

**Name: Anik Manik**

**Email address: iamanik4@gmail.com**

**Contact number: 9477672426**

**Anydesk address: 400 728 410**

**Years of Work Experience: 2.6 years**

**Date: 24th Jan 2021**

## **Self-Case Study 2: SIIM-ACR Pneumothorax Segmentation**

**Identify Pneumothorax disease in chest x-rays**

### **Introduction:**

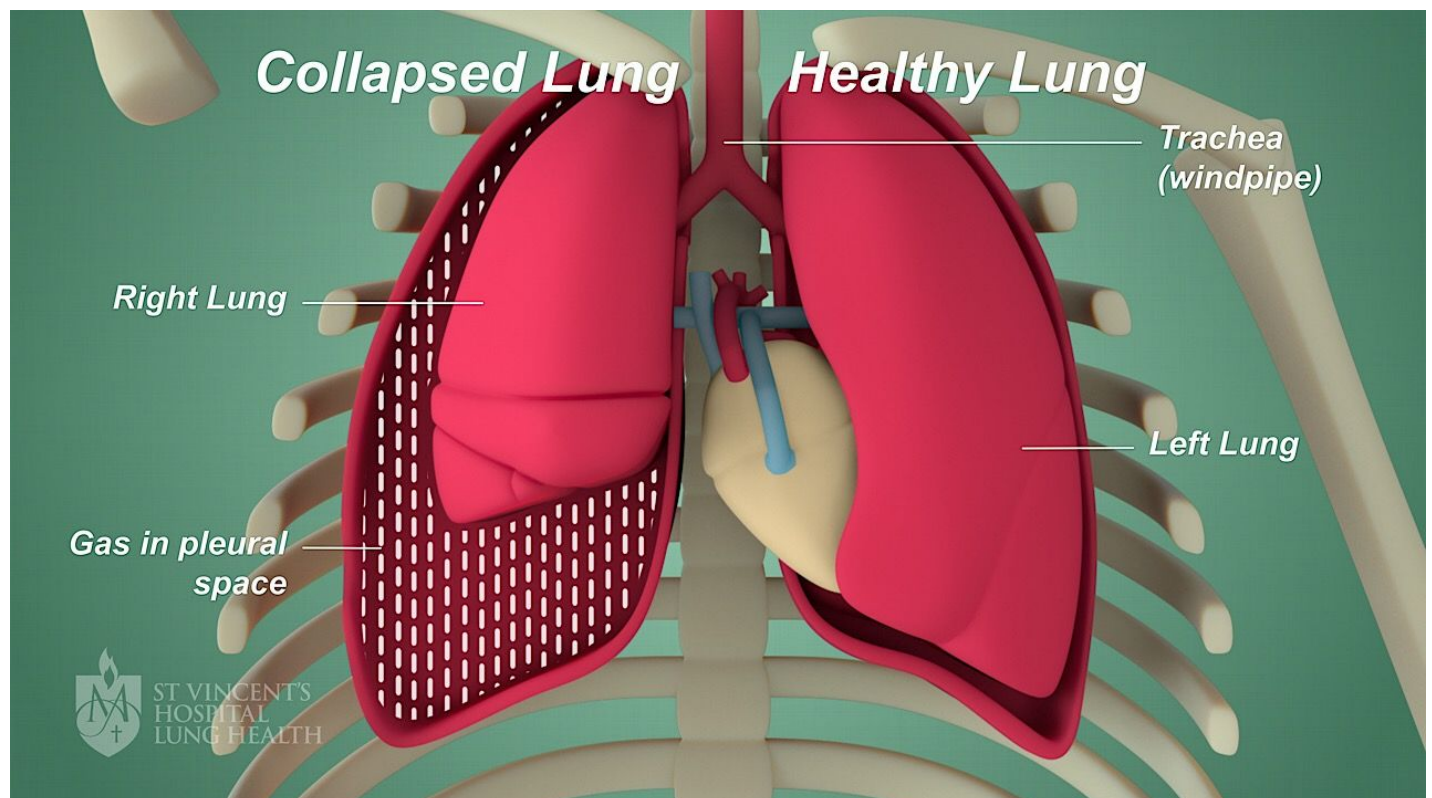


Image Source: <https://www.svhlunghealth.com.au/Images/UserUploadedImages/3457/Pneumothorax%20still.jpg>  
(<https://www.svhlunghealth.com.au/Images/UserUploadedImages/3457/Pneumothorax%20still.jpg>).

## What is Pneumothorax disease?

A pneumothorax is when the lung has collapsed due to air entering the space around your lungs (known as the pleural space). In a healthy body, the lungs are touching the walls of the chest. Air can enter the pleural space through an opening in your chest wall or in the lung. Air in the pleural space creates an increase in pressure around the lung and causes it to collapse. The lung may fully collapse, but most often only a part of it collapses. This collapse can also put pressure on the heart, causing further symptoms.

