

A NOVEL CNN ARCHITECTURE TO DIAGNOSE PNEUMONIA AND ITS CAUSES FROM CHEST X-RAY

Ozan Güldali 509191238, Cihat Kırankaya 507192112

<https://github.com/ozanguldali/final-covid-chestxray>

1. INTRODUCTION

COVID-19 has infected more than 85 million people worldwide, killed 1.85 million, and is still spreading rapidly around the world. It is important to detect COVID-19 as quickly and accurately as possible in order to control the spread of the disease and treat patients. While Polymerase Chain Reaction (PCR) is still the core truth of COVID-19 diagnosis, the sensitivity of PCR testing is not high enough for the low viral load or lab error present in test samples. The increase of these COVID-19 cases is considered to have a major impact on the increase of Pneumonia disease [1].

The whole world is sometimes faced with epidemics as it is currently facing COVID-19 disease. The spread and infection factors of this disease are very high. Large numbers of people from most countries become infected and continue to spread within six months of their first report. The systems necessary for any pandemic are not ready for some stages; therefore, mitigation becomes necessary with the available capacity. Recently, there has been an increase in COVID-19 borne pneumonia cases [1].

Pneumonia is an infection that makes breathing difficult when the lungs are filled with various fluids. Symptoms include severe shortness of breath, chest pain, chills, cough, headache, fever, and tiredness. Community-acquired pneumonia is still a recurrent cause of mortality [2].

It had been realized that from a clinical point of view the COVID-19 and bacteria infection caused clusters of fatal pneumonia [3]. Computed tomography (CT) has been an essential imaging method that helps the diagnosis and management of patients with COVID-19 and bacteria-based pneumonia. With this method, various information about the radiological manifestations of COVID-19 and bacterial pneumonia have emerged. It is one of the cases that the CT findings of COVID-19 pneumonia do not overlap in some people with symptoms that vary over time [4].

Some findings in the studies conducted have revealed interesting data in various patient groups. The asymptomatic patient group demonstrated early CT changes, supporting those first observed in the family cluster of COVID and bacteria. Conversely, other studies have shown abnormal CT findings with either positive PCR results for COVID-19 or initial false negative PCR results in the absence of CT changes. Noncompliance can be observed for different reasons in symptomatic patients with compatible CT and PCR findings. However, some individuals with COVID-19 and bacteria borne pneumonia may have a diagnostic dilemma given the various symptoms. The same is true for bacterial pneumonia. Therefore, it is extremely important to diagnose pneumonia automatically using tomography data and without human error [9].

Some reviews in the literature suggest that CT is a sensitive method for detecting COVID-19 and bacterial pneumonia even in asymptomatic individuals. In addition, CT can be a particularly important screening tool in a small proportion of patients with false negative PCR

results [5]. It has been evaluated to be an important method for the diagnosis of the disease in individuals who did not show any symptoms of pneumonia and had negative results in PCR tests. On the other hand, on a large scale, the use of deep learning techniques can be used to mitigate COVID-19, bacterial or other virus based pneumonia disease like outbreaks in terms of stopping the spread, diagnosis of disease, drug and vaccine discovery, treatment, patient care and much more.

Important Deep learning studies on COVID-19 and bacterial pneumonia are mentioned below. In many studies, transfer learning, use of pre-trained models, data augmentation are observed.

