COVID-19 Transfer Learning Using Lung Images

Jacob Wirgård Wiklund **Fredrik Segerhammar** **Boesinger Leopaul**

Abstract

With the emergence of COVID-19, several teams of researchers are developing models to detect viral presence in the lungs of patients. However, there are still few X-ray images available. We employ transfer learning, using several pretrained models to determine which model is most suitable as well as preprocess these images to determine how accuracy can be improved on the currently available small datasets. Using Grad-CAM, we verified which parts of the images were taken into account for the classification. We found that VGG16 was the best model for this problem out of the ones we used, and that we were able to improve the accuracy by preprocessing the images. After analyzing the images with Grad-CAM we determined that any model trained solely on this dataset is not suitable for medical applications.

1 Introduction

The effect that COVID-19 has had on developed nations has so far been serious. This is mainly due to its infectiousness, and as such, being able to quickly and easily detect the presence of the virus in the lungs of patients is vital. Even more so when extreme viral presence is highly dangerous. A few years ago, a dataset (NIH [2017]) containing around 112,000 X-ray images of patients suffering from several different diseases was published by the NIH, and recently a relatively small dataset (Cohen [2020]) containing X-ray images of the lungs of COVID-19 infected patients has been published. Because of the scarcity of COVID-19 images, we decided to use transfer learning on pretrained models. We compared the accuracy of several models. We then analyzed what the models were taking into account, using Grad-CAM (Ramprasaath R. Selvaraju [2016]), a Class Activation Mapping and visualization method that can help you see which part of an image was the most important in the decision done by the classifier. and conducted some experiments regarding how to improve results by preprocessing the images. We had an initial plan to train our own model on the NIH data, however we were unable to do this due to the Google Cloud Platform being out of resources. Instead we utilized a pretrained model (Goren [2019]) on the NIH data to perform transfer learning, achieving poor results.

2 Related Work

In recent months there has been a lot of research looking into the possibility to detect COVID-19 in the lungs of infected patients. Previously, several teams had already applied machine learning models to X-ray images for the classification of several other lung diseases. Abiyev and Ma'aiatah published a CNN in 2018 that achieved 92% accuracy on the NIH dataset we also acquired in this study (Rahib H. Abiyev [2018]).

Several authors have shown that it is possible to achieve a very high accuracy even for X-ray images of COVID-19. For example Basu et al showed that you can achieve an accuracy of above 95% fairly easily with transfer learning (Sanhita Basu [2020]), using the images from the same NIH dataset we acquired. However, little has been done in terms of investigating how watermarks affect the images and what the effect of mixing these datasets is.

3 Data

The data that we are using consists of chest X-ray pictures, some of them being images of patients that are ill with the COVID-19, and the rest of them being X-ray images of patients where no disease was discovered.

The first dataset (Cohen [2020]), of COVID-19 images, is a public dataset using images collected from public sources as well as from an indirect collection from hospitals and physicians by Joseph Paul Cohen. We chose it because it was the first available dataset with a significant size (86 images taken from the same point of view).

The second dataset (NIH [2017]) comes from the United State's National Institutes of Health . We decided to use this dataset because it contains a lot (112,000) of chest images of different possible findings. This meant that we could compare COVID-19 chest X-rays with X-rays of chests with no finding, but also pneumonia, and other illnesses. For training, since we wanted to keep a balanced dataset, we only sampled as many images in this dataset as we had in the first dataset.

We only kept images with posteroanterior view, as they were the images that were the most common and showing the best viewing of the lungs. We applied basic data augmentation techniques, such as random rotations, some shearing and some zooming, because the images came from different places and were thus taken from slightly different points of view, but we didn't apply any elaborate data augmentation. We however applied data pre-processing, corresponding to the models we used (Karen Simonyan [2014]), and in later parts, to enhance the robustness of our classifier. When loading and looking through the COVID-19 dataset, we saw that there was a lot of artifacts, such as symbols and letters in the pictures, so we decided to experiment on the impact these symbols have on the robustness of our models.