A pneumothorax can be severe, depending on how much air is trapped in the pleural space. A small amount of trapped air can usually resolve by itself, provided there are no other complications. Larger amounts of trapped air can be serious and lead to death if medical treatment is not obtained.

## Causes:

### 1. Primary spontaneous:

Primary spontaneous pneumothorax (PSP) occurs in young people (aged 15-34) without any history of lung disease. The direct cause of PSP is unknown. People at risk include smokers, tall men and those who have had a family member with a pneumothorax.

### 2. Secondary spontaneous:

Secondary spontaneous pneumothorax (SSP) can be caused by a variety of lung diseases (such as chronic obstructive pulmonary disease, cystic fibrosis, tuberculosis, pneumonia, lung cancer, sarcoidosis, pulmonary fibrosis or cystic lung diseases) and tissue disorders (such as Marfan's Syndrome). SSP carries more serious symptoms than PSP, and it is more likely to cause death.

### 3. Traumatic pneumothorax:

A traumatic pneumothorax is the result of an impact or injury. Potential causes include blunt trauma or an injury that damages the chest wall and pleural space. One of the most common ways this occurs is when someone fractures a rib. The sharp points of the broken bone can puncture the chest wall and damage lung tissue. Other causes include sports injuries, car accidents, and puncture or stab wounds.

### 4. Tension pneumothorax

Tension pneumothorax is caused by a leak in the pleural space that resembles a one-way valve. As a person inhales, the air leaks into the pleural space and becomes trapped. It cannot be released during an exhale. This process leads to increased air pressure in the pleural space that is life-threatening and needs immediate treatment.

## Symptoms:

1. Shortness of breath
2. Abnormally fast heart rate (known as tachycardia)
3. Chest pain, which may be more severe on one side of the chest
4. Sharp pain when inhaling
5. Blue discoloration of the skin or lips
6. Cold sweats

Some cases of pneumothorax have almost no symptoms. These can only be diagnosed with an X-ray or another type of scan.

## Diagnosis:

Diagnosis of a pneumothorax is typically done via a **chest X-ray**, which takes images to detect the presence of air in the pleural space (area round the lungs). A CT scan and thoracic ultrasound can also be used to help diagnose a pneumothorax.

A doctor (radiologist) then examines the X-ray report to diagnose pneumothorax and the affected area.

## Business Problem:

Pneumothorax is typically detected on chest X-ray, examined by a doctor or a radiologist. But this requires manual effort. Since current imaging volumes are large in number, it takes a long time to review every image and prepare report. Pneumothorax can cause life threatening emergency due to lung collapse and respiratory or circulatory distress if it is not detected early.

Our objective is to build an automated method to predict X-rays with pneumothorax and segmentize the affected area. This will help to prioritize the treatment of patients with pneumothorax. Automatic image segmentation can assist doctors in the treatment and diagnosis of diseases with higher accuracy, accelerate diagnosis process, and improve efficiency.

## DL Formulation:

Build an automatic image segmentation model using deep convolution network to predict pneumothorax and segmentize the affected area. We will use the given images and RLE masks to train our model.

## Business Constraints:

1. There is no such latency constraints for this problem but the model should predict within few minutes.
2. Along with prediction the model should segmentize the pneumothorax affected area.

## Dataset column analysis:

```
In [1]: import pandas as pd
train_rle = pd.read_csv("train-rle.csv")
train_rle.head()
```

Out[1]:

	Imageld	EncodedPixels
0	1.2.276.0.7230010.3.1.4.8323329.6904.151787520...	-1
1	1.2.276.0.7230010.3.1.4.8323329.13666.15178752...	557374 2 1015 8 1009 14 1002 20 997 26 990 32 ...
2	1.2.276.0.7230010.3.1.4.8323329.11028.15178752...	-1
3	1.2.276.0.7230010.3.1.4.8323329.10366.15178752...	514175 10 1008 29 994 30 993 32 991 33 990 34 ...
4	1.2.276.0.7230010.3.1.4.8323329.10016.15178752...	592184 33 976 58 956 73 941 88 926 102 917 109...

The given data consists of Imageld and EncodedPixels. For every Imageld we have an image in DICOM format. EncodedPixels with '-1' value indicates the image is without pneumothorax. Images with pneumothorax have mask in run-length-encoded (RLE) format. We have to decode and create the mask.

