Pneumonia X-ray Classification based on CNN

Yaran Nan, Yuanyuan Gan

y4nan@uwaterloo.ca, ganyuanyuancc@gmail.com

December 21, 2018

Abstract

Medical image classification is a simple application and it still has the potential to be improved. This can be utilized in multiple ways to provide valuable information. Especially, it is difficult to process the small amount and unbalanced data-set. In this paper, we try to present a pneumonia classification system which is based on modified VGG to assist radiologists in pneumonia diagnosis. The model was trained and tested based on an open source pneumonia x-ray data-set. The experiment results demonstrate that our fine-tuned VGG model can achieve a precise classification. The result proves the strong ability of modified VGG in classifying the small data collections.

Keywords— Lung pneumonia, X-ray, VGG, CNN, Neural Networks

1 Introduction

1.1 Background

1.1.1 Pneumonia

In the medical field, the Pneumonia is described as an infection of the lung tissue. One or both side of a patient's lung will not be working properly, and they filled with microorganisms, some fluid, and inflamed cells. There are few types of pneumonia. There commonly are bacterial pneumonia, viral pneumonia, mycoplasma pneumonia, and others. Bacterial pneumonia is frequently caused by various bacteria. It can affect all age. Viral pneumonia is usually caused by various viruses, such as flu (influenza), and this type caused almost one-third of pneumonia cases. Mycoplasma can cause mild, widespread pneumonia that affects all age groups [1]. Mycoplasma pneumonia comes with different symptoms. It comes with a severe cough produces mucus [2]. Mycoplasma pneumoniae is recently determined as an important cause of community acquired pneumonia (CAP) worldwide [3]. Pneumonia accounts for 16 percent of all death of children under 5 years old [4]. Pneumonia can be one of the major risk factors for adult pulmonary disease.[5]. Pneumonia diagnosis is based on patient history and physical exam from clinics additionally with radio-logical imaging commonly chest X-ray [6]. The major diagnosis of pneumonia is based on symptoms and confirmed by a chest X-ray image [7]. The image is looking for inflammation in a patient's lungs; therefore it is the best test for diagnosing pneumonia. It usually combines with a blood test obtain a complete blood count (CBC) have a deep understanding of whether the patient has a bacterial infection that have spread to the bloodstream. Additionally, there is sputum test, Chest computed tomography (CT) scan, plural fluid culture and so on[8].

1.1.2 Pneumonia Chest X-ray

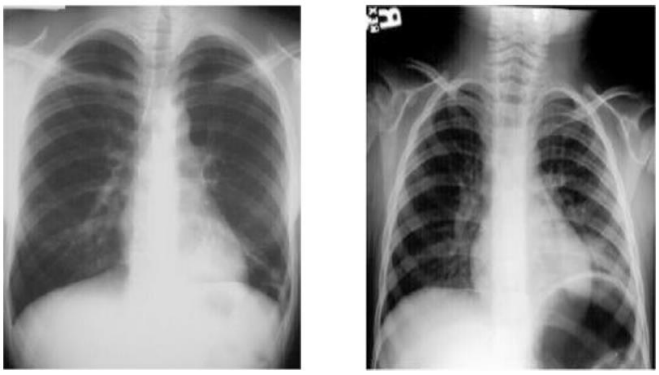


Figure 1: X-ray image for left lower lobe and round pneumonia

Figure 2: X-ray image for Right upper lobe and right middle lobe

In the chest x-ray, the radiologist always looks for infiltrates (white shadows) in the lungs which identified as an infection. It also helps to determine if the patient has any complications related to pneumonia. Pneumonia consolidation depends upon the amount and distribution of the air spaces involved. This could be present as confluent lobar or segmental

opacity. If the interstitium is preponderantly concerned, it may appear as a reticulonodular pattern. Air bronchograms would confirm an alveolar process. The lung volume should not be lost and may be increased. Pneumonia could also be difficult by symptom or inflammatory disease formation. Refer to the figures for few chest x-ray examples of pneumonia and their locations[9]. However, pneumonia detection in chest radiography can be difficult for radiologists. The presentation of pneumonia X-ray images is mostly overlapped and not clear. It can mimic many other abnormalities [10].

1.2 Introduction of CNN

Convolution neural networks (CNN) is a special architecture of artificial neural networks. CNN utilize layers with convolution filters that are applied to local features. It applies commonly in visual recognition tasks such as image classification, localization, and detection. CNN has been proved effective and productive in image classification [11]. CNN is applied in many medical image processing procedure, and it is a good fit for analyzing X-ray images due to the end-to-end network and easiness of training [12]. In order to build a convolution Neural, there are four major steps: convolution, pooling, flattening, and full connection.

