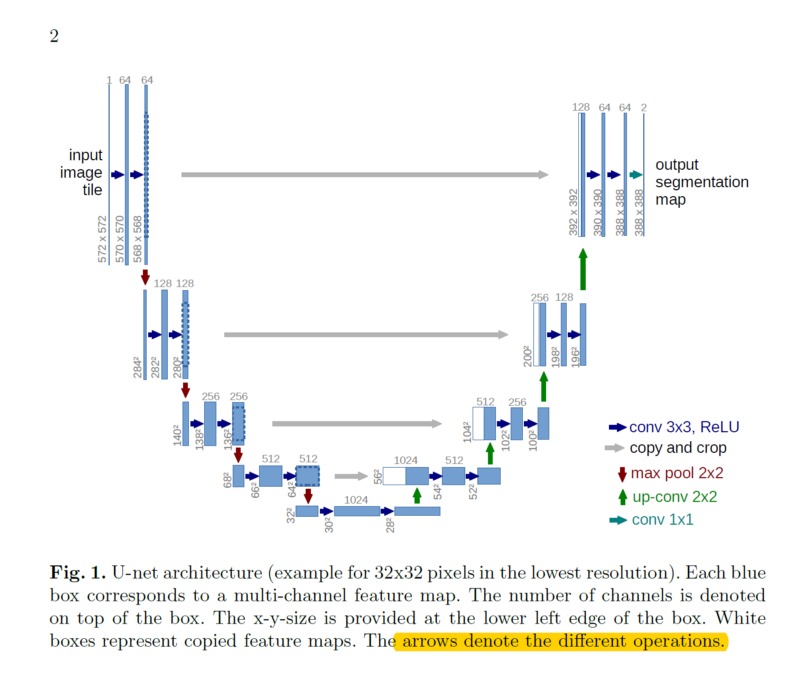
The goal of semantic image segmentation is to label each pixel of an image with a corresponding class of what is being represented. Because we’re predicting for every pixel in the image, this task is commonly referred to as dense prediction.

**6.5. UNET Architecture and Training**

The UNET was developed by Olaf Ronneberger et al. for Bio Medical Image Segmentation. The architecture contains two paths. First path is the contraction path (also called as the encoder) which is used to capture the context in the image. The encoder is just a traditional stack of convolutional and max pooling layers.

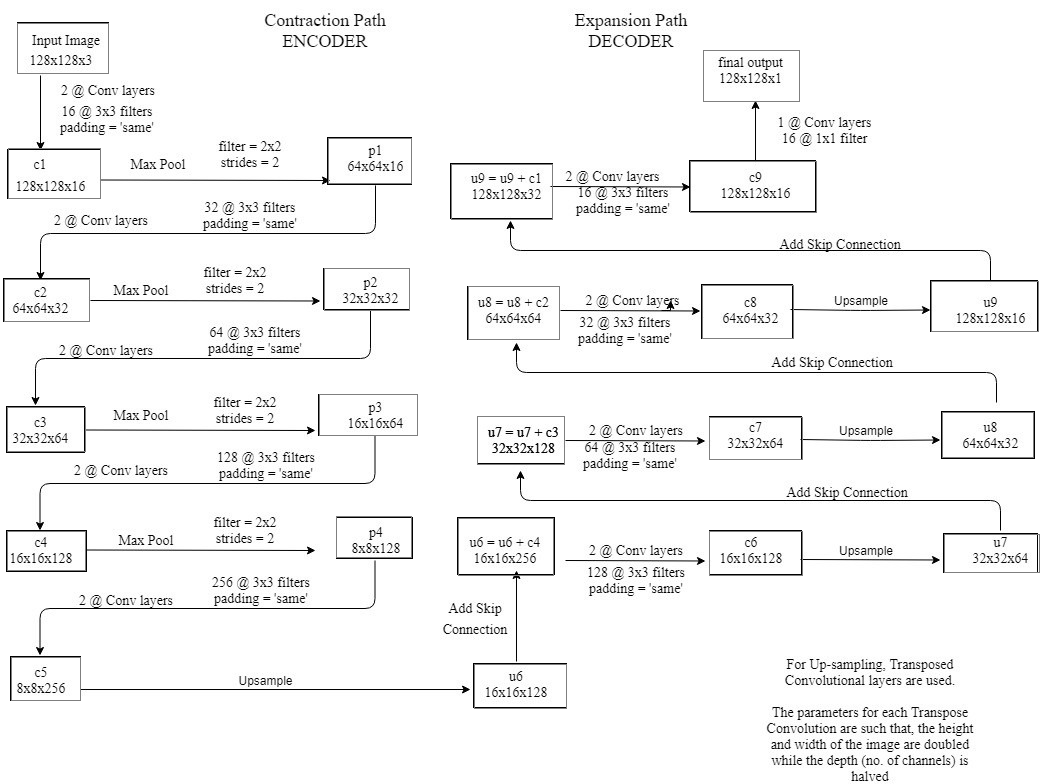
The second path is the symmetric expanding path (also called as the decoder) which is used to enable precise localization using transposed convolutions. Thus it is an end-to-end fully convolutional network (FCN), i.e. it only contains Convolutional layers and does not contain any Dense layer because of which it can accept image of any size.