## Performance metric:

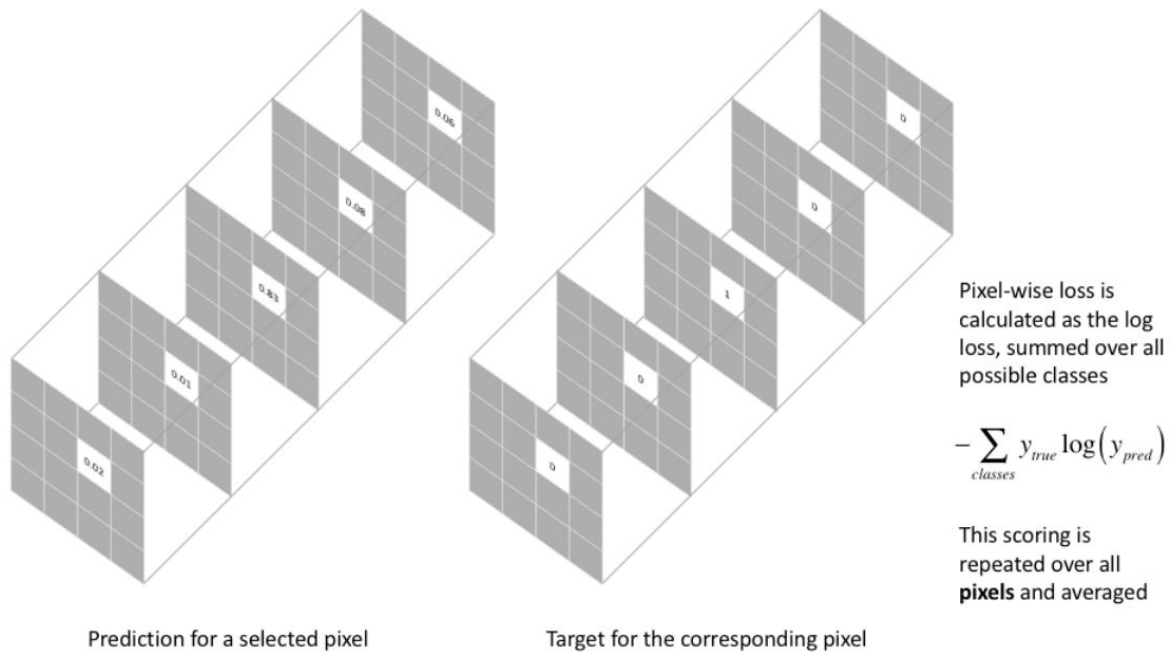
I have given the images along with masks. I have to train model using that data and predict masks for the test data. So, this is a Semantic Image Segmentation problem.

In this Semantic Image Segmentation problem I am going to use 2 most commonly used loss metric.

reference: <https://www.jeremyjordan.me/semantic-segmentation/#loss> (<https://www.jeremyjordan.me/semantic-segmentation/#loss>).

### 1. Pixel-wise cross entropy loss:

This loss examines each pixel individually, comparing the class predictions (depth-wise pixel vector) to our one-hot encoded target vector. Pixel wise loss is calculated as the log loss summed over all possible classes.



### 2. Dice loss:

Dice Loss = 1 - Dice Coefficient

Where Dice Coefficient(D) =

$$D = \frac{2 \sum_i^N p_i g_i}{\sum_i^N p_i^2 + \sum_i^N g_i^2}$$

Fig.3: Dice coefficient

Here,  $p_i$  = predicted pixel values.

$g_i$  = ground truth pixel values.

In image segmentation scenario the values of  $p_i$  and  $g_i$  are either 0 or 1.

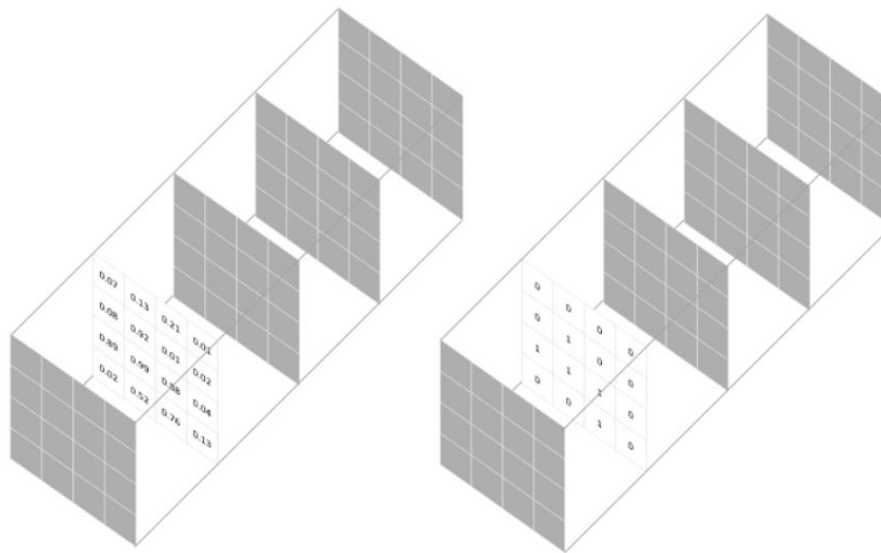
1 --> pixel is a boundary

0 --> pixel is not a boundary

In the dice coefficient,

Numerator --> 2 \* Sum of correctly predicted boundary pixels. (when pi and gi both are 1)

Denominator --> Sum of total boundary pixels of both predicted and ground truth.