The first layer is always the convolution layer. Convolution networks were originally sourced from biological processes. The connection between neurons reorganize the animal visual cortex. Individual cortical neurons respond to stimuli only in the receptive fields which is known as a restricted region of the visual field. The different neurons overlap so that they can fully cover the visual field. CNN's use relatively little prepossessing compared to other image classification algorithms. There are many CNN mature architectures are used and developing, and our model is based on the VGG structure.

1.3 Research Motivation

There is an massive amount of X-ray images taken in hospitals and they must be graded by radiologists. Manually examining those scans is time consuming and subjective [13]. The use of machine learning in the medical field has significantly grown in recent years, and the open source data-sets have become more readily available [14]. It provides us the ground point to perform image scanning automatically to assist radiologists. In the project, we focuses on parameter fine-tuning and evaluating how well the result can be. The chosen data has some special characteristics which are useful for observing the performance of the CNN. As we all know, CNN works well on a large data-set. However, our data-set is relatively small, what types of results can give by CNN in the project is what we are concentrating. How to design the network and perform the fine-tuning during training is a challenge for us in the project. It will be a meaningful try for solving a similar problem in the future.

2 Material and Methods

2.1 Data

The chest x-ray image data is open source and it was retrieved from Kaggle. These Chest X-ray images were selected one to five years old patients from Guangzhou Women and Children's Medical Center. For the better analysis of chest x-ray images, all of the chest radio-graphics were initially screened by removing all low quality and unreadable images. The diagnoses for the images were then graded by two experienced physicians before being used in training the AI system. In order to avoid for any errors, the evaluation set was also assessed by a third expert. The image format is in JPEG. The image set is organized with train, test, and validation and it contains 5863 images in total. Training set has total 5216 images with 3875 pneumonia images and 1341 normal images respectively. The validation set is quite small, only has 8 pneumonia and 8 normal images. Test set has 624 images, 390 of them are pneumonia while other 234 images are labeled as normal. Refer to the figure below, it clearly showed the set of image data is imbalanced. The X-ray images shown below are from the set. The orange bar is a normal lung and the blue bar is diagnosed with pneumonia.

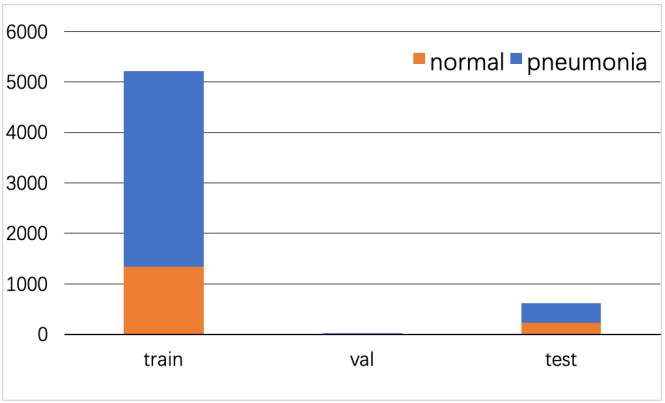


Figure 3: Image data distribution of Normal vs Pneumonia for training, validation and testing

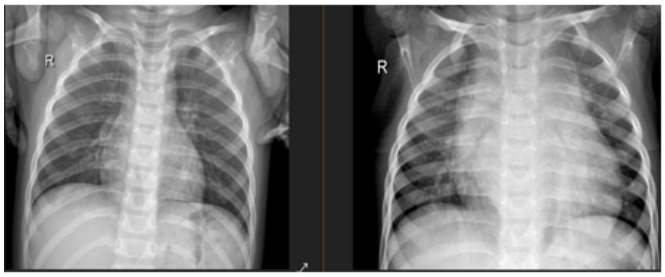


Figure 4: Normal vs Pneumonia X-ray image

2.2 VGG Architecture

VGG is one of the CNN architectures which is wildly applied in the classification task. It is from the Visual geometry group, Department of Engineering Science at the University of Oxford. The Oxford Visual Geometry Group's deep convolution neural network is designed for object recognition and modified and adapted to the task of utility estimation [15]. It is a generic design can be configured; therefore, it is a changeable structure. Convolution layers with a small filter size of 3*3 from 11 weights layers (8 conv-layers and 3 full connected lay-

ers) to 19 weights layers (16 conv-layers and 3 full connected layers). The width of conv-layer is rather small, starting from 64 to 512 [16]. It did a great job in image classification and localization. It is simple and efficient to be used in a deep neuron network framework. The structure is simple and elegant while trained weights can be used as transfer learning weights. The model in the project based on the configuration of VGG and after fine-tuning to get the best result of this particular medical image classification.