4 Methods

It is a hard task to distinguish COVID-19 X-ray images from regular lung diseases such as pneumonia with limited amount of data. The dataset containing images of COVID-19 is relatively small with 86 images but luckily we have access to the big NIH dataset X-rays of chests from Kaggle which enables us to use modern deep learning techniques to work with limited amount of data. Our initial plan was to work with transfer learning and try to use a pretrained convolutional neural network (CNN) with the NIH dataset and fine-tune the last layers using the COVID-19 data. This is a good approach because it requires less labeled samples from each class to train the model. However, due to the limitations of using a pretrained network we also decided that we would try to train our own CNN to compare the result with the pretrained network. This enables us to do more experiments during the training process of the NIH dataset and get a better understanding of the pretrained networks' impact in the transfer learning process. In general, the metrics used for the performance of binary classification tests are known as sensitivity (true positive rate) and specificity (true negative rate). For the most part, we tried sticking to these measures, and providing confusion matrices, however in some cases, we reverted to only comparing models on the general accuracy (rate of correctly classified samples), and given that our dataset has been selected to be balanced, and is very small, we figured it was easier to compare results using a single measure, rather than a convoluted comparison of four measures.

4.1 Comparison between transfer learning from ImageNet weights, NIH weights, and no transfer learning

In order to be able to measure how useful transfer learning is for our dataset, we wanted to try three kinds of models: one that does transfer learning using weights that were pretrained on the ImageNet dataset, one that does transfer learning using weights that were pretrained on the NIH chest X-ray dataset, and one that doesn't do any transfer learning. This would allow us to determine what accuracy one can get with a very simple model, and to then determine if ImageNet can be used for medical transfer learning, or if we should generally focus on only doing transfer learning from models that were already trained on medical data.

When we were training our network without any transfer learning, we remarked that it is possible to get very good validation accuracies using very simple models. An example is a network that simply flattens the images, and then feeds it into a fully connected layer of 64 nodes, and then the last layer which determines if an image is coming from the COVID-19 or the no-finding dataset. This architecture (shown in Figure [1]) was able to get a very decent accuracy (see Figures [3, 4]).

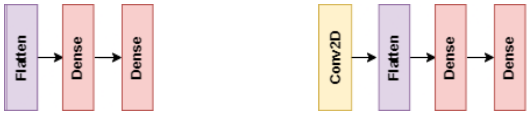


Figure 1: Architecture of the models without transfer learning

Another very simple architecture, is a convolutional layer, followed by the same architecture as before. Its accuracy was very close to the one described before.

Then, for the ImageNet pretrained model, we used a pretrained VGG16, from which we only kept the convolutional layers, and added fully-connected layers to produce a binary classifier (see Figure [2])

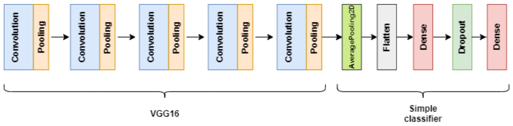


Figure 2: Architecture used with the VGG16 model with ImageNet weights

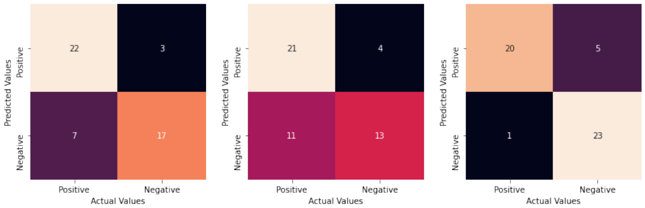


Figure 3: Confusion matrices obtained on basic Fully-connected model, basic CNN model, and VGG ImageNet model (in that order)

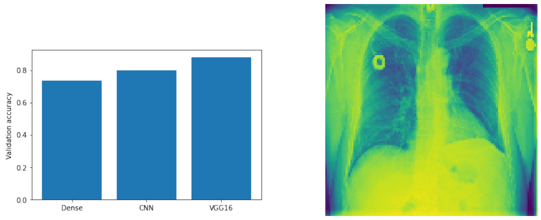
From Figure [3], we see that our models trained without any weight initialization suffer the most in their false-negative rates, whereas the ImageNet pretrained model suffers the most in the false-positive rates. In a test perspective this means that the ImageNet model is better regardless of the overall accuracy, because we would rather have a high false-positive rate than a high false-negative rate.

In order to have a better idea of the accuracy, and because of the size of the COVID-19 dataset, we decided to also evaluate these models on a dataset made of 200 images with no-findings, because we have no issue getting more images of lungs with no findings. A comparison of the accuracy is shown in Figure [4]

Afterwards, we wanted to try the experiment using an NIH-pretrained model. We had initially planned on training our own model, but we, like other groups, had issues with Google Cloud Platform not having enough resources to fulfill our request. Given that the NIH dataset is made of 112,000 images, we were not looking forward to training our model using our own machines, so we reverted to an existing kernel we found on Kaggle (Goren [2019]) (where the dataset was posted).