Prediction for a selected class

Target for the corresponding class

Soft Dice coefficient is calculated for each class mask

$$1 - \frac{2 \sum_{\text{pixels}} y_{\text{true}} y_{\text{pred}}}{\sum_{\text{pixels}} y_{\text{true}}^2 + \sum_{\text{pixels}} y_{\text{pred}}^2}$$

This scoring is repeated over all classes and averaged

### 3. Intersection over Union (IoU) Score:

reference: <https://www.jeremyjordan.me/evaluating-image-segmentation-models/> (<https://www.jeremyjordan.me/evaluating-image-segmentation-models/>)

The Intersection over Union (IoU) metric, also referred to as the Jaccard index. This is a method to quantify the percent overlap between the target mask and our prediction output. This metric is closely related to the Dice coefficient.

The IoU metric measures the number of pixels common between the target and prediction masks divided by the total number of pixels present across both masks.

$$IoU = \frac{\text{target} \cap \text{prediction}}{\text{target} \cup \text{prediction}}$$

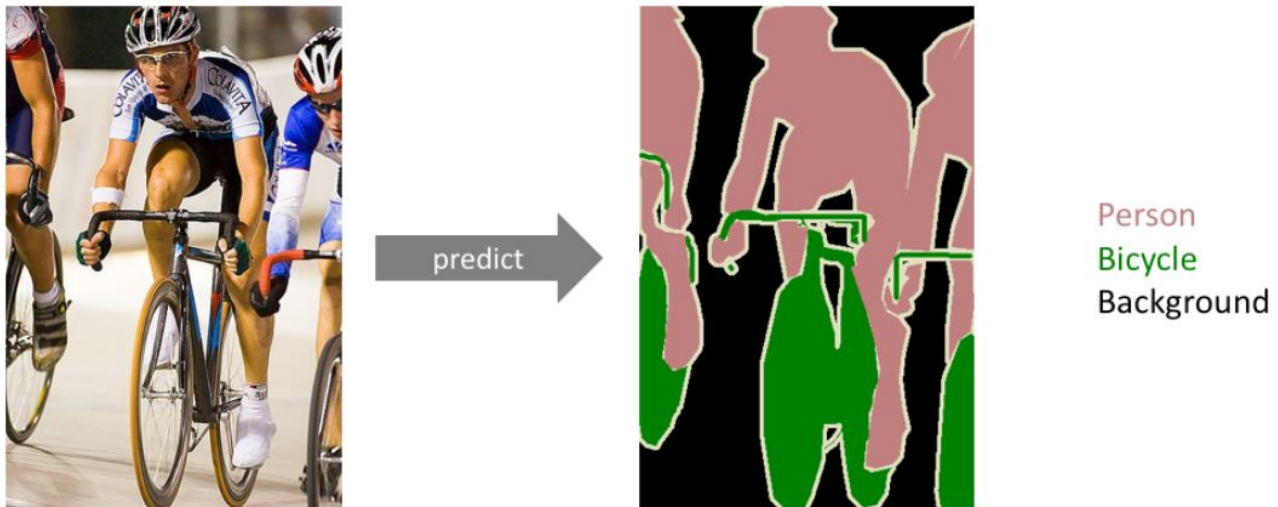
## Research-Papers/Solutions/Architectures/Kernels

Mention the urls of existing research-papers/solutions/kernels on your problem statement and in your own words write a detailed summary for each one of them. If needed you can include images or explain with your own diagrams. it is mandatory to write a brief description about that paper. Without understanding of the resource please don't mention it.

### 1. Blog: An overview of semantic image segmentation.

<https://www.jeremyjordan.me/semantic-segmentation/> (<https://www.jeremyjordan.me/semantic-segmentation/>)

In this blog it is discussed that how convolution neural network is used for semantic image segmentation. Image segmentation is a computer vision task in which we label specific regions of an image according to what's being shown.



An example of semantic segmentation, where the goal is to predict class labels for each pixel in the image. [\(Source\)](#)

The goal of semantic image segmentation is to label each pixel of an image with a corresponding class of what is being represented. Here we are not separating instances of the same class. If we have 2 objects of the same category in the input image, we are not separating them. In the above image we are separating Person, Bicycle and Background. But we are not separating individual person.



- 0: Background/Unknown
- 1: Person
- 2: Purse
- 3: Plants/Grass
- 4: Sidewalk
- 5: Building/Structures

Our objective is to take an input image (RGB-  $\text{height} \times \text{width} \times 3$  / Grayscale -  $\text{height} \times \text{width} \times 1$ ) and output a segmentation map where each pixel contains a class label represented as an integer ( $\text{height} \times \text{width} \times 1$ ).