In the original paper, the UNET is described as follows:



In the original paper, the size of the input image is 572x572x3, however, we will use input image of size 128x128x3. Hence the size at various locations will differ from that in the original paper but the core components remain the same.

Below is the detailed explanation of the architecture:



**Points to note:**

* 2@Conv layers means that two consecutive Convolution Layers are applied
* c1, c2, …. c9 are the output tensors of Convolutional Layers
* p1, p2, p3 and p4 are the output tensors of Max Pooling Layers
* u6, u7, u8 and u9 are the output tensors of up-sampling (transposed convolutional) layers
* The left hand side is the contraction path (Encoder) where we apply regular convolutions and max pooling layers.
* In the Encoder, the size of the image gradually reduces while the depth gradually increases. Starting from 128x128x3 to 8x8x256
* This basically means the network learns the “WHAT” information in the image, however it has lost the “WHERE” information
* The right hand side is the expansion path (Decoder) where we apply transposed convolutions along with regular convolutions
* In the decoder, the size of the image gradually increases and the depth gradually decreases. Starting from 8x8x256 to 128x128x1
* Intuitively, the Decoder recovers the “WHERE” information (precise localization) by gradually applying up-sampling
* To get better precise locations, at every step of the decoder we use skip connections by concatenating the output of the transposed convolution layers with the feature maps from the Encoder at the same level:

u6 = u6 + c4

u7 = u7 + c3

u8 = u8 + c2

u9 = u9 + c1

After every concatenation we again apply two consecutive

regular convolutions so that the model can learn to assemble a more

precise output

* This is what gives the architecture a symmetric U-shape, hence the name UNET
* On a high level, we have the following relationship:

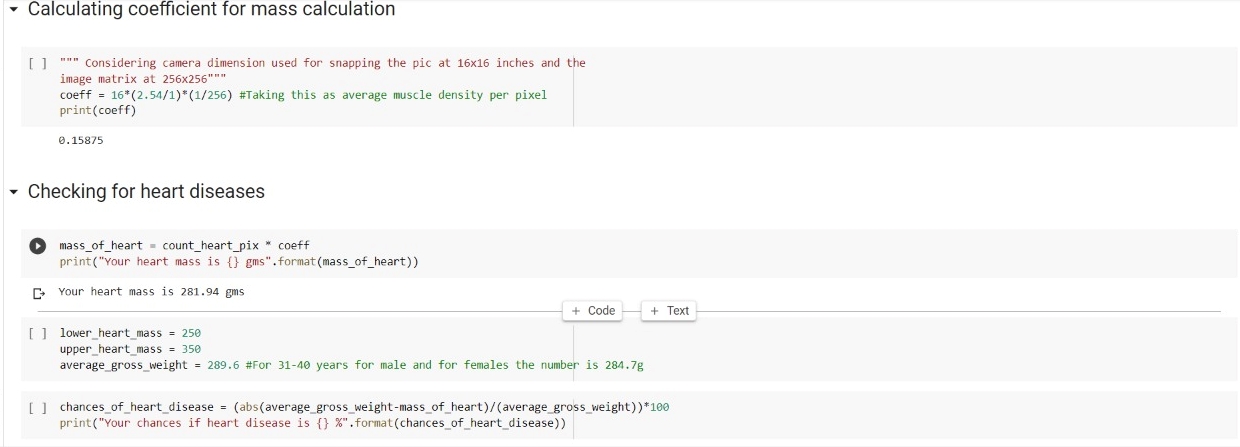
Input (128x128x1) => Encoder =>(8x8x256) => Decoder =>Ouput (128x128x1)