When loading the model, it was clear that it was not as efficient as we had hoped, only achieving random-like accuracy on the NIH dataset by itself. We thus performed a Grad-CAM with some pictures in the dataset to see if there was any particular reason why this was happening and to also discriminate between the convolutional and fully-connected layers, which were to blame. One of these Grad-CAMs is shown in Figure [4]. We see that all parts of the image are used, mostly evenly, in the classification, with the lungs less used than the other parts. We were not very pleased with this result, as it meant that there was probably already an issue with the convolutional layer.

We however, still tried to use this model for transfer learning, both with the complete network frozen, with, of course, a different fully-connected layer at the end, and with only the convolutional part kept, but in both cases, the results were even worse than guessing at chance.



(a) Accuracy on the no-finding dataset

(b) Grad-CAM obtained from the NIH pretrained model

Figure 4

4.2 Trying out different ImageNet pretrained models

We decided to further try out different well known CNN architectures that achieved promising results in recent years (ResNet50 [Kaiming He [2015]], MobileNetV2 [Mark Sandler [2018]]) on the ImageNet dataset to see which ones perform the best when using them for transfer learning. We were expecting to be able to simply exchange the VGG16 model with these and to achieve similar results, but we actually received more interesting results. The data needed different preprocessing, but that was fine since Keras has built-in preprocessing functions for all the applications that it offers.

However, when training the models, we remarked that the MobileNet and ResNet were very highly overfitting. The training accuracy reached 100% in a few epochs, but the validation accuracy was not as good. With MobileNetV2, we were still able to achieve decently good accuracy, but still lower than with the VGGNet. However, for the ResNet, the validation accuracy was not better than our most basic network. We saw that multiple other people had issues with ResNet overfitting, and that it was due to the batch normalization, where freezing layers makes it unable to perform well. Unfortunately, even when unfreezing these layers, we were not able to get a very good accuracy with the ResNet. It could have been an issue on our end, but from this experiment, we concluded that the VGGNet was both easier to set up and better performing than the two others, which is why we kept it. Figure [5]'s results follow the same trend as before, the overall accuracy is simply lower than for the VGGNet.

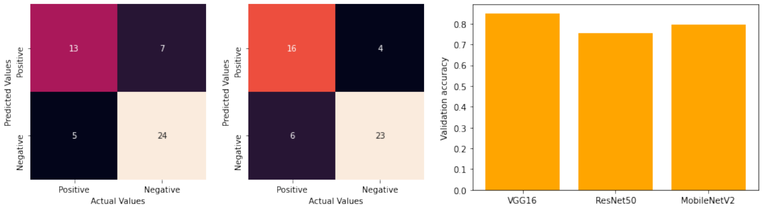
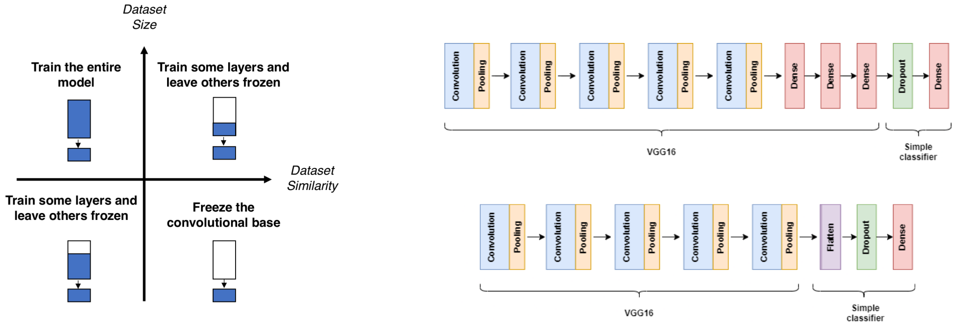


Figure 5: Validation accuracy and confusion matrices for the ResNet and MobileNet models (respectively)

4.3 Transfer learning testing

We know from empirical results that can be summed up in Figure [6], that different techniques can be used depending on the size of the dataset for which we are retraining our model and the data similarity, in order to achieve a better accuracy. Hence why we decided to experiment on the accuracy we are able to obtain when freezing different layers for our ImageNet-pretrained model.We did three experiments : the first one was freezing all the layers, and keeping a selection of fully-connected layers. The second one was freezing only the convolutional layers, and keeping a selection of fully-connected layers. The third one was removing all the fully-connected layers, and training a selection of convolutional layers. Something to note is that we always removed layers starting from the end, so having 2 fully-connected layers meant that we only removed the last fully-connected layer for example.