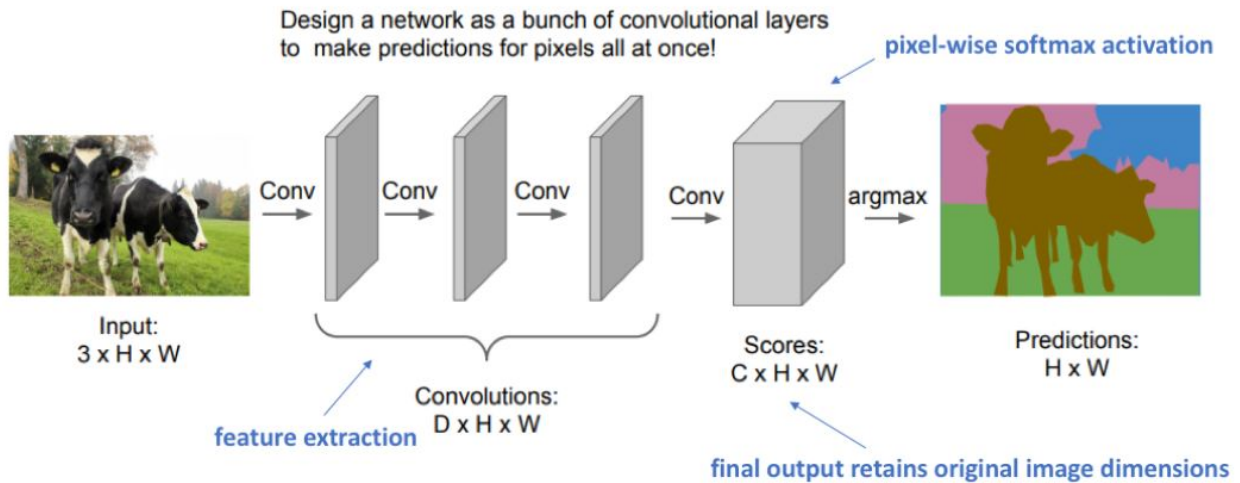
For this task we will create our target by one-hot encoding the class labels i.e creating an output channel for each of the possible classes. So, the number of output channel will be equal to number to target labels. Then a prediction can be collapsed into a segmentation map by taking the argmax of each depth-wise pixel vector to get a output shape =  $\text{height} \times \text{width} \times 1$ .

## Constructing an architecture:

### Approach 1:

#### Stack a number of convolutional layers:

This is a naive approach towards constructing a neural network for this semantic segmentation task. In this approach we need to simply stack a number of convolutional layers (with same padding to preserve dimensions) and output a final segmentation map.



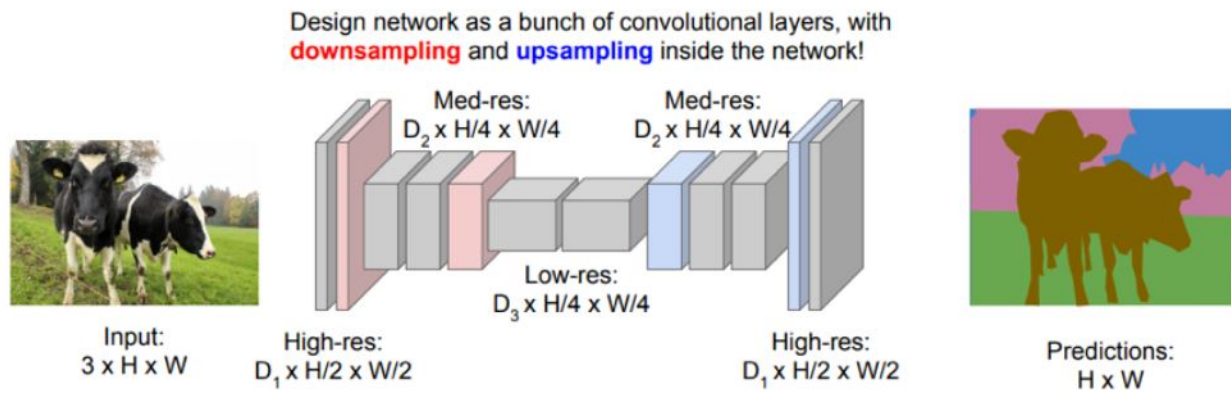
**Downside:** Preserving image dimensions throughout entire network will be computationally expensive.



## Approach 2:

### Encoder/Decoder structure:

Here we downsample the spatial resolution of the input, developing lower-resolution feature mappings which are learned to be highly efficient at discriminating between classes, and then upsample the feature representations into a full-resolution segmentation map.