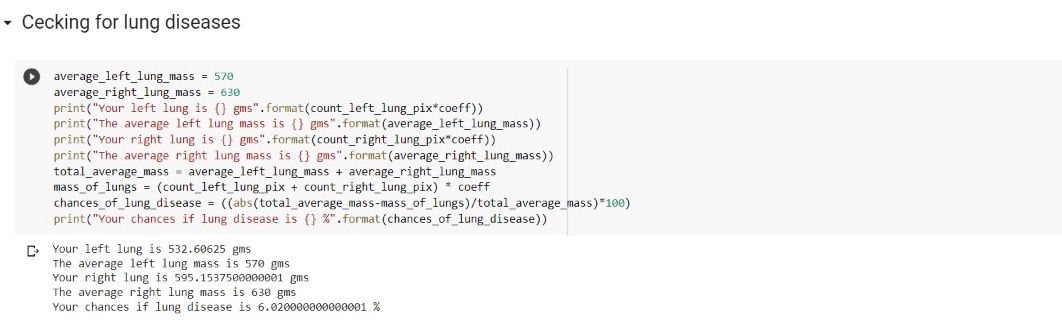
**8. Test Case Design:**

Segmented images (output of UNet Architecture) are used to calculate the number of pixels with the help of which mass is calculated and compared to the average mass to find out the chances of diseases.

From the research, it has been proven that, the higher the weight have higher risk of cardiovascular disease. Also due to some environmental, biological and life style it is possible that mass of lungs and hearts may vary from the average masses which can result in disease.

Firstly calculated pixel size on the basis of camera dimension and range of pixel. this coefficient is considered as average muscle density per pixel.

