# COVID-19 Detection from Chest X-ray Images: Image Enhancement, Segmentation and CNN Feature Fusion

**Austėja Vaitkutė**      **Guostė Kavalnytė**
**Gabrielė Ruminavičiūtė**      **Mindaugas Stagys**

December 6, 2021

### Abstract

The 2019 novel infectious disease named COVID-19 has emerged to become a pandemic that is rapidly spreading all over the world. Early detection of this disease, which has caused serious effect on the healthcare system and global economy, is of vital importance. The chest radiography imaging (CXR) has proved to be an effective screening technique in diagnosing the COVID-19. This study is aimed to automatically detect COVID-19 disease from CXR images while maximizing the accuracy in detection. The proposed framework involves image enhancement technique called Contrast Limited Adaptive Histogram Equalization (CLAHE), image segmentation using U-Net, feature extraction from pre-trained convolutional neural networks (CNNs), namely VGG19 and ResNet-50 architectures, UMAP for dimensionality reduction, and classification with SVM or XGBoost algorithms. Our dataset consists of three classes: COVID-19, viral pneumonia and normal CXR imagery. To establish the robustness of the proposed model, stratified 5-fold cross-validation is carried out. The results suggest that image enhancement, segmentation and future fusion methods used in this paper could indeed improve classification results. The proposed framework achieved a 94.96% balanced accuracy rate, 94.96% precision, 89.31% recall, 96.01% specificity and 91.83% F1-score.

## 1 Literature review

There have been several studies trying to develop automatic COVID-19 classification systems, mainly using Convolutional Neural Networks (CNNs) trained on an ImageNet [7] dataset and machine learning classifiers. Dias Junior et al. [8] compared various classifiers applied on deep features extracted from pre-trained CNNs (VGG19, Inception-v3 and ResNet50). Authors showed that the results achieved with PSO-optimized XGBoost are superior to those obtained using other classifiers. Besides, VGG19 performed better than other CNN architectures. Saha et al. [30] trained a simple CNN network for extracting deep and high-level features from CXR images. With the extracted features, an ensemble of binary machine learning classifiers such as random forest, support vector machine, decision tree and AdaBoost was applied for the detection of COVID-19. It was shown that ensemble learning could indeed obtain better predictive performance.

Several studies showed that classification results might be improved with image segmentation and enhancement at a preprocessing stage. Toğaçar et al. [32] restructured the original dataset with the fuzzy color technique and then overlayed structured images over original ones to help eliminate the noises. Rahman et al. [26] compared five different image enhancement techniques. A modified U-Net model was proposed and compared with the standard U-Net model for lung segmentation. Six different pre-trained CNN models and a shallow CNN model were investigated on the plain and segmented lung CXR images. ChexNet with modified U-Net and gamma correction-based enhancement technique outperformed other methods in detecting COVID-19. Lung segmentation using U-Net was also proposed by Arias-Garzón et al. [2].

Authors compared VGG16 and VGG19 network performance. On a test set the better results were achieved using VGG19. The segmentation task showed a high probability of providing extra information.

Feature extraction and selection from the dataset is one of the essential steps for classification tasks. In most of the recent articles, features are extracted by neural networks. Some authors suggested alternative methods for future extraction. Nabizadeh et al. [20] proposed a bag of visual words approach with a Speeded-Up Robust Features (SURF) algorithm. Authors achieved high accuracy and specificity, however, since SURF can be susceptible to noise, it might misclassify low quality images. Rajpal et al. [27] considered a classification framework which combines a set of handpicked features with those obtained from a CNN. Authors constructed a pool of frequency and texture based features, then used PCA for dimensionality reduction, concatenated these features with the ones extracted from ResNet-50 and finally passed them to a feed forward neural network. It was shown that handpicked features can enhance the learning ability of the transfer learning model. Toğaçar et al. [32] were able to produce faster and more accurate results by applying Social Mimic Optimization (SMO) for future selection. Koyuncu and Barstuğan [17] compared combinations of four types of radiomic features: first-order statistics (FOS), gray level co-occurrence matrix (GLCM), gray level run length matrix (GLRLM) and gray level size zone matrix (GLSZM). Authors conclude that FOS features are necessary for achieving the best performance. Best results were achieved using a combination of FOS and GLRLM features, minmax as the normalization approach, Bhattacharyya distance or Receiver Operating Characteristic Curve (ROC) rankings as the feature selection method, and Gauss-map-based chaotic particle swarm optimization – Neural network (GM-CPSO–NN) as the classifier unit.

Some authors considered task-specific CNN model improvements. Ouchicha et al. [23] discussed the impact of the size of CNN filters for ground-glass opacity detection. Ground-glass opacities are considered as the typical early features of COVID-19 infection found in CXR images. These features may appear in different sizes, shapes, quantities and locations. A small filter may not capture holistic information of the large sized ground-glass opacities features. Meanwhile, large filter might result in the loss of details and poor classification accuracy. Authors trained a model called CVDNet based on the residual neural network. Proposed method is constructed by using two parallel convolution levels with different filter sizes to capture local and global features of the inputs. Fan et al. [10] suggested spatial attention and channel attention modules which would learn to focus on the pathological features of the CXR images and suppress the shadow and skeletal noise features.

Despite an excellent performance in computer vision, CNNs also have some drawbacks. Since CNNs have a large number of parameters, they require large datasets for training purposes. Furthermore, CNNs are not scale or rotation invariant. Afshar et al. [1] addressed these shortcomings with a capsule network called COVID-CAPS. This model was able to achieve satisfying performance with a much lower number of trainable parameters than traditional CNNs. Authors also showed that pre-training could further improve the accuracy and AUC. Quan et al. [24] extended capsule network by a fusion with DenseNet. Authors demonstrated that on smaller datasets performance of the proposed framework is indeed better than that of using CNN alone. Model significantly improved the recall compared to [1] but was much higher in the number of network trainable parameters. Dixit et al. [9] suggested a highly accurate approach without the use of CNNs. First, data preprocessing step involved image segmentation using k-means clustering. Then features were selected using particle swarm optimization (PSO) with differential evolution (DE) algorithm for local search to improve the efficiency and avoid local optima trap. Finally, optimized features were classified with SVM.