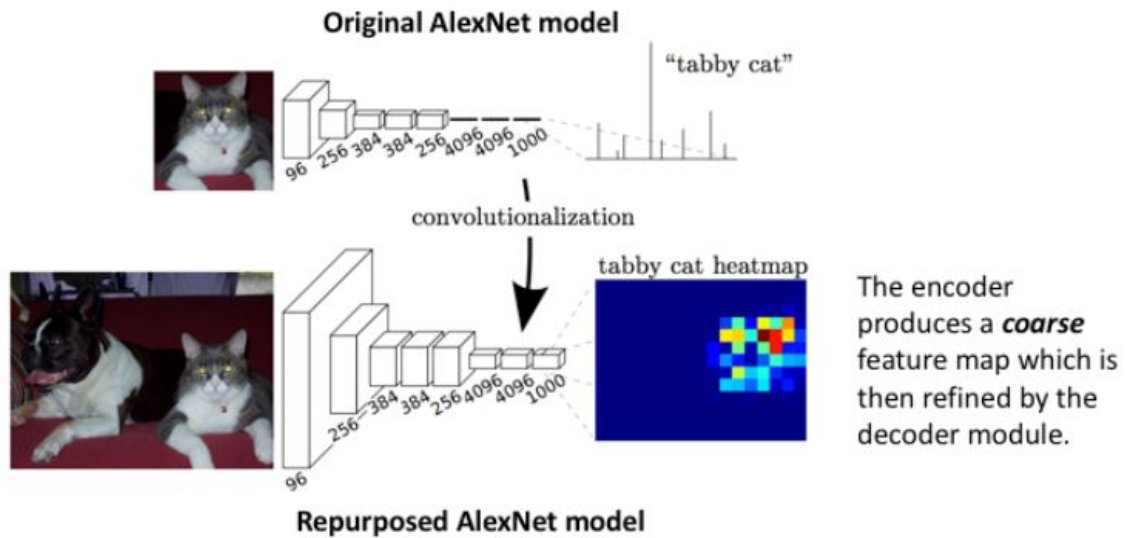


**Solution:** Make network deep and *work at a lower spatial resolution* for many of the layers.

### Approach 3:

#### Use AlexNet as encoder appending a decoder module with transpose convolutional layers:

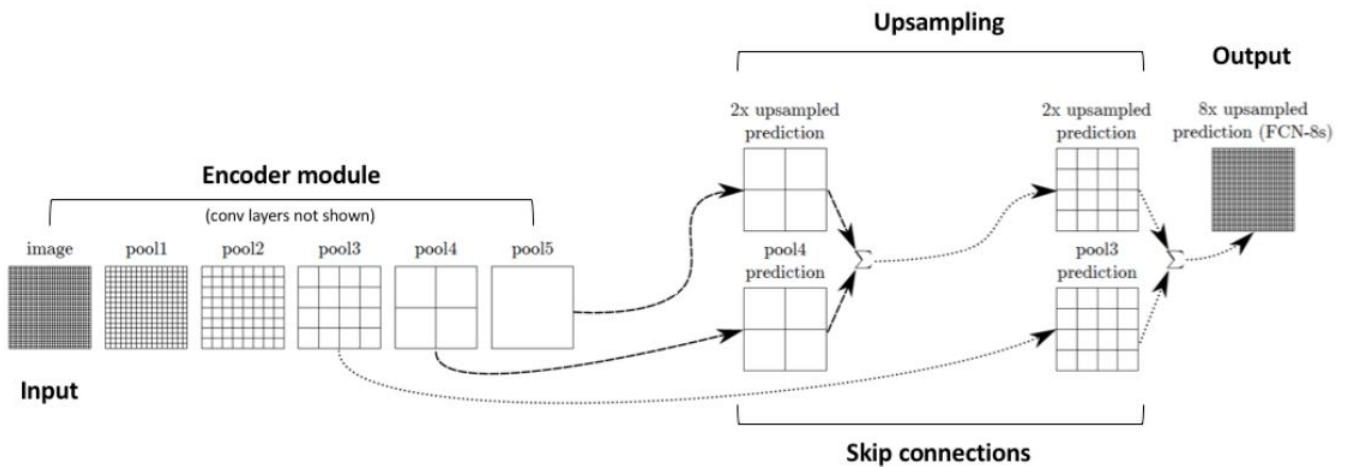
The author used existing image classification AlexNet network to serve as encoder module of this network, appending a decoder module with transpose convolutional layers to upsample the coarse feature maps into a full-resolution segmentation map.



## Approach4:

### U-Net architecture:

In the previous AlexNet based model, the encoder module reduces the resolution of the input by a factor of 32, the decoder module struggles to produce fine-grained segmentations. This is because semantic segmentation faces an inherent tension between semantics and location. To address this problem the author added skip connections from earlier layers and summing feature maps. These skip connections from earlier layers in the network (prior to a downsampling operation) should provide the necessary detail in order to reconstruct accurate shapes for segmentation boundaries.