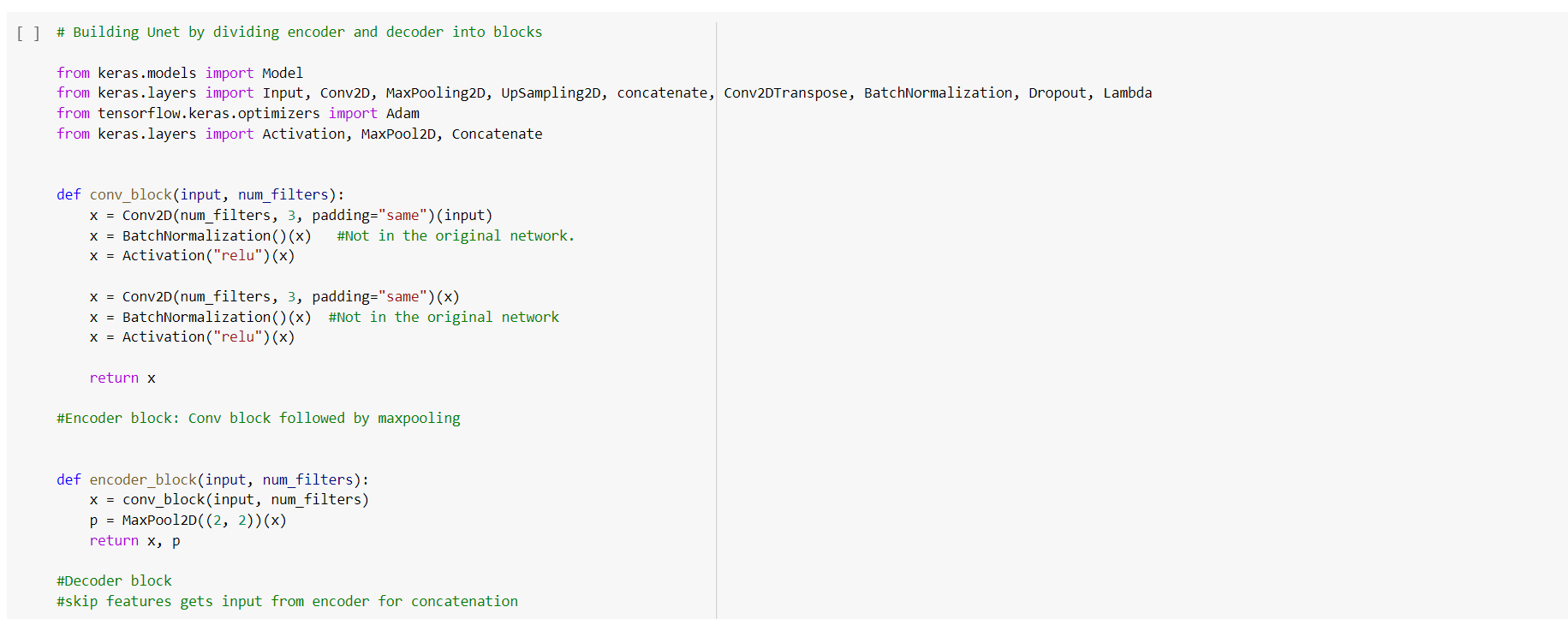


**9.Implementation Details:**

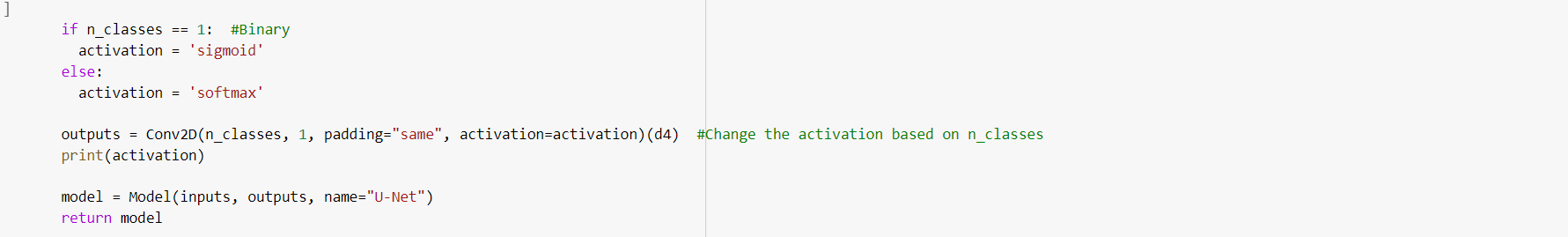
Following images shows input including chest image and their mask. The JSRT dataset has been arranged according to the masks.

****

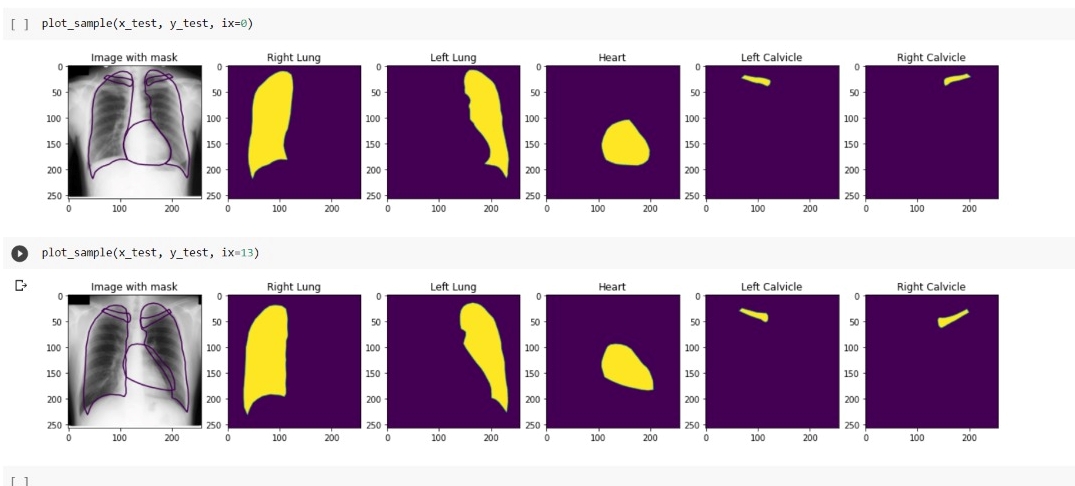
Attaching the model that we have used i.e, U-Net.