(a) Empirical results in transfer learning (Marcelino [2018])

(b) Our simple testing architecture, strapped on a VGG16 model, for the first and second experiments (TOP) and third experiment (BOTTOM)

Figure 6

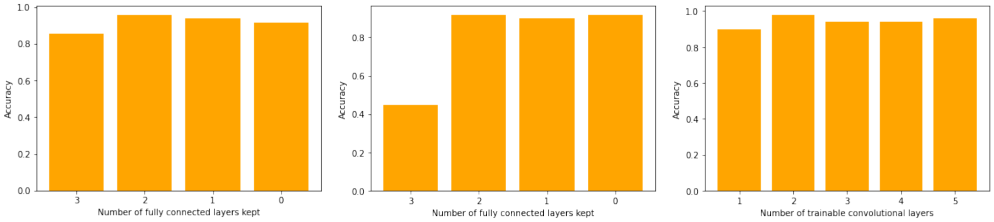


Figure 7: Validation accuracy on the first, second and third experiments (in the same order) on the ImageNet pretrained model

What is interesting about Figure [7], is that we see that in general here, freezing the weights of the model doesn't affect the accuracy very much. Furthermore, in these little differences there doesn't seem to be a general trend in the number of layers to freeze or train (this is likely due to the fact that this classification is a very easy problem to solve). The only outlier we see, is when keeping all three convolutional layers and training them, the validation accuracy is really low. We could have expected this to be caused by the fact that on our VGG16, the third fully-connected layer would be the one that would distinguish between the 1000 classes ImageNet offers, but since we have removed the softmax activation from it, and the first experiment does not show the same trend, we cannot really comment further on the reasons for this.

The results we would expect here is that since the NIH data is more similar to the COVID-19 data compared to the ImageNet data, we would be in the down-right quadrant of Figure [6], able to only freeze the convolutional base, and build on top of that, compared to the ImageNet trained model, where we would need to train more layers in order to get a better accuracy.

Unfortunately, for the same reasons expressed as before, our NIH pretrained model was clearly not performing well, so there was no use in trying different configurations when the baseline was wrong.

5 Experiments

5.1 Grad-CAM verification

After training our first classifier, we wanted to verify that our model used the right parts of the image to determine if an image came from a COVID-19 patient or a patient with no finding. This issue is especially common in transfer learning because the images used for the first learning come from a different source as the ones used in the second learning, and in the prediction. This is the case here, where we had to merge two datasets, the one with COVID-19 images, and the NIH dataset from which we took X-rays with no findings. In that intent, since we knew that convolutional networks are really hard to interpret, we looked at Grad-CAM (Ramprasaath R. Selvaraju [2016]). We used the implementation done in [Chakraborty [2020]]. We remarked that the COVID-19 dataset suffers from one particular issue: it features many images where there are letters, and symbols, both in the corners of the images (for the letters), and in the middle (for the symbols). From the Grad-CAM images, we could see that the most important features were the letters, as can be seen in Figure [8] in the middle Grad-CAM, obtained without image cropping. We found out that cropping the images was enough to fix the issue, and the models were actually achieving a better accuracy after this change. However, this didn't fix all issues, as there were still symbols and letters left that weren't placed in the corners. Since for all these, the color used is white, we also tried preprocessing images, by filtering parts with very high brightness (these white symbols), and inpainting them (OpenCV). The result can be seen in Figure [9], where we see that the Grad-CAM no longer featured any zone other than the lungs, which is what we want here.

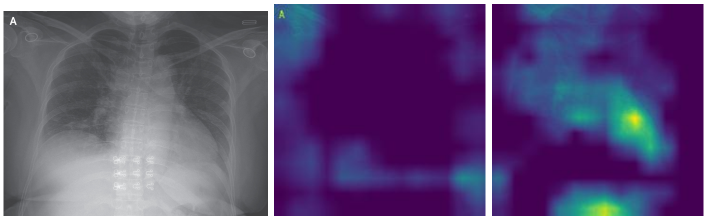


Figure 8: The effect of cropping on our classifier

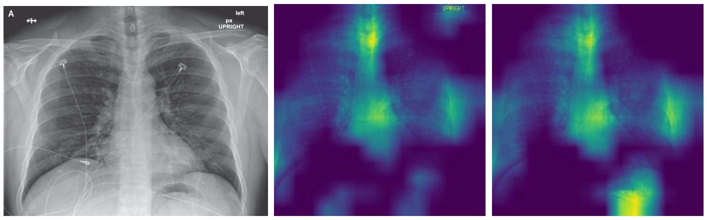


Figure 9: The effect of text inpainting on our classifier, left gradCAM without inpainting and right gradCAM with inpainting