2.3 Software and Hardware set-up

The programming language using in this project is python, running on the Jupyter Notebook. The major package pool is Keras, which is a high-level neural networks API. It is capable of running on top of TensorFlow. It seamlessly runs on CPU and GPU. The GPU used for our experiment is from Google Cloud Platform's virtual engine, the type is NVIDIA Tesla P100.

3 Experiments

3.1 Experiment design

3.1.1 Training

The Convolution Neuron Network are training base on mini-batch gradient decent and back-propagation. The batch size is 16. Because the data-set is not large and the network is not deep, so we decide to use random weights to be initialized. We use Adam as our optimizer. Adam is an first-order gradient-based optimization algorithm that extends stochastic gradient descent. It is based on adaptive estimates of lower-order moments. It is computationally efficient and consumes little memory. The learning rate is set to be 1×10^{-3} as default and based on the training and testing result for each trial, divided roughly by 2 to improve the result. The choices of learning rates are 1×10^{-3}, 5×10^{-4}, 2×10^{-4}, 1×10^{-4}, and 5×10^{-5}. The activation function is using ReLU the same as the VGG configuration. The loss is calculating by 'categorical crossentrop' which is multi-class log-loss.

The model architecture is based on the structure of VGG. Firstly, use 13 convolution layers with a 3*3 filter in 5 blocks and 3 full connected layers. Additionally, each block followed by one 2*2 max-pooling layer. Because the classification of our task is two classes: pneumonia and normal, The last fc layer has the output size of 2 with"soft-max" function. Carefully chose 1024 and 512 as other two full connected layers' hyper-parameters. During fine-tuning, the structure may be slightly changed by adding more or deleting several conv-layers to increase or decrease the depth of the neuron network. Add batch normalization layers between convolution layers and dropout layers between full connected layers. Finally, wisely choose the hyper-parameter of dropout layers.

3.1.2 Training Image Size

The input image of convolution neuron network is fixed to 224*224 the same as the VGG architecture. The network will catch the whole original images. We try two color modes: one is gray-scale and the other one is RGB. With these two color-modes, the input images will be converted to have 1 or 3 channels. Whereas there is no significant difference between these two-color modes during fine-tuning.

3.1.3 Testing

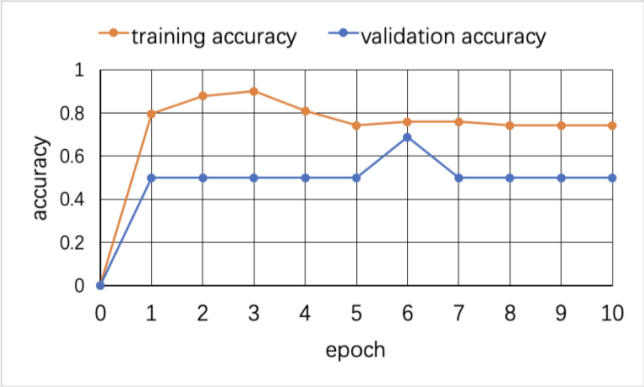
Given the trained neuron network, test images are also fixed to the size of 224*224 and then calculated through the neuron network forward from the first layer to the last full connected layer. The output for each image is an array of shape (*, 2). The return after testing is an array of predictions with integer 0 or 1 (normal or pneumonia). One method to measure the classification is to measure the testing accuracy. While in this particular data-set, one problem is the data-set is not balanced. And another consideration in the medical area, is we always prefer pneumonia image should be detected, so the recall is a very important criterion of whether the classifier is useful. We also plot a confusion matrix to make sure the precision is also acceptable.