Tobias et al. [6] used MobileNetV2, a CNN-based and pre-trained model, in order to classify the Chest X-ray image in the normal or pneumonia category and observed the results. In the results obtained, it was seen that the model has an accuracy percentage of around 90%. Stephen et al. [7] proposed a new CNN model to detect the presence of pneumonia from chest X-ray data and determine its source. They constructed CNN model from scratch to extract features from a given chest X-ray image and classify it to determine if a person is infected with pneumonia. They have applied different algorithms to the model to obtain a remarkable classification performance. It was evaluated that the results obtained could be remarkable and alternative to other models. Brunese et al. [4] proposed a new model by making modifications on ResNet18 in order to diagnose Covid-19 and bacteria-borne pneumonia using pre-trained and transfer learning. The proposed model has been compared with other state of art techniques. In the results, it was observed that the proposed model was more successful than other models and reached a high accuracy percentage. Elshennawy and Ibrahim [8] proposed efficient and powerful deep learning models for detecting and classifying bacteria and Covid-19 borne pneumonia. In the study, performance comparison was made using pre-trained models and LSTM. In the results obtained, it is evaluated that the proposed architectures give very high results. Emhamed et al. [9], to diagnose chest X-rays and to simplify the Pneumonia detection proposed the deep learning model. Their proposed model is compared with seven state of art machine learning models. In the results obtained, it has been evaluated that the proposed model is very successful on detecting pneumonia cases.

The aim of this study is to propose a new model for the efficiently diagnosis of pneumonia from Chest X-Ray data and compare the proposed model with the most used models. Hence, A new CNN model has been proposed for the detection of Pneumonia disease for four different classes. For this aim, ProposedNet architecture was designed by determining the most appropriate hyperparameters. It has been used ResNet50, AlexNet and SqueezeNet1_1 pre-trained models to obtain a higher prediction accuracy for dataset. In addition, the diagnosis of pneumonia was conduct with CovidNet [10], DarkCovidNet [11] and NovelNet [12], which are successfully used in the literature for COVID-19 classification, and compared with the proposed model. Cross-entropy loss, which is the combination of log softmax and negative log-likelihood loss functions, were used SGD+Momentum, Adam, AdamW optimizers experimentally. To improve the success percentage and extract deep features of the proposed model, then the features were inserted into support vector machine (SVM) algorithms to final classification. The all models of the results were analyzed comparatively. The study is organized as follows: Dataset is explained in detail in Section 2.1. CNN layers, proposed model and loss function are summarized in Section 2.2. and 2.3., respectively. Optimization methods, experimental setup, and performance metrics are explained 2.4., 2.5. and 2.6. respectively. Discussion and obtained results from proposed models are presented in Section 3. Finally, in Section 4 the conclusion and the future works are summarized.

2. MATERIALS AND METHODS

2.1. Dataset

The four main classes were considered on the given dataset such as healthy people (normal), bacterial pneumonia patients (bacterial), COVID-19 pneumonia patients (viral-covid) and other viral pneumonia patients (viral-other). The original structure contains highly unbalanced data, that is why we construct a new structure, that is a new dataset from original. Since the total number of COVID-19 labeled data is too small, we augmented the Viral-Covid labeled training data after dividing original COVID-19 data into train and test data by the ratio of 0.3. That is, we chose 40 COVID-19 data as training set and rest 18 data as testing set. Then we augmented the chosen Viral-Covid labeled training data by horizontal flip, vertical flip, 90 degrees rotation, 180 degrees rotation and 270 degrees rotation. That is, we create 5 more data from 1 data, and in the end, we obtained 240 Viral-Covid labeled training data. Testing data was left untouched. Finally, we chose 240 training data and 80 testing data for normal, bacterial and viral-other labels from original. The process is as follows:

Table 1: Dataset Re-Construction

Class	Original Set	Training Set		Testing Set
		Chosen	After Augmentation	
Normal	1576	240	240	80
Bacterial	2772	240	240	80
Viral-Covid	58	40	240	18
Viral-Other	1493	240	240	80
Total	5899	760	960	258

2.2. CNN Layers and Proposed Model

CNNs are architectures made up of multiple sequential layers. These layers perform different functions. To reveal the distinctive features of data in these architectures according to the problem type they are applied [13] layer functions used in the model proposed in the study are as follows:

2.2.1 Convolution Layer

This layer is the building block of CNN architectures and is used to reveal the distinctive features of input data. This layer applies various filters to the data to reveal low and high-level features in the data [14]. After convolution, the size of the input data changes. These dimensions vary according to the layers to be applied. The outputs of the convolution layers are called activation maps and are basically shown as:

$$X_j^l = f \left(\sum_{i \in M_j} X_i^{l-1} * k_{ij}^l + b_j^l \right)$$

The convolution process is defined as above. Here, the previous layers are shown with X_i^{l-1} , the learnable kernels are k_{ij}^l and the bias term is b_j^l . M_j matches the input map section [15].