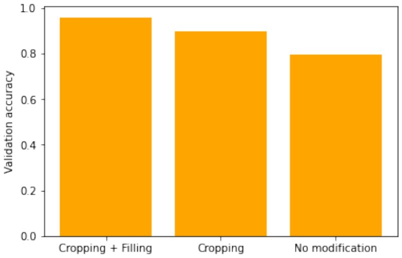


Figure 10: Validation accuracies obtained with no modification on images, with cropping and with cropping and filling up white symbols

The accuracies in [10] are not very far apart, but it comforts us in the sense that we wanted to keep only the important parts of the images for the model to not get confused. Of course, these Grad-CAMs are not at all a guarantee that the features used by our model completely correspond to the ones that a COVID-infected lung would have, or that they are exclusive to COVID-19 and not common with pneumonia or other lung illnesses, but these issues are not really ones we can fix ourselves, given that we are not medical professionals. Our very basic networks were also still performing very well on these pre-processed datasets, hence why tried another technique.

5.2 Adding symbols to the non-COVID dataset

A known issue for the the training process is the watermarks that are present in the COVID-19 dataset, but not in the corners. This is especially a problem due to the fact that it can be used during the classification to distinguish between COVID-19 and non-COVID images. To handle this problem we performed a number of experiments, one of them was to extract watermarks from the COVID-19 dataset and to add them to the no-finding dataset, see Figure [11]. This was done by adding a threshold filter to the COVID-19 images that extracted the watermarks and then applying them to the non-COVID images by using contour extraction. The idea of this experiment was to help the training process by adding the watermarks to both the testing and training data, thereby making them less crucial for the classification.

The problem with this experiment was that the threshold unfortunately extracted other parts from the image which resulted in distorted images in the non-COVID dataset, which had a negative effect on the accuracy, see figure [11] right picture. The modified dataset containing watermarks was tested on both a basic training model using just a fully connected layer of 64 nodes and a more advanced model using a VGG16 model pretrained on ImageNet. The basic and the more advanced model both achieved a validation accuracy of around 50% which is insufficient compared to the other experiments.

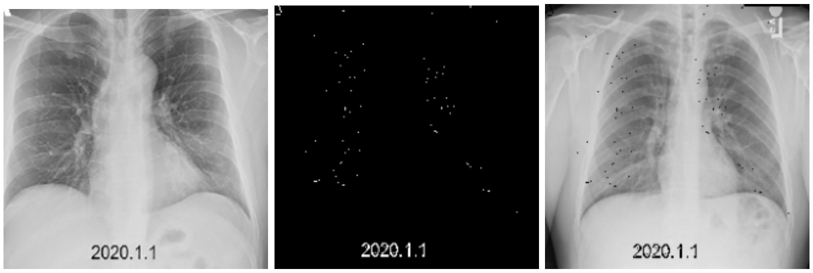


Figure 11: Example of the extracting process

After analysing the model with the help of Grad-CAM we could see that it worked for some images but not for all, see figure [12] left picture and middle picture. Another problem that was discovered during this experiment was the impact that bones and other features had on the classification. In many cases the classification seemed to be based on difference in bone structure and other specific features like a woman's breast, see figure [12] right picture. Indeed we saw that, while the Grad-CAMs done on COVID-19 images were showing mostly zones corresponding to the lungs, a good portion of Grad-CAMs performed on the No-Finding dataset showed a bias towards other zones such as the bone structure. This is a major flaw and could be a factor for the over fitting and other problems that were discovered. However, it's not certain that the model uses solely the bones for classification because of the fact that lungs are black by default. The way Grad-CAM computes the image, is by multiplying the image values by a mask of high importance zones. This multiplication means that dark (low value) zones, will of course appear closer to blue than yellow. However, even when changing the multiplication by a superposition with transparency, or by simply looking at the mask by itself, we noticed that the lungs were indeed less used in the classification.