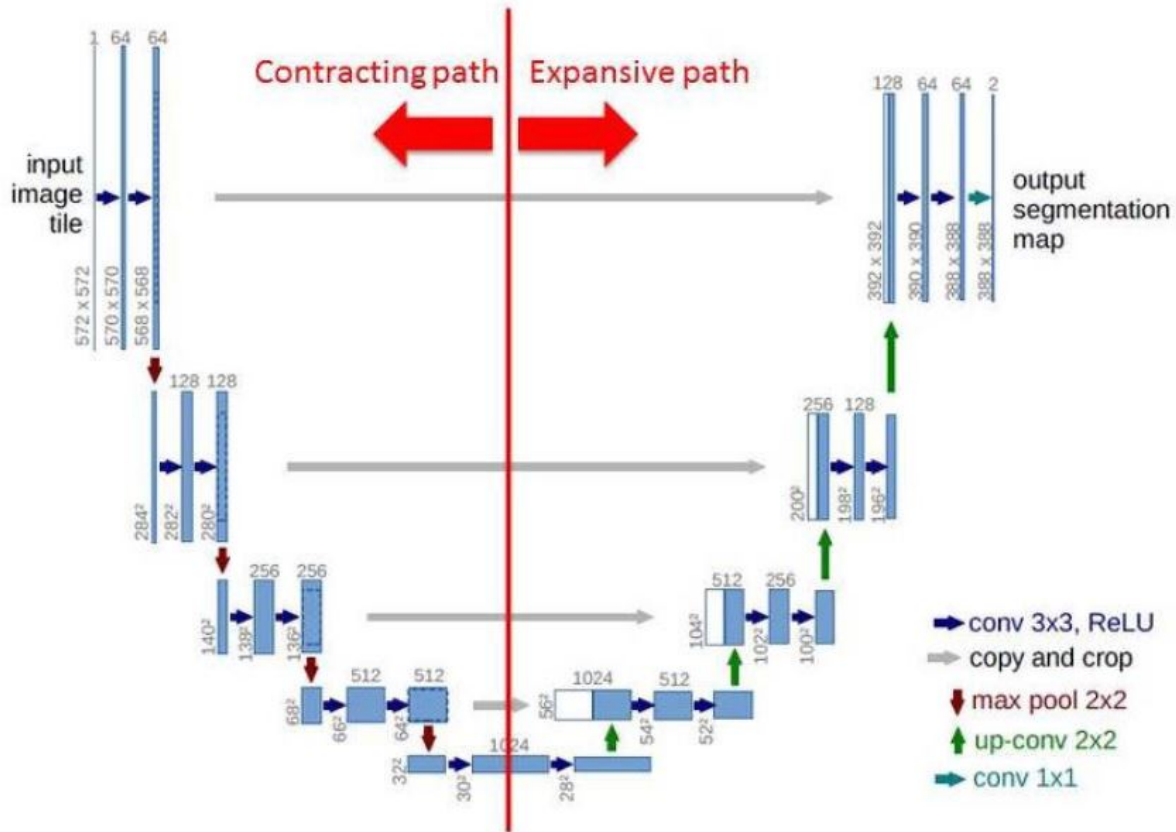
More concretely the concept is proposed in U-Net architecture which consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. This U-Net architecture is very popular for image segmentation. There are some advanced variants of U-Net which performs better than U-Net for image segmentation task.

## 2. Research Paper

### U-Net: Convolutional Networks for Biomedical Image Segmentation

<https://www.semanticscholar.org/paper/U-Net%3A-Convolutional-Networks-for-Biomedical-Image-Ronneberger-Fischer/6364fdaa0a0eccd823a779fcdd489173f938e91a> (<https://www.semanticscholar.org/paper/U-Net%3A-Convolutional-Networks-for-Biomedical-Image-Ronneberger-Fischer/6364fdaa0a0eccd823a779fcdd489173f938e91a>)

# Network Architecture



The left part of this U-Net architecture is contracting path which consists of encoders. This path is also known as downsampling path. The right part is called expansive path and consists of decoders, also known as upsampling path.

Skip connections are there in between from encoder to decoder, this connections helps to get localizations of the objects in semantic segmentation task.

The contracting path follows the typical architecture of a convolutional block. Each convolution block consists of 2 convolution layer followed by a maxpool layer. In the last convolution block of encoder part there is no maxpool layer. This downsampling part increase the number of layers of the image and decrease the resolution of the image. We are capturing the context information using this encoder part.

In expansive path also we have convolution blocks. Each convolution block consists of 2 convolution layer followed by a transpose convolution or upsampling layer. This part increase the size of the image and decrease the number of layers. For every decoder block concatenation is done from encoder to decoder to get the localization information. Encoder part helps to find out the type of the object and decoder part helps to get the localization information.