Table 1: Comparison of related works on COVID-19 detection from chest x-ray images

| Study | Dataset | Proposed model | Acc (%) | Prec (%) | Rec (%) | Spec (%) | F1 (%) |
|---|---|---|---|---|---|---|---|
| Afshar et al. [1] | 266 COVID-19<br>13604 non-COVID | COVID-CAPS | 98.30 | – | 80.00 | 98.60 | – |
| Arias-Garzón et al. [2] | 3240 COVID-19<br>5469 normal | VGG19 + U-Net | 96.90 | – | 98.10 | 94.60 | 97.90 |
| Dias Junior et al. [8] | 206 COVID-19<br>1341 normal | VGG19 +<br>XGBoost + PSO | 98.71 | 98.89 | 99.63 | – | 99.25 |
| Dixit et al. [9] | 358 COVID-19<br>8066 normal<br>5551 pneumonia | k-means +<br>Differential<br>evolution +<br>PSO + SVM | 99.35 | 100.00 | 99.30 | – | 99.30 |
| Fan et al. [10] | 500 COVID-19<br>500 normal | Multi-Kernel-Size<br>Spatial-Channel<br>Attention Network | 98.20 | 98.10 | 98.10 | 98.30 | 98.10 |
| Koyuncu and Barstuğan [17] | 80 COVID-19<br>160 normal<br>160 bacterial<br>pneumonia | Gauss-map-based<br>chaotic particle<br>swarm optimization<br>neural network | 99.25 | 96.39 | 100.00 | 99.06 | 98.16 |
| Nabizadeh et al. [20] | 1510 COVID-19<br>20723 normal | Bag of Visual<br>Words + SVM | 99.84 | – | 99.65 | 99.95 | – |
| Ouchicha et al. [23] | 219 COVID-19<br>1341 normal<br>1345 viral<br>pneumonia | CVDNet<br>(modified ResNet) | 96.69 | 96.72 | 96.84 | – | 96.68 |
| Quan et al. [24] | 250 COVID-19<br>250 normal<br>250 pneumonia | DenseNet-121 +<br>CapsNet | 90.70 | 91.10 | 90.70 | 95.30 | 90.90 |
| Rahman et al. [26] | 3616 COVID-19<br>8851 normal<br>6012 other<br>lung infections | ChexNet +<br>modified U-Net +<br>gamma enhancement | 96.29 | 96.28 | 96.29 | 97.27 | 96.28 |
| Rajpal et al. [27] | 520 COVID-19<br>520 normal<br>520 pneumonia | ResNet-50 +<br>handpicked<br>features +<br>PCA + FNN | 97.40 | 96.03 | 96.90 | – | 96.90 |
| Saha et al. [30] | 2300 COVID-19<br>2300 normal | CNN +<br>ML classifiers | 98.91 | 100.00 | 97.82 | – | 98.89 |
| Toğaçar et al. [32] | 295 COVID-19<br>65 normal<br>98 pneumonia | fuzzy color +<br>stacking + SMO +<br>MobileNetV2 +<br>SqueezeNet + SVM | 99.27 | 99.27 | 99.30 | 99.80 | 99.27 |

## 2 Materials and methods

The overall structure of the proposed framework has 5 stages: image preprocessing, segmentation, feature extraction, dimensionality reduction and classification. First, we perform image augmentation using CLAHE image enhancement technique. Then we propose a modified U-Net architecture for lung segmentation. Once U-Net is trained and applied on classification dataset, we perform feature extraction using pre-trained CNNs. Extracted features are then transformed and dimensionality is reduced with a UMAP algorithm. Finally, several machine learning classifiers are compared.
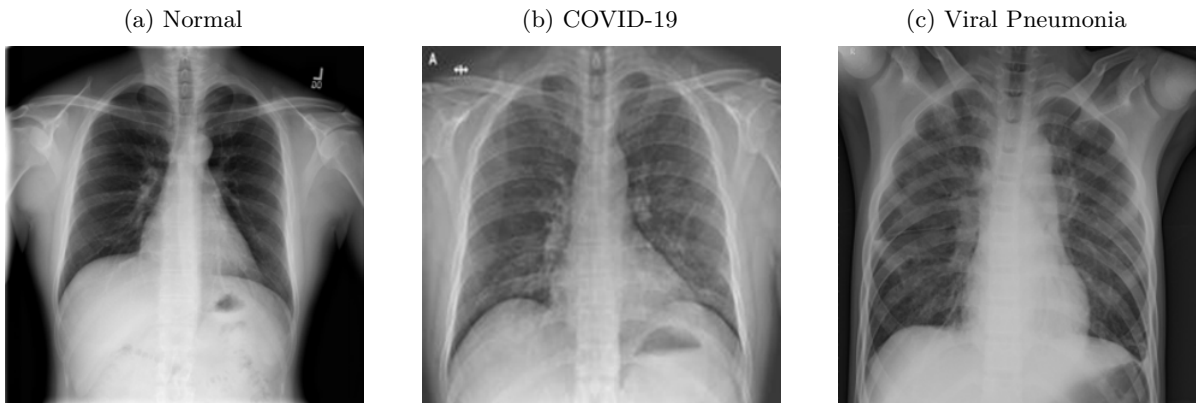
## 2.1 Datasets

### 2.1.1 Lung segmentation

Candemir et al. [3], [14] manually generated gold standard lung boundary segmentations for two CXR datasets under the supervision of a radiologist. The first set was collected by the tuberculosis control program of Department of Health and Human Services of Montgomery County, MD, USA. This set was comprised of 138 posterior-anterior x-rays, of which 80 x-rays were normal and 58 x-rays were abnormal with manifestations of tuberculosis. The second set was collected by Shenzhen No. 3 Hospital in Shenzhen, Guangdong providence, China. Set contains 326 normal x-rays and 336 abnormal x-rays showing various manifestations of tuberculosis. We used the same 800 images and masks to train the segmentation network. In this work, 80% of images were randomly chosen for training and the remaining 20% were selected as the validation set.