3.2 Experiment process

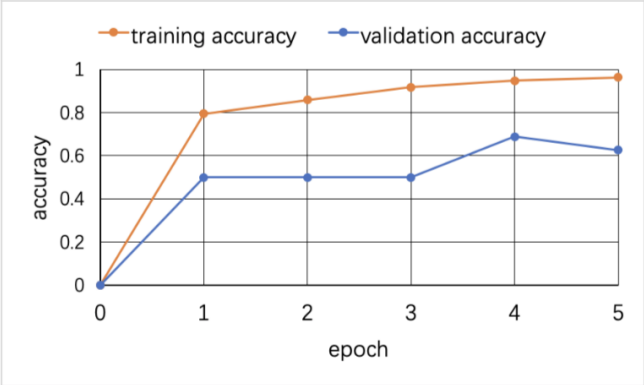
Learning rate is the first hyper-parameter to set for the fine-tuning process. For general neural network training, we choose several learning rates and sweep out ones with bad behavior during training. In our experiment, due to the number of the epoch is small, we pick one leaning rate and see if it is suitable for the particular model. During training, it is easy to decide whether a smaller learning rate should be chosen. In Figure 5 (a), the training accuracy is stuck at a low point. This is a training example of a situation in which we should choose another learning rate. We divide the learning rate by 2 roughly then train the data again.

Because the data-set is small, we face the over-fitting many times as shown in Figure 5 (b), the training accuracy is high while validation accuracy is decreasing. One method we tried is to delete several convolution layers of structure. This has a side effect on the process and a fairly good structure for our training data is omitting the last block of convolution layers. In addition, adding batch-normalization layers and dropout layers improves the result and the training process is showing in Figure 5 (c). Our validation set only has 16 images, so the validation accuracy change quite a bit by just predicting one or two images different. But the overall trends can be detected during training.

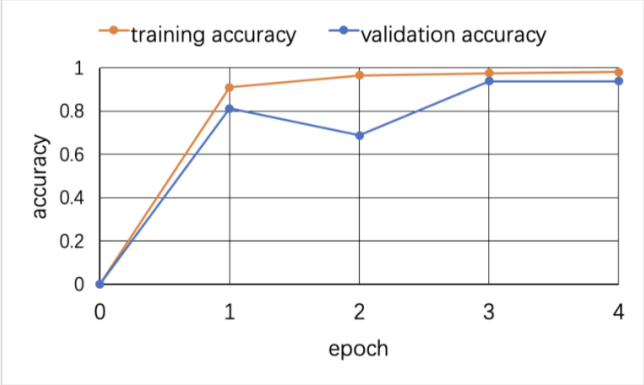
One interesting observation during fine-tuning is the relationship between test accuracy and recall. The figure below shows some model cases' results of test accuracy and recall. We observed that almost every model has a very high recall, and some even achieve up to 1. It means all pneumonia images have been successfully detected. However, as recall becomes close to 1, the accuracy decreases. Looking at the training procedure, when this situation occurs, the accuracy of each epoch has similar behavior. The training accuracy reaches a very high point and becomes stable, while the validation accuracy is not much high compared to the training accuracy. Since the data is small and there are only 16 validation images, the validation accuracy during the training is assisting fine-tuning. This leads us to look back into the input data. Our data-set is unbalanced, for training data, pneumonia vs normal is around 3:1. As for test images, the number of two types of images is 390 and 234, and the ratio is around 2:1. If all the test images are labeled to 1(pneumonia), the accuracy



(a)



(b)



(c)

Figure 5: Three examples of training process during fine-tuning

is 0.64. For training data, the number of pneumonia images is more than normal images, the training process will have the ability to catch more pneumonia features than normal features. In other words, the neural network will be influenced by pneumonia label more than by normal label, the learning result has a bias. This directly projected on the test data, the recall is high for almost all the cases, even it is not over-fitting. As for over-fitting, in this particular situation, the test behavior is similar. Therefore, after these considerations, we do not choose the model which has the best recall.

	Model1	Model2	Model3	Model4	Model5
Recall	0.99	0.99	0.97	0.88	1
Accuracy	0.76	0.79	0.83	0.85	0.71

Figure 6: Five examples of test results during fine-tuning

4 Results

After 19 times fine-tuning and based on the principle of choosing the best model, the best result we achieved is showed in Figure 7 and the Figure 8 shows its confusion matrix. The recall is around 0.97, and test accuracy is over 80 percent. When first trained the data, the accuracy was only around 70 percent and there is a huge jump between before and after fine-tuning. There is an improvement of the model to fit this particular cases by looking inside of the input images to find out the way of fine-tuning. The result is quite meet our expectation and we can say this model is a successful classifier.

In the model, there are 10 convolution layers in four blocks. It consists of one pooling layer in each block, three full connected layers, two batch normalization layers in the fourth block, and two dropout layers after the first two full connected layers. The hyper-parameters of dropout layers are both 0.5 and 0.5. The learning rate chosen is 5×10^{-5}. The number of epochs is 4 because the data-set is not large and has a trend of over-fitting of a large number of epochs.