2.2.2 Rectified Linear Units (ReLU)

The nonlinear layer follows the convolution layer in the model. This layer gives the system a non-linear feature and is called the activation layer. Activation maps are used to calculate the outputs of the neural network using linear combinations [12].

For this purpose, the most frequently used activation function Rectifier (ReLU) was preferred. The function is defined as follows:

$$f(x) = \max(0, x)$$

2.2.3 Pooling Layer

It is common to periodically insert a Pooling layer in-between successive Conv layers in a CNN architecture. Its function is to progressively reduce the spatial size of the representation to reduce the number of computational nodes in the network. Hence overfitting is tried to under control [12].

An input with size $W_1 \times H_1 \times D_1$ is allowed in this layer. Two parameters can be mentioned here; their spatial extent F, the stride S, produces a volume of size $W_2 \times H_2 \times D_2$ where:

$$W_2 = (W_1 - F) / S + 1, H_2 = (H_1 - F) / S + 1 \text{ and } D_2 = D_1$$

In our proposed model, MaxPooling was applied at the exit of each convolutional block.

2.2.4 Batch Normalization


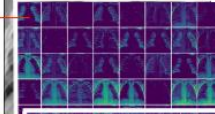




Batch normalization is a technique for training deep neural networks that standardize inputs to one layer for each mini group. This has the effect of stabilizing the learning process and significantly reducing the number of training sessions required to train deep networks [12]. This processing element has been used in all convolutional layers.

2.2.5 Fully-Connected Layers

This structure is used to transfer the activations obtained by transferring data over the network for the next unit. It is located at the end of the architecture to provide connections between all the activations and compute nodes on these layers. Three layers are used in the proposed model.

In this study, a new CNN model that consists of four convolutional blocks and three fully-connected layers for diagnose pneumonia. At the end of the proposed model, the cross-entropy loss is used. The details of the proposed model are given in Table 2 below.

Table 2: The details of the proposed model with the corresponding activation maps

		LAYER TYPE	OUTPUT	KERNEL	STRIDE	PADDING	PARAMETERS	
Convolutional Layers	Chest X-Ray CT	ImageInput-0	[3, 224, 224]				0	
	Block 1	Conv2d-1	[64, 222, 222]	5	1	1	4,864	
		ReLU-2	[64, 222, 222]				0	
		Conv2d-3	[64, 220, 220]	5	1	1	102,464	
		ReLU-4	[64, 220, 220]				0	
		MaxPool2d-5	[64, 110, 110]	2	2	1	0	
	Block 2	Conv2d-6	[128, 110, 110]	5	1	1	73,856	
		ReLU-7	[128, 110, 110]				0	
		Conv2d-8	[128, 110, 110]	5	1	1	147,584	
		BatchNorm2d-9	[128, 110, 110]				256	
		ReLU-10	[128, 110, 110]				0	
		MaxPool2d-11	[128, 55, 55]	2	2	1	0	
	Block 3	Conv2d-12	[256, 56, 56]	3	1	1	131,328	
		ReLU-13	[256, 56, 56]				0	
		Conv2d-14	[256, 57, 57]	3	1	1	262,400	
		ReLU-15	[256, 57, 57]				0	
		Conv2d-16	[256, 58, 58]	3	1	1	262,400	
		BatchNorm2d-17	[256, 58, 58]				512	
		ReLU-18	[256, 58, 58]				0	
		MaxPool2d-19	[256, 29, 29]	2	2	1	0	
	Block 4	Conv2d-20	[512, 30, 30]	3	1	1	524,800	
		ReLU-21	[512, 30, 30]				0	
		Conv2d-22	[512, 31, 31]	3	1	1	1,049,088	
		ReLU-23	[512, 31, 31]				0	
		Conv2d-24	[512, 32, 32]	3	1	1	1,049,088	
		BatchNorm2d-25	[512, 32, 32]				1,024	
		ReLU-26	[512, 32, 32]				0	
		MaxPool2d-27	[512, 16, 16]	2	2	1	0	
Fully-Connected Layers		AdaptiveAvgPool2d-28	[512, 4, 4]				0	
		Flatten-29	[8192]				0	
	FC 1	Linear-30	[1024]				8,389,632	
	FC 2	ReLU-31	[1024]				0	
		Dropout2d-32	[1024]				0	
		Linear-33	[1024]				1,049,600	
	FC 3	ReLU-34	[1024]				0	
		Dropout2d-35	[1024]				0	
		Linear-36	[4]				4,100	