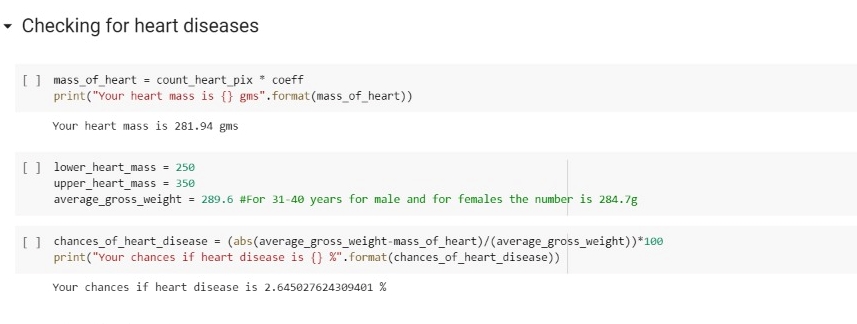


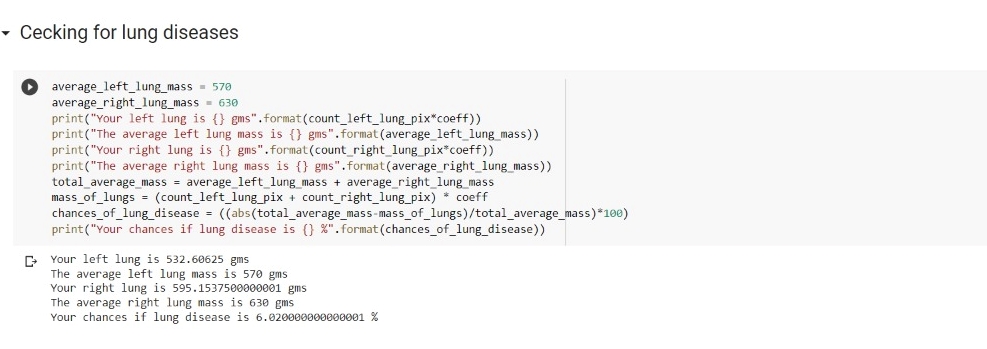


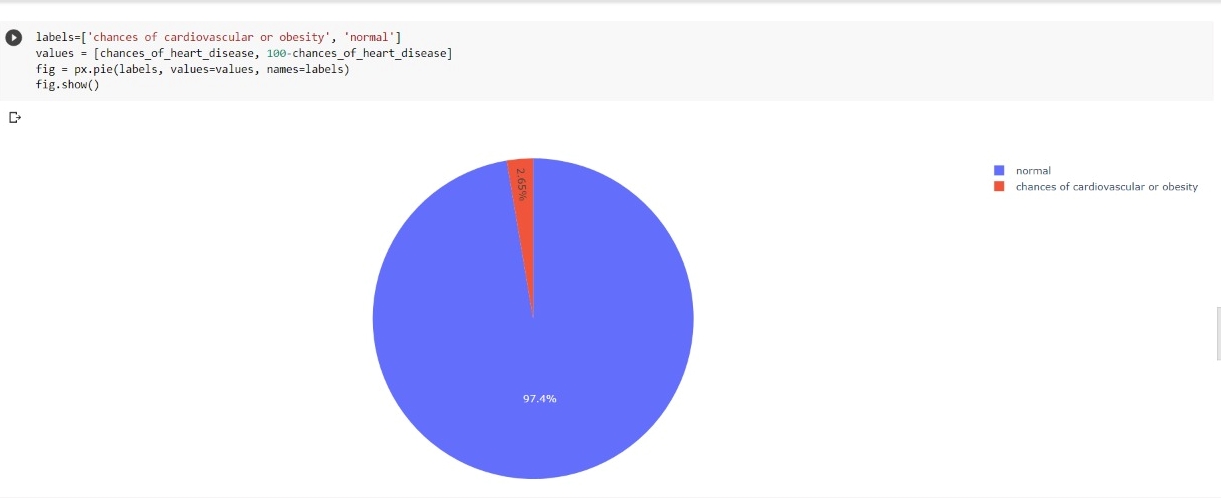
The Output after semantic segmentation:

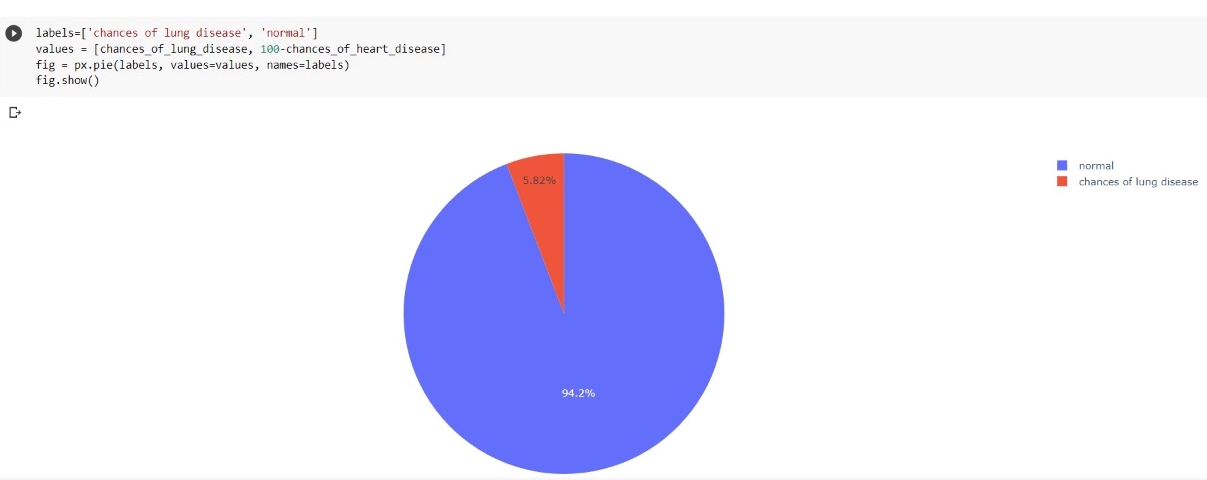


Application of segmented image for calculating the chances of disease in heart and lungs on the basis of their pixel sizes:









**10.Cost Estimation:**

Asthe dataset and architecture of the model are already exist online so this project is basically free. But the time consumed to train the model having 247 Xray images of JSRT dataset was 20-25 minutes.**11.Future Work:**

The accuracy that we have achived is approx. 92% which can be further increased by increasing the size of data or increasing the depth of model.

The output that we got from segmentation can be helpful in multiple ways in medical science like, getting clear observation of xrays, predicting the chances of diseases,etc.**12.Conclusion:**

Recent advances in deep learning have led to a promising performance in many medical image analysis tasks. In this project, we review all studies using deep learning on chest radiographs, categorizing works by task: image-level prediction (classification and regression), segmentation, localization, image generation and domain adaptation. And then selected a model where segmentation along with UNet architecture of convolutional neural network is used. There are many application like, getting clear observation of xrays, predicting the chances of diseases, etc. In this project, chances of getting heart and lung diseases are predicted by calculating the pixel size and comparing with the average mass of heart and lungs.