### 2.1.2 Image classification

Classification task was performed on CXR images obtained from Kaggle's COVID-19 Radiography Database [25]. Dataset is comprised of three different groups of CXR images: patients infected with COVID-19, viral pneumonia and images of healthy individuals. 3615 images classified as COVID-19 were compiled using 6 databases: 2474 images are collected from Valencia Region Image Bank (BIMCV) padchest dataset [6], 183 images from a Germany medical school [33], 400 images compiled by Haghanifar et al. [12], 182 images compiled by Cohen et al. [5], 258 images from Eurorad and 119 images provided by Italian Society of Medical and Interventional Radiology (SIRM) [21]. 8851 normal CXR images were obtained from RSNA Pneumonia Detection Challenge [22]. Finally, 1341 normal images as well as 1345 viral pneumonia images were obtained from a database (version 2) compiled by Kermany et al. [16]. A random selection of 80% of images in each category was used for training and stratified five-fold cross-validation, and the remaining 20% was used for testing.

Figure 1: A sample of CXR images

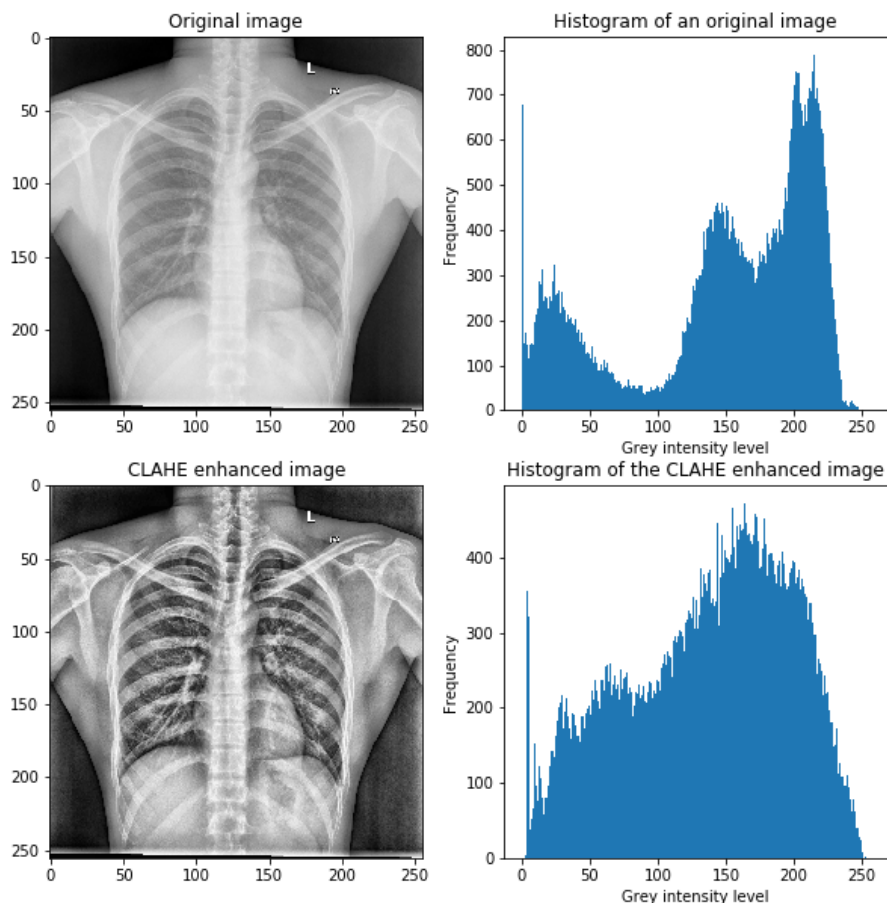(a) Normal  (b) COVID-19  (c) Viral Pneumonia



## 2.2 Contrast limited adaptive histogram equalization (CLAHE)

The histogram equalization (HE) is a computer image processing technique used to improve contrast in images. The method is useful in images with backgrounds and foregrounds that are both bright or both dark. For instance, the method can lead to better views of bone structure in x-ray images. HE, however, also produces some undesirable effects such as oversaturation of certain regions and noise artifacts. An improved HE variant is called adaptive histogram equalization (AHE). It differs from ordinary histogram equalization in the respect that the adaptive method computes several histograms, each corresponding to a distinct section of the