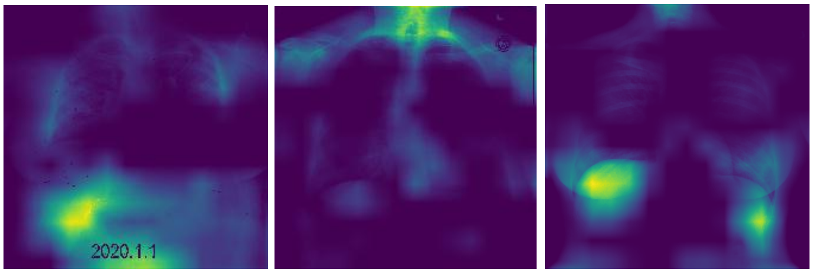


Figure 12: Results from Grad-CAM analysis

6 Conclusion

We learned that transfer learning is generally a very easy technique to perform when working with convolutional networks. We saw that even for a medical dataset, using an ImageNet pretrained network worked very well, achieving accuracy very close to the one we found in recent papers. We first learned that the COVID-19 dataset contained a lot of watermarks and symbols, and that even with data augmentation, a very basic network would be able to achieve decent accuracy. Among more complex networks, we learned that VGG16 performed the best. However, we were not able to make any particular claim on which layers to train and freeze, but we saw on the other hand that it is not always worth trying out which layers to freeze, especially on datasets where shallower networks seem to perform well. It was an interesting experiment to add symbols to the non-COVID dataset but unfortunately it did not yield any good results. However, all these results need to be taken with caution because we saw by using Grad-CAM, that even with preprocessing, the network was not always using lungs as the discriminator during classification. As future work, it would be very enriching to be able to continue these experiments using a model trained on a medical dataset (NIH), comparing results with the ones obtained on the ImageNet pretrained model, along with using different data augmentation methods.

References

United States' NIH. Nih chest x-rays dataset, 2017. URL https://www.kaggle.com/nih-chest-xrays/data.

Joseph Paul Cohen. Covid chest x-ray dataset, 2020. URL https://github.com/ieee8023/covid-chestxray-dataset.

Abhishek Das-Ramakrishna Vedantam-Devi Parikh Dhruv Batra Ramprasaath R. Selvaraju, Michael Cogswell. Grad-cam: Visual explanations from deep networks via gradient-based localization, 2016. URL https://arxiv.org/abs/1610.02391.

Adam Goren. Nih chest x-ray multi-classification, 2019. URL https://www.kaggle.com/adamjgoren/nih-chest-x-ray-multi-classification/output?fbclid=IwAR3AxO-BdMdi32XisBhYVnl7xLfnpJOsX1mZnF9S_4XnhUBLxyxzf2fe5EO.

Mohammad Khaleel Sallam Ma'aitah Rahib H. Abiyev. Deep convolutional neural networks for chest diseases detection, 2018. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6093039/.

Nilanjan Saha Sanhita Basu, Sushmita Mitra. Deep learning for screening covid-19 using chest x-ray images, 2020. URL https://arxiv.org/pdf/2004.10507.pdf.

Andrew Zisserman Karen Simonyan. Very deep convolutional networks for large-scale image recognition, 2014. URL https://arxiv.org/abs/1409.1556.

Shaoqing Ren Jian Sun Kaiming He, Xiangyu Zhang. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

Menglong Zhu Andrey Zhmoginov Liang-Chieh Chen Mark Sandler, Andrew Howard. Mobilenetv2: Inverted residuals and linear bottlenecks, 2018. URL https://arxiv.org/abs/1801.04381.

Pedro Marcelino. Transfer learning from pre-trained models. *Towards Data Science*, 2018. URL https://towardsdatascience.com/transfer-learning-from-pre-trained-models-f2393f124751.

Souradip Chakraborty. An attempt- detection of covid-19 presence from chest x-ray scans using cnn and class activation maps. *Towards Data Science*, 2020. URL https://towardsdatascience.com/detection-of-covid-19-presence-from-chest-x-ray-scans-using-cnn-class-activation-maps-c1ab0d7c294b.

OpenCV. Inpainting documentation. URL https://docs.opencv.org/2.4/modules/photo/doc/inpainting.html.

Minaee Shervin, Kafieh Rahele, Sonka Milan, Yazdani Shakib, and Soufi Ghazaleh. Deep-covid: Predicting covid-19 from chest x-ray images using deep transfer learning, 2020.

Richard Socher Li-Jia Li-Kai Li Jia Deng, Wei Dong and Li Fei-Fei. Imagenet: A large-scale hierarchical image database, 2009. URL http://www.image-net.org/papers/imagenet_cvpr09.pdf.