### 3. Research Paper

#### DC-UNet: Rethinking the U-Net Architecture with Dual Channel Efficient CNN for Medical Images Segmentation

<https://www.semanticscholar.org/paper/DC-UNet%3A-Rethinking-the-U-Net-Architecture-with-CNN-Lou-Guan/5fb044a4a3925f1d95260af9d5b95e47c1ee7e3c> (<https://www.semanticscholar.org/paper/DC-UNet%3A-Rethinking-the-U-Net->

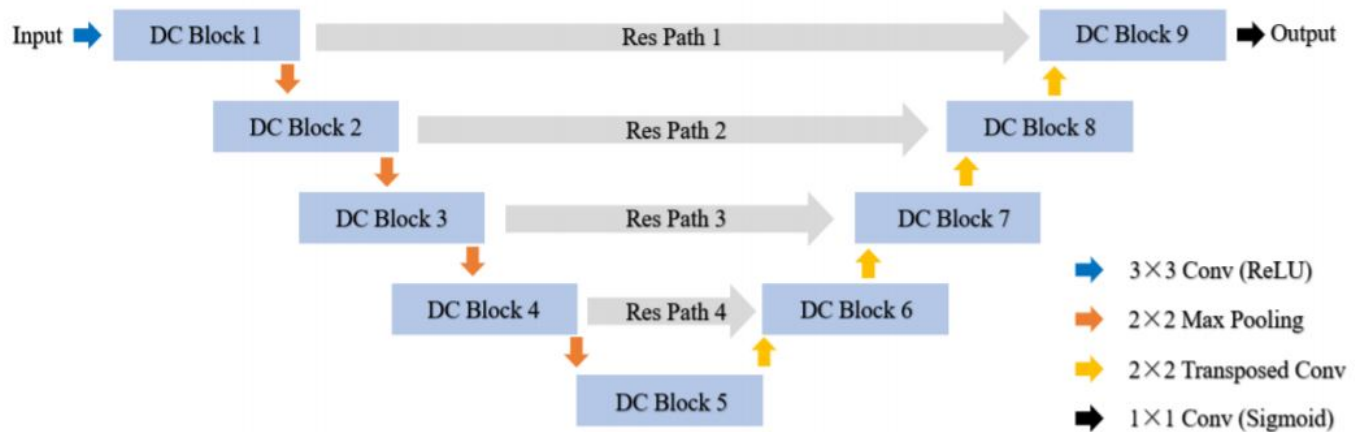


Fig. 8. Architecture of DC-UNet

Even though U-Net performs well in medical image segmentation task, there is some limitations of it. DC-UNet comes up with some modifications of classical U-Net model.

1. Designed efficient CNN architecture to replace encoder and decoder.
2. Applied residual module to replace skip connection between encoder and decoder to improve based on the-state-of-the-art U-Net model.
3. Used the Tanimoto similarity to replace the Jaccard similarity for gray-to-gray image comparisons.

With these modifications DC-UNet is a potential successor to the U-Net architecture.

## 4. Existing Solution:

### 2nd Place Solution:

This solution consists of 2 parts. Classification and Segmentation.

#### Classification:

In this part Unet is used to classify whether an image is pneumothorax or not.

Loss = BCE + focal loss

Augmentation: hflip, scale, rotate, bright, blur

Backbone: seresnext 50, seresnext101, efficientnet-b3

#### Segmentation:

Two models are used to segment, unet and deeplabv3.

Imaged which consists of pneumothorax is used.

Loss = Dice Loss

Augmentation: hflip, scale, rotate, bright, blur

Backbone: seresnext 50, seresnext101, efficientnet-b3

## 5. Existing Solution:

### 6th Place Solution:

<https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107743> (<https://www.kaggle.com/c/siim-acr-pneumothorax-segmentation/discussion/107743>)

This solution is based on EncodingNet (ResNets, 512 and 1024 size) and UNet (EfficientNet4, se-resnext50, SENet154 with 512, 640 and 1024 sizes).

1. Best augmentations were related to crops and rotations.
2. contrast and brightness transformations didn't work well.
3. Loss: BCE + Dice
4. Lower image sizes reduced score a lot so model with full sized image resulted better.

## First Cut Approach:

### Data Collection:

Data is provided by the organizer in kaggle platform. So, I don't need to spend time do data collection.

<https://www.kaggle.com/seesee/siim-train-test> (<https://www.kaggle.com/seesee/siim-train-test>)

### Data Cleaning:

X-ray images are provided in DICOM format, which consists of image along with additional informations. I need to extract these informations for EDA. Then I need to convert these images in PNG format to train my model.

Masks are given in run length encoded format, I need to decode it to create mask in PNG format for every image.

I need to split the data into train and validation set for every image along with it's original mask.

Augmentations I want to use:

1. crop
2. rotation
3. hflip
4. bright
5. blur

### **Performance Metric:**

Loss = Cross Entropy Loss + Dice Loss

Score = IoU Score

### **Modelling:**

First I am going to use pretrained models. For this task I will use the below combinations.

Model: Unet and Linknet

Backbone: Resnet34, Resnet50

Encoder Weight: imagenet

I will also define my own Unet model and try to do some modifications to make it work better.

Then I will define some advanced Unet model like DC-UNet. I will train the model and compare the performances to pick the best one.

### **Prediction:**

After model is trained I will predict the validation mask and plot original image along with the original and predicted mask.