image. It is therefore suitable for improving the local contrast and enhancing the definitions of edges in each region of an image. AHE, however, has a tendency to overamplify noise in relatively homogeneous regions of an image. A variant of adaptive histogram equalization called contrast limited adaptive histogram equalization (CLAHE) prevents excessive contrast enhancement by limiting the amplification with a clip boundary which represents the maximum height of a histogram. In this study, we set a threshold value for contrast limiting to 4.0. CLAHE is applied on the input image divided into equally sized rectangular tiles. We set the size of these tiles in row and column to (16, 16).
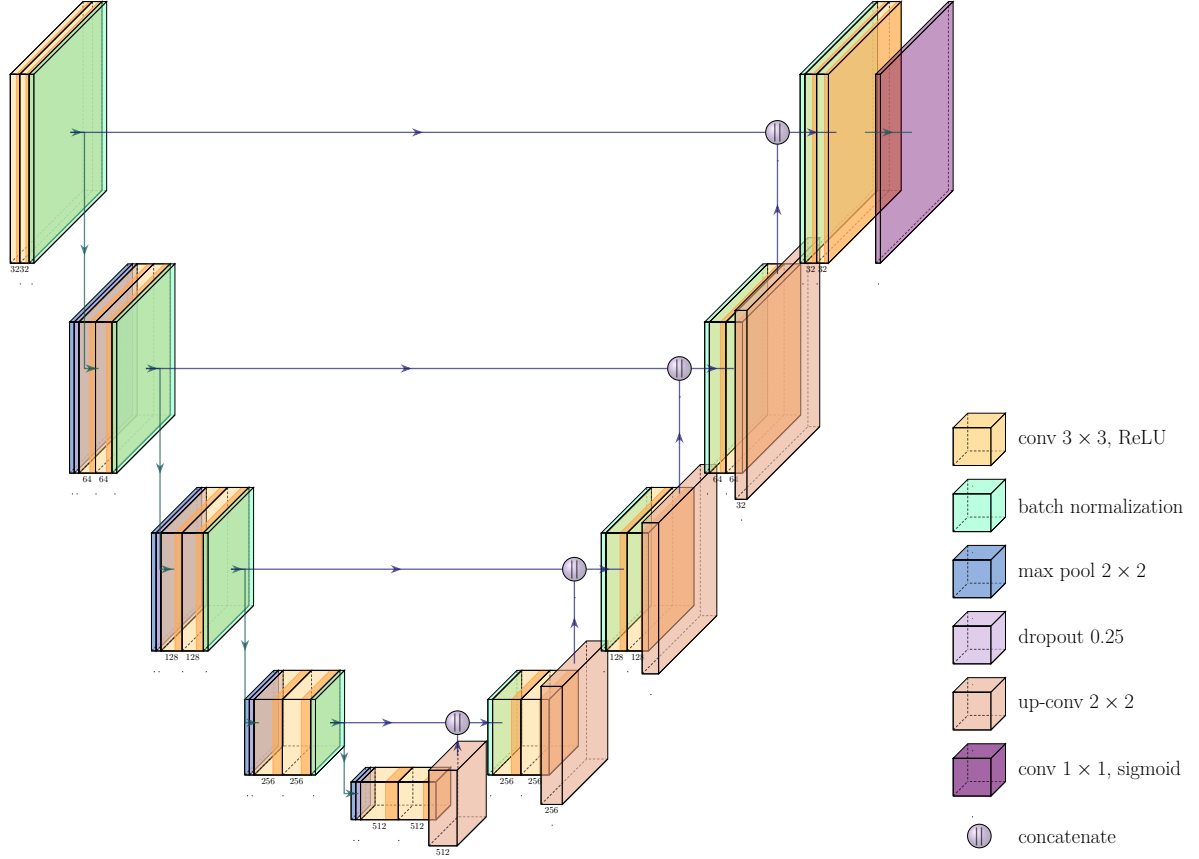
Figure 2: Image preprocessing effect



After the original CXR image is preprocessed with CLAHE, the gray distribution is more uniform, the lung contour is clearer and the contrast is more obvious, which can improve the segmentation and classification capability.

## 2.3  U-Net

U-Net is a convolutional neural network that was developed for biomedical image segmentation at the Computer Science Department of the University of Freiburg [29]. In the original network, each encoding block consists of two consecutive $3 \times 3$ convolutional layers each followed by a rectified linear unit (ReLU) and a max pooling operation with a stride of 2 for down sampling. In our modified U-net architecture two consecutive convolutional layers are followed by batch normalization, which should help stabilizing the learning process and reduce the number of training epochs. We have also added a dropout layer with a rate of 0.25 after the max pooling layer to reduce overfitting. The expansive pathway combines the feature and spatial information through a sequence of up-convolutions and concatenations with high-resolution features from the contracting path. Our proposed model additionally has batch normalization after concatenation

layer in each of the decoding blocks. At the final layer $1 \times 1$ convolution is utilized to map the output from last decoding block to 2 channel feature maps, where a pixel-wise sigmoid function is applied to map each pixel into a binary class of background or lung.

Figure 3: Modified U-Net architecture



## 2.4 Feature extraction using transfer learning

The concept of transfer learning has been introduced for solving deep learning problems arising from insufficiently labeled data, or when the CNN model is too deep and complex. Aiming to tackle these challenges, studies in a variety computer vision tasks demonstrated the advantages of transfer learning strategies from an auxiliary domain in improving the detection rate and performance of a classifier. In a transfer learning strategy, we transfer the weights already learned on a cross-domain dataset into the current deep learning task instead of training a model from scratch. With the transfer learning strategy, the deep CNN can obtain general features from the source dataset that cannot be learned due to the limited size of the dataset in the current task. Transfer learning strategies have various advantages, such as avoiding the overfitting issue when the number of training samples is limited, reducing the computational resources, and also speeding up the convergence of the network.

### 2.4.1 VGGNet

Visual Geometry Group (VGGNet) was proposed by Simonyan and Zisserman [31]. This architecture obtained top performances on ImageNet Large Scale Visual Recognition Challenge (ILSVRC) in 2014. This architecture provides better features extraction from input images by using $3 \times 3$ filter size. VGG16 and VGG19 are two versions of VGG-Net architecture with different depths and layers. In this study, VGG19 was chosen for feature extraction.

### 2.4.2 ResNet

Deep Residual Learning Network (ResNet) was proposed by He et al. [13]. This architecture won ILSVRC classification task in 2015 with good results on ImageNet and MS-COCO object detection competitions. ResNet introduced the concept of residual blocks. The main goal of residual blocks is to add a connection (instead of concatenation) from the input of the first block to the output of the next block in order to train a more deeper network with better recognition ability. This architecture can solve the issues of vanishing gradients and parameter explosion by shortcut connection using the residual blocks. In this study we have chosen a variant of ResNet called ResNet50V2.

## 2.5 Dimensionality reduction using UMAP

Uniform Manifold Approximation and Projection (UMAP) [19] is an algorithm for dimension reduction based on manifold learning techniques and ideas from topological data analysis. UMAP constructs a high dimensional graph representation of the data then optimizes a low-dimensional graph to be as structurally similar as possible. In order to construct the initial high-dimensional graph, UMAP builds something called a "fuzzy simplicial complex", which is a representation of a weighted graph, with edge weights representing the likelihood that two points are connected. To determine connectedness, UMAP extends a radius outwards from each point, connecting points when those radii overlap. Choosing this radius is critical - too small a choice will lead to small, isolated clusters, while too large a choice will connect everything together. UMAP overcomes this challenge by choosing a radius locally, based on the distance to each point's $n$th nearest neighbor. UMAP then makes the graph "fuzzy" by decreasing the likelihood of connection as the radius grows. Finally, by stipulating that each point must be connected to at least its closest neighbor, UMAP ensures that local structure is preserved in balance with global structure. Once the high-dimensional graph is constructed, UMAP optimizes the layout of a low dimensional embedding to be as similar as possible.