As we mentioned before, our data-set is from Kaggle. There are other people attempted to solve this problem. Most of them are using CNN as the classifier. One of those works has drawn the most of attentions [17]. This model used CNN with a quite simple structure. The work also used transfer learning. the best result of his model is achieved 0.98 of recall and 0.79 of precision. We evaluated our result by comparing with other people's results. They are listed in the figure 8. Other results are quite close to our result. This implies that we have got the best result for this particular problem via CNN method.

	Recall	AUC	Accuracy	Method
Our Best Result	0.9692	0.9413	0.8301	CNN
Other 1	0.9487	Not known	Not known	CNN
Other 2	0.98	Not known	Not known	CNN
Other 3	0.99	0.90	Not known	CNN
Other 4	0.8974	0.829	0.7404	CNN

Figure 7: Our best result vs. other results from Kaggle on recall, AUC and accuracy

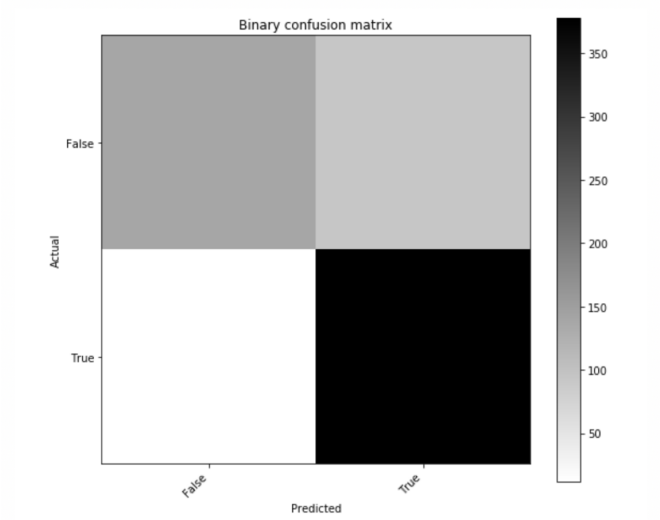


Figure 8: Confusion matrix of our best result

Layer (type)	Output Shape	Param #
conv1_1 (Conv2D)	(None, 224, 224, 64)	1792
conv1_2 (Conv2D)	(None, 224, 224, 64)	36928
pool1 (MaxPooling2D)	(None, 112, 112, 64)	0
conv2_1 (Conv2D)	(None, 112, 112, 128)	73856
conv2_2 (Conv2D)	(None, 112, 112, 128)	147584
pool2 (MaxPooling2D)	(None, 56, 56, 128)	0
conv3_1 (Conv2D)	(None, 56, 56, 256)	295168
conv3_2 (Conv2D)	(None, 56, 56, 256)	590080
conv3_3 (Conv2D)	(None, 56, 56, 256)	590080
pool3 (MaxPooling2D)	(None, 28, 28, 256)	0
conv4_1 (Conv2D)	(None, 28, 28, 512)	1180160
bn4_1 (BatchNormalization)	(None, 28, 28, 512)	2048
conv4_2 (Conv2D)	(None, 28, 28, 512)	2359808
bn4_2 (BatchNormalization)	(None, 28, 28, 512)	2048
conv4_3 (Conv2D)	(None, 28, 28, 512)	2359808
pool4 (MaxPooling2D)	(None, 14, 14, 512)	0
flatten (Flatten)	(None, 100352)	0
fc1 (Dense)	(None, 1024)	102761472
dropout1 (Dropout)	(None, 1024)	0
fc2 (Dense)	(None, 512)	524800
dropout2 (Dropout)	(None, 512)	0
fc3 (Dense)	(None, 2)	1026

Total params: 110,926,658
Trainable params: 110,924,610
Non-trainable params: 2,048

Figure 9: Model summary of detailed layers inside CNN with the layer shape and parameter number

5 Limitations

Although in the project we achieved relatively high recall, the accuracy is not over 85 percent. The fine-tuning is based on the structure of VGG, while if any other architecture do better is unknown. The data-set is small which means other methods in addition to CNN might do better than CNN. Looking inside the model itself, we all understand the basic calculations inside the neural network and we can get every layer of weights after training. However, but we still don't know much about why CNN can do the task so well. Additionally, in the medical field, the classification is not considered meaningful. If the classification can deliver more information, that will be much useful.