**7. REFERENCES:**

1. <https://towardsdatascience.com/unet-line-by-line-explanation-9b191c76baf5>
2. <https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47>
3. <https://towardsdatascience.com/types-of-convolutions-in-deep-learning-717013397f4d>
4. <https://github.com/zubaerimran/prepare_jsrt/blob/master/prepare_jsrt.ipynb>
5. <https://github.com/sohiniroych/U-net-for-Multi-class-semantic-segmentation>
6. <https://blog.paperspace.com/unet-architecture-image-segmentation/>
7. <https://arxiv.org/abs/1701.08816>
8. Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are

features in deep neural networks?, in: Advances in Neural Information Processing

Systems, Curran Associates, Inc.. pp. 1–9.

1. Baltruschat, I., Steinmeister, L., Nickisch, H., Saalbach, A., Grass, M., Adam, G., Knopp, T., Ittrich, H., 2020. Smart chest X-ray worklist prioritization using artificial intelligence: a clinical workflow simulation. Eur. Radiol.doi:10.1007/s00330-020-07480-7.
2. Baltruschat, I.M., Nickisch, H., Grass, M., Knopp, T., Saalbach, A.,

2019a. Comparison of Deep Learning Approaches for Multi-Label

Chest X-Ray Classification. Scientific Reports 9, 6381. doi:10.1038/

s41598- 019-42294-8.

1. Baltruschat, I.M., Steinmeister, L., Ittrich, H., Adam, G., Nickisch, H., Saalbach, A., von Berg, J., Grass, M., Knopp, T., 2019b. When Does Bone Suppression And Lung Field Segmentation Improve Chest X-Ray Disease

Classification?, in: 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019),

IEEE. pp.1362–1366.doi:10.1109/ISBI.2019.8759510.

1. Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. ImageNet Classification with Deep Convolutional Neural Networks.
2. Girshick, R., 2015. Fast r-CNN, in: 2015 IEEE International Conference on Computer Vision (ICCV),

IEEE. pp. 1440–1448. doi:10.1109/iccv.2015.169.

1. Girshick, R., Donahue, J., Darrell, T., Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE. pp. 580–587. doi:10.1109/cvpr.2014.81.
2. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L., 2018b.

DeepLab: Semantic image segmentation with deep convolutional nets,

atrous convolution, and fully connected CRFs. IEEE Transactions on Pattern

Analysis and Machine Intelligence 40, 834–848. doi:10.1109/tpami.

2017.2699184.

1. Yu et al. (2020) A model based on U-Net and Faster R-CNN to detect PICC catether and its tip
2. Zhang et al. (2019b) Tailored Mask R-CNN for simultaneous detection and segmentation
3. Wessel et al. (2019) Uses Mask R-CNN iteratively to segment and detect ribs. LC R PR
4. Sogancioglu et al. (2020) Comparison of image-level prediction and segmentation models for cardiomegaly
5. Que et al. (2018) A network with DenseNet and U-Net for classification of cardiomegaly
6. Chen et al. (2018a) CycleGAN based DA model with semantic aware loss for lung segmentation
7. Wang et al. (2020d) U-Net based architecture based on dense connections
8. Xue et al. (2020) Cascaded U-net with sample selection with imperfect segmentations
9. <https://www.ahajournals.org/doi/10.1161/CIR.0000000000000973>
10. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3250069/>
11. <http://www.people.vcu.edu/~mhcrosthwait/clrs322/pixelconnect.html>
12. <https://www.bing.com/search?q=volume+of+left+lung&cvid=124d5f1cbe414e7b803c3a66e7630c80&aqs=edge..69i57.4481j0j1&FORM=ANNTA1&PC=LCTS>
13. <https://www.google.com/search?q=mass+of+heart+in+cm&ei=MnOQYvuOPILCz7sP3YC_kAQ&ved=0ahUKEwj777KOiv_3AhUC4XMBHV3AD0IQ4dUDCA4&uact=5&oq=mass+of+heart+in+cm&gs_lcp=Cgdnd3Mtd2l6EAMyBAgAEEcyBAgAEEcyBAgAEEcyBAgAEEcyBAgAEEcyBAgAEEcyBAgAEEcyBAgAEEc6BwgAEEcQsANKBAhBGABKBQhAEgExSgQIRhgAUDRYNGCiBWgBcAJ4AIABAIgBAJIBAJgBAKABAcgBCMABAQ&sclient=gws-wiz>