Basic parameters of UMAP include dimensionality of the reduced dimension space, number of neighbors, minimum distance and distance metric. Number of neighbors controls how UMAP balances local versus global structure in the data. It does this by constraining the size of the local neighborhood UMAP will look at when attempting to learn the manifold structure of the data. This means that low values of number of neighbors will force UMAP to concentrate on very local structure (potentially to the detriment of the big picture), while large values will push UMAP to look at larger neighborhoods of each point when estimating the manifold structure of the data, losing fine detail structure for the sake of getting the broader of the data. Minimum distance controls how tightly UMAP is allowed to pack points together. It provides the minimum distance apart that points are allowed to be in the low dimensional representation. This means that low values of minimum distance will result in clumpier embeddings and larger values will prevent UMAP from packing points together and will focus on the preservation of the broad topological structure instead. Distance metric controls how distance is computed in the ambient space of the input data. We have obtained 20-dimensional feature space using 100 neighbors, minimum distance of 0.1 and Manhattan distance, which are parameters obtained using stratified 5-fold cross-validation that was repeated 3 times.

## 2.6 Classifiers

## 2.7 Support Vector Machine (SVM)

The SVM algorithm is highly preferred owing to its significant accuracy while requiring less computational power. The objective of SVM is to find a hyperplane in an $N$-dimensional space that distinctly classifies the data points. Our dimension $N$ is equal to our number of features, which is 2560 (512 features extracted from VGG19 and 2048 features extracted from ResNet50V2). SVM has 3 main hyperparameters: *kernel*, *C* and *gamma*. The main function of

the *kernel* is to take low dimensional input space and transform it into a higher-dimensional space; $C$ is the penalty parameter, which represents misclassification or error term that tells the SVM optimisation how much error is bearable; *gamma* defines how far influences the calculation of plausible line of separation, when *gamma* is higher, nearby points will have high influence, whereas low gamma means far away points also be considered to get the decision boundary. In order to achieve a good result and at the same time to avoid overfitting 5-fold cross-validation was repeated 3 times and the following hyperparameters were tuned with: *kernel* = rbf (radial basis function kernel), $C = 1000$ and *gamma* = 0.0001.

## 2.8 XGBoost

Another classifier used in this work is XGBoost, owing to its strong performance in terms of speed, scalability and hardware resources compared to other existing solutions. XGBoost consists of the library proposed in [4] and is based on the gradient increase framework developed by Friedman [11]. XGBoost offers an efficient and scalable implementation of the gradient-based decision tree algorithm. Its main hyperparameters include maximum depth of the tree (*max_depth*), learning rate (*eta*), minimum loss reduction *gamma*, subsample ratio (*subsample*), subsample ratio of columns (*colsample_bytree*), minimum child weight (*min_child_weight*), L1 (*alpha*) and L2 (*lambda*) regularization terms. Increasing *max_depth* increases the complexity of the model, however, the propensity for overfitting is also increased; *eta* determines the size of the model's evolution step in each iteration; *gamma* denotes the minimum loss reduction required to form an additional partition in a leaf node of the tree, increasing this makes the algorithm more conservative; *subsample* denotes subsample ratio of the training instances, subsampling is useful to prevent overfitting; *colsample_bytree* indicates the column subsample rate when building each tree; *min_child_weight* denotes the minimum sum of the instance weights required in a child node, increasing its value makes the algorithm more conservative; Increasing *alpha* and *lambda* make model more conservative. Hyperparameters optimized with random search and cross-validation were as follows: *max_depth* = 8, *eta* = 0.045, *gamma* = 0 (default), *subsample* = 0.84, *colsample_bytree* = 0.73, *min_child_weight* = 2, *alpha* = 0.45 and *lambda* = 1.

## 2.9 Performance metrics

The different model's performance in classification was evaluated using six performance metrics: overall accuracy, balanced accuracy, recall (sensitivity), specificity, precision, and F1 score (Dice coefficient) using the following equations:

$$\text{Overall Accuracy} = \frac{TP + TN}{P + N} \qquad\qquad \text{Recall} = \frac{TN}{TP + FN}$$

$$\text{Balanced Accuracy} = \frac{1}{\sum \hat{w}_i} \sum_i \mathbb{1}_{\hat{y}_i = y_i} \hat{w}_i \qquad \text{Specificity} = \frac{TN}{TN + FP}$$

$$\text{Precision} = \frac{TP}{TP + FP} \qquad\qquad \text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\hat{w}_i = \frac{w_i}{\sum_j \mathbb{1}_{y_j = y_i} w_j}$$
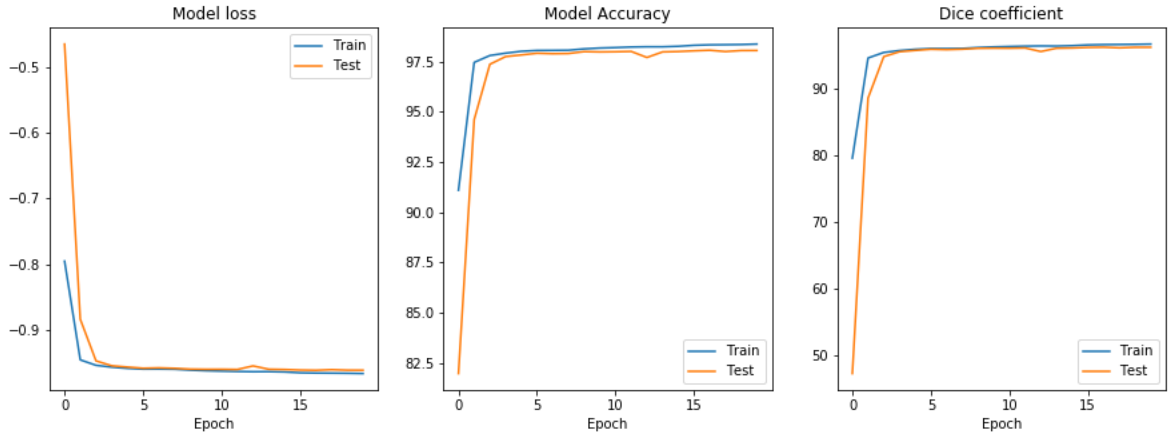
Here $y_i$ and $\hat{y}_i$ are, respectively, true and predicted values of $i$-th sample, $w_i$ is the corresponding sample weight and $\mathbb{1}_A$ is the indicator function. $TP$, $TN$, $FP$, $FN$ represent the true positive, true negative, false positive, and false negative, respectively. Since the dataset is imbalanced and it is important to accurately classify minority class (COVID-19 or pneumonia), macro-average method (arithmetic mean of individual scores) was chosen for aggregation of precision, recall, specificity and F1 score.