6 Conclusion

In the present work, we have described our pneumonia classification with convolution neural networks built on top of VGG. It experiments few VGG models and finds the best model in order to receive high recall and accuracy. Through modifying the parameters, evaluating the results and comparing these results, we found the best model. In the end, the result is also compared with others from Kaggle. It delivers a satisfying result.

References

[1] Hopkin Medicine. What you need to know about pneumonia. URL `https://www.hopkinsmedicine.org/healthlibrary/conditions/adult/infectious_diseases/pneumonia_85,P01321`.

[2] Lung Health Diseases. Pneumonia symptoms, causes, and risk factors, 2015.

[3] Phane Le Thanh Huong, Pham Thu Hien, Nguyen Thi Phong Lan, Tran Quang Binh, Dao Minh Tuan, and Dang Duc Anh. First report on prevalence and risk factors of severe atypical pneumonia in vietnamese children aged 1–15 years. *BMC Public Health*, 14(1):1304, Dec 2014. ISSN 1471-2458. doi: 10.1186/1471-2458-14-1304.

[4] unknown. Pneumonia facts. Technical report, 2016.

[5] Johnny Y.C. Chan, Debra A. Stern, Stefano Guerra, Anne L. Wright, Wayne J. Morgan, and Fernando D. Martinez. Pneumonia in childhood and impaired lung function in adults: A longitudinal study. *Pediatrics*, 135 (4):607–616, 2015. ISSN 0031-4005. doi: 10.1542/peds.2014-3060.

[6] Saeed Ali Alzahrani, Majid Abdulatief Al-Salamah, Wedad Hussain Al-Madani, and Mahmoud A. Elbarbary. Systematic review and meta-analysis for the use of ultrasound versus radiology in diagnosing of pneumonia. *Critical Ultrasound Journal*, 9(1):6, Feb 2017. ISSN 2036-7902. doi: 10.1186/s13089-017-0059-y.

[7] Pneumonia in adults: diagnosis and management, 2018. URL `https://www.ssidiagnostica.com/upload/files/english/Guidelines/UK.pdf`.

[8] Pneumonia national heart, lung, and blood institute, . URL https://www.nhlbi.nih.gov/health-topics/pneumonia,author={unknown}.

[9] unknown. X-ray atlas: Chest x-ray. *unknown*. URL https://www.glowm.com/atlas_page/atlasid/chestXray.html.

[10] Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Ding, Aarti Bagul, Curtis Langlotz, Katie Shpanskaya, Matthew P. Lungren, and Andrew Y. Ng. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *CoRR*, abs/1711.05225, 2017.

[11] Feride Karacaer, Imen Hamed, Fatih Özogul, Robert H. Glew, and Dilek Özcengiz. The function of probiotics on the treatment of ventilator-associated pneumonia (vap): facts and gaps. *Journal of Medical Microbiology*, 66(9): 1275–1285, Jan 2017. doi: 10.1099/jmm.0.000579.

[12] Yuxi Dong, Yuchao Pan, Jun Zhang, and Wei Xu. Learning to read chest x-ray images from 16000+ examples using cnn. In *Proceedings of the Second IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies*, CHASE '17, pages 51–57, Piscataway, NJ, USA, 2017. IEEE Press. ISBN 978-1-5090-4721-5. doi: 10.1109/CHASE.2017.59.

[13] Suman Sedai, Dwarikanath Mahapatra, Zongyuan Ge, Rajib Chakravorty, and Rahil Garnavi. Deep multiscale convolutional feature learning for weakly supervised localization of chest pathologies in x-ray images. In Yinghuan Shi, Heung-Il Suk, and Mingxia Liu, editors, *Machine Learning in Medical Imaging*, pages 267–275, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00919-9.

[14] Joseph Bullock, Carolina Cuesta-Lazaro, and Arnau Quera-Bofarull. Xnet: A convolutional neural network (cnn) implementation for medical x-ray image segmentation suitable for small datasets. *CoRR*, abs/1812.00548, 2018.

[15] Hemami Sheila S. Scott, Edward T. and. No-reference utility estimation with a convolutional neural network. *Society for Imaging Science and Technology*. doi: https://doi.org/10.2352/ISSN.2470-1173.2018.09.IRIACV-202.

[16] Simonyan, Karen, Zisserman, and Andrew. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2015.

[17] . URL https://www.kaggle.com/aakashnain/beating-everything-with-depthwise-convolution.