# 3 Experimental results
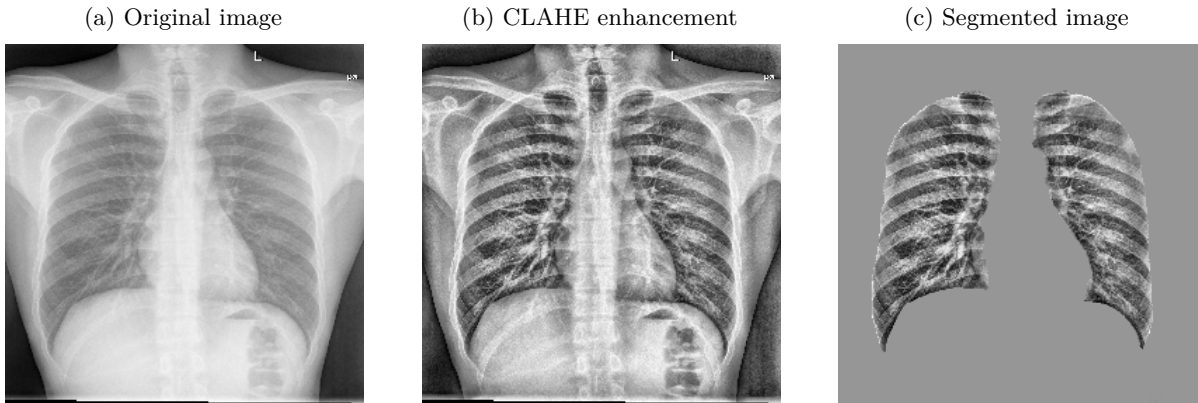
## 3.1 Segmentation performance

U-net model was trained for 20 epochs with a batch size 4 and a learning rate 0.0001, which would decay with a factor of 0.5 if validation loss did not improve for 3 epochs.

Figure 4: Learning curves



Just after 3 epochs training and validation loss decreased to a point of stability with a minimal generalization gap which indicates a good fit. We have obtained 98.05% accuracy and 96.24% Dice coefficient on the test set, which is a higher than several previous studies [3], [28], [15]. Figure 5 shows an example of successfully segmented lungs.

Figure 5: Lung segmentation



(a) Original image    (b) CLAHE enhancement    (c) Segmented image
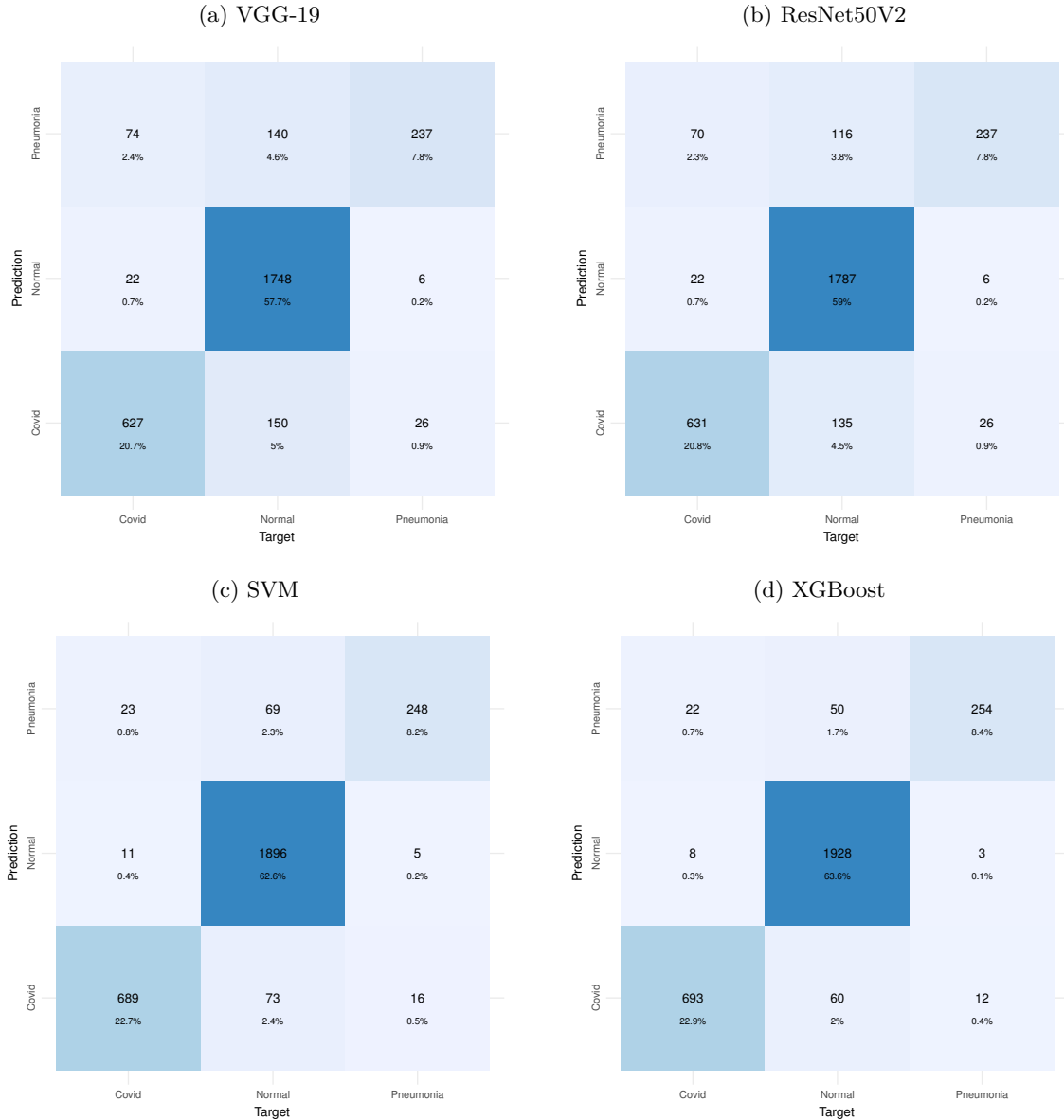
## 3.2 Classification performance

Proposed framework is compared against fine-tuned VGG19 and ResNet50V2 network performance. Fully connected head of these networks was composed of *Dense* layer with 128 units with ReLu activation, followed by *Dropout* layer with a rate of 0.5, then followed by another *Dense* layer with 64 units with ReLu activation and finally the softmax classifier. It can be seen from Table 2 that the classification performance using combined features is superior to the one using VGG19 or ResNet50V2 alone in all metrics. It is also observed that CLAHE enhancement as well as segmentation indeed improved classification performance across all models.

Table 2: Binary classification network performance comparison

| Model | Data | Accuracy | Balanced Accuracy | Precision | Recall | Specificity | F1 |
|-------|------|----------|-------------------|-----------|--------|-------------|-----|
| VGG19 | Original | 84.85 | 84.52 | 84.52 | 74.62 | 89.68 | 77.96 |
| | CLAHE | 85.68 | 85.65 | 85.65 | 75.66 | 90.13 | 79.09 |
| | CLAHE & U-Net | 86.07 | 86.36 | 86.37 | 76.22 | 90.34 | 79.70 |
| ResNet50V2 | Original | 85.48 | 85.62 | 85.62 | 75.40 | 90.03 | 78.79 |
| | CLAHE | 86.50 | 86.66 | 86.66 | 76.81 | 90.63 | 80.34 |
| | CLAHE & U-Net | 87.62 | 87.69 | 87.69 | 78.05 | 91.33 | 81.52 |
| SVM | Original | 88.88 | 89.18 | 89.18 | 80.14 | 91.92 | 83.66 |
| | CLAHE | 91.12 | 91.36 | 91.36 | 83.78 | 93.33 | 86.97 |
| | CLAHE & U-Net | 93.50 | 93.51 | 93.51 | 86.89 | 95.00 | 89.75 |
| XGBoost | Original | 89.70 | 90.64 | 90.64 | 81.41 | 92.35 | 85.03 |
| | CLAHE | 91.95 | 92.76 | 92.76 | 84.86 | 93.81 | 88.17 |
| | CLAHE & U-Net | 94.88 | 94.96 | 94.96 | 89.31 | 96.01 | 91.83 |

Figure 6: Confusion Matrices (CLAHE & U-Net)

(a) VGG-19

(b) ResNet50V2



(c) SVM

(d) XGBoost

# 4    Conclusion and future work

In this paper we have classified CXR images as normal, COVID-19 or pneumonia. X-ray machines are widely available and provide images for a quick diagnosis, therefore techniques discussed in this study could be very useful in early diagnosis of the aforementioned diseases. It was shown that CLAHE algorithm could can indeed alleviate the heterogeneity of the image which proved to be useful in the classification task. A modified U-Net architecture was proposed and successfully trained with a 98.05% accuracy. Feature fusion of different pre-trained CNN models combined with UMAP dimensionality reduction and machine learning classifiers reduced the time and saved computational resources which would be needed to train CNN from scratch. Moreover, the proposed framework outperformed the fine-tuned VGG-19 and ResNet50V2 networks.

A potential limitation of this work is a so called "Frankenstein" dataset used in this study. For pneumonia class we have used the dataset of Kermany et al. [16], which consists of paediatric patients aged between one and five. Although this is arguably one of the most popular datasets for pneumonia, many similar papers failed to mention the issue of combining CXR images of adult patients with COVID-19 and very young patients with pneumonia. While the normal class of our datasets also contains some child CXR images, classification models are likely to overperform as they are merely detecting children versus adults. It is demonstrated by Maguolo and Nanni [18] that by excluding the lung region entirely, the authors could identify the source of the images in the Cohen et al. [5] and Kermany et al. [16] datasets with an AUC between 0.9210 and 0.9997, and "diagnose" COVID-19 with an $AUC = 0.68$. Although in this study we used lung segmentation to address this shortcoming, child lungs are still different from adult lungs. Furthermore, it was found that since the lungs of children are not fully developed, it is difficult to predict the diseases using their CXR image.

The proposed framework could be improved in several ways. First, a more reliable dataset could be used to train the models. Also, U-Net could be trained on the dataset which would additionally consist COVID-19 and pneumonia patient lungs and corresponding masks. Other image enhancement techniques such as gamma correction or balance contrast enhancement could as well be explored. Moreover, the model can be trained with images of more diseases.

# 5    Acknowledgements

# References

[1] P. Afshar, S. Heidarian, F. Naderkhani, A. Oikonomou, K. N. Plataniotis, and A. Mohammadi. Covid-caps: A capsule network-based framework for identification of covid-19 cases from x-ray images. *Pattern Recognition Letters*, 138:638–643, 2020.

[2] D. Arias-Garzón, J. A. Alzate-Grisales, S. Orozco-Arias, H. B. Arteaga-Arteaga, M. A. Bravo-Ortiz, A. Mora-Rubio, J. M. Saborit-Torres, J. Ángel Montell Serrano, M. de la Iglesia Vayá, O. Cardona-Morales, and R. Tabares-Soto. Covid-19 detection in x-ray images using convolutional neural networks. *Machine Learning with Applications*, 6:100138, 2021.

[3] S. Candemir, S. Jaeger, K. Palaniappan, J. P. Musco, R. Singh, Z. Xue, A. Karargyris, S. Antani, G. Thoma, and C. J. McDonald. Lung segmentation in chest radiographs using anatomical atlases with non-rigid registration. *IEEE Trans. Medical Imaging*, 33(2):577–590, 2014.

[4] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754, 2016.

[5] J. P. Cohen, P. Morrison, and L. Dao. Covid-19 image data collection. *arXiv 2003.11597*, 2020.

[6] M. de la Iglesia Vayá, J. M. Saborit, J. A. Montell, A. Pertusa, A. Bustos, M. Cazorla, J. Galant, X. Barber, D. Orozco-Beltrán, F. García-García, M. Caparrós, G. González, and J. M. Salinas. Bimcv covid-19+: a large annotated dataset of rx and ct images from covid-19 patients, 2020.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[8] D. A. Dias Júnior, L. B. da Cruz, J. O. Bandeira Diniz, G. L. França da Silva, G. B. Junior, A. C. Silva, A. C. de Paiva, R. A. Nunes, and M. Gattass. Automatic method for classifying covid-19 patients based on chest x-ray images, using deep features and pso-optimized xgboost. *Expert Systems with Applications*, 183:115452, 2021.

[9] A. Dixit, A. Mani, and R. Bansal. Cov2-detect-net: Design of covid-19 prediction model based on hybrid de-pso with svm using chest x-ray images. *Information Sciences*, 571:676–692, 2021.

[10] Y. Fan, J. Liu, R. Yao, and X. Yuan. Covid-19 detection from x-ray images using multi-kernel-size spatial-channel attention network. *Pattern Recognition*, 119:108055, 2021.

[11] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189 – 1232, 2001.

[12] A. Haghanifar, M. Molahasani Majdabadi, and S. Ko. Covid-19 chest x-ray image repository, Jun 2020.

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.

[14] S. Jaeger, A. Karargyris, S. Candemir, L. Folio, J. Siegelman, F. Callaghan, Z. Xue, K. Palaniappan, R. K. Singh, S. Antani, G. Thoma, Y.-X. Wang, P.-X. Lu, and C. J. McDonald. Automatic tuberculosis screening using chest radiographs. *IEEE Transactions on Medical Imaging*, 33(2):233–245, 2014.

[15] A. Kalinovsky and V. Kovalev. Lung image segmentation using deep learning methods and convolutional neural networks. 10 2016.

[16] D. S. Kermany, K. Zhang, and M. H. Goldbaum. Labeled optical coherence tomography (oct) and chest x-ray images for classification, 2018.

[17] H. Koyuncu and M. Barstuğan. Covid-19 discrimination framework for x-ray images by considering radiomics, selective information, feature ranking, and a novel hybrid classifier. *Signal Processing: Image Communication*, 97:116359, 2021.

[18] G. Maguolo and L. Nanni. A critic evaluation of methods for covid-19 automatic detection from x-ray images, 2020.

[19] L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020.

[20] Z. Nabizadeh-Shahre-Babak, N. Karimi, P. Khadivi, R. Roshandel, A. Emami, and S. Samavi. Detection of covid-19 in x-ray images by classification of bag of visual words using neural networks. *Biomedical Signal Processing and Control*, 68:102750, 2021.

[21] I. S. of Medical and I. R. (SIRM). Sirm radiography dataset. `https://sirm.org/category/senza-categoria/COVID-19/`, 2020.

[22] R. S. of North America. Rsna pneumonia detection challenge. radiological society of north america. `https://www.kaggle.com/c/rsna-pneumonia-detection-challenge`, 2018.

[23] C. Ouchicha, O. Ammor, and M. Meknassi. Cvdnet: A novel deep learning architecture for detection of coronavirus (covid-19) from chest x-ray images. *Chaos, Solitons & Fractals*, 140:110245, 2020.

[24] H. Quan, X. Xu, T. Zheng, Z. Li, M. Zhao, and X. Cui. Densecapsnet: Detection of covid-19 from x-ray images using a capsule neural network. *Computers in Biology and Medicine*, 133:104399, 2021.

[25] T. Rahman, M. Chowdhury, and A. Khandakar. Covid-19 radiography database. `https://www.kaggle.com/tawsifurrahman/covid19-radiography-database`. Accessed: 2021-11-27.

[26] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, S. B. Abul Kashem, M. T. Islam, S. Al Maadeed, S. M. Zughaier, M. S. Khan, and M. E. Chowdhury. Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images. *Computers in Biology and Medicine*, 132:104319, 2021.

[27] S. Rajpal, N. Lakhyani, A. K. Singh, R. Kohli, and N. Kumar. Using handpicked features in conjunction with resnet-50 for improved detection of covid-19 from chest x-ray images. *Chaos, Solitons & Fractals*, 145:110749, 2021.

[28] R. Rashid, M. U. Akram, and T. Hassan. Fully convolutional neural network for lungs segmentation from chest x-rays. In A. Campilho, F. Karray, and B. ter Haar Romeny, editors, *Image Analysis and Recognition*, pages 71–80, Cham, 2018. Springer International Publishing.

[29] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation, 2015.

[30] P. Saha, M. S. Sadi, and M. M. Islam. Emcnet: Automated covid-19 diagnosis from x-ray images using convolutional neural network and ensemble of machine learning classifiers. *Informatics in Medicine Unlocked*, 22:100505, 2021.

[31] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[32] M. Toğaçar, B. Ergen, and Z. Cömert. Covid-19 detection using deep learning models to exploit social mimic optimization and structured chest x-ray images using fuzzy color and stacking approaches. *Computers in Biology and Medicine*, 121:103805, 2020.

[33] H. B. Winther, H. Laser, S. Gerbel, S. K. Maschke, J. B. Hinrichs, J. Vogel-Claussen, F. K. Wacker, M. M. Höper, and B. C. Meyer. Covid-19 image repository, May 2020.