2.3. Cross-Entropy Loss Function

Cross-entropy objective function is the combination of log softmax and negative log-likelihood loss functions.

Each estimated class likelihood is compared to the target performance 0 or 1 of the actual class, and a loss is determined that penalizes the probability depending on how far the real expected value is formed.

$$-\frac{1}{n} \sum_{i=1}^n y_i \ln(\hat{y}_i)$$

such that y is the actual label and \hat{y} is the classifier's output where n is the number of samples

2.4. Optimization Methods

In the experiments, Adam with weight decay, Adam with decoupled weight decay and SGD with momentum were used. The short brief of optimizers can be found in [Table 3](#).

Table 3: Optimizers

(f : loss function / η : initial learning rate / $\beta, \beta_{1,2}$: momentum parameters / g_t : gradient at time t along θ / w : weight decay rate)

Optimizer	Brief	Formula
SGD Momentum	Stochastic gradient descent is an iterative method for optimizing an objective function with suitable smoothness properties. It can be regarded as a stochastic approximation of gradient descent optimization. Momentum remembers the weight update at each iteration, and determines the next update as a linear combination of the gradient and the previous update [15].	$g_t = \nabla_{\theta} f_t(\theta_{t-1})$ $v_t = \beta v_{t-1} + (1 - \beta) g_t$ $\theta_t = \theta_{t-1} - \eta v_t$
Adam Weight Decay	Adaptive moment estimation is an optimization algorithm that uses the running averages of both the gradients and the second moments of the gradients. Before each step, the gradient of loss function is calculated and a step in the opposite direction is taken with corresponding rate. This process is also called as Adam with L2 regularization [16].	$g_t = \nabla_{\theta} f_t(\theta_{t-1}) + w \theta_{t-1}$ $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$ $v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$ $\theta_t = \theta_{t-1} - \eta \frac{m_t / (1 - \beta_1^t)}{\sqrt{v_t / (1 - \beta_2^t)} + \epsilon}$
Adam Decoupled Weight Decay	Adam with decoupled weight decay does not use the regularization as the normalized by square root of v . Thus, it is only proportional to the weight itself [16].	$g_t = \nabla_{\theta} f_t(\theta_{t-1})$ $m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$ $v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$ $\theta_t = \theta_{t-1} - \eta \frac{m_t / (1 - \beta_1^t)}{\sqrt{v_t / (1 - \beta_2^t)} + \epsilon} + w \theta_{t-1}$

2.5. Experimental Setup

The source code was written as Python programming language with the help of PyTorch and Sklearn library. For GPU support, Google Colaboratory was used with the number of workers as four on experiments.

The dataset was constructed by choosing 240 samples for training and 80 samples for testing from each of Normal, Bacterial and Viral-Other labels of the given original data. From the Viral-Covid labeled data, 18 samples were chosen for testing, and 40 samples were chosen for training and then each Viral-Covid training datum was augmented 5 times. That is, the dataset totally has 960 training and 258 testing samples. The training data was shuffled on each epoch during experiments.

To prepare the data for experiments, each was resized to 224*224, center cropped (squared), gray-scaled, and normalized by the mean of [0.485, 0.456, 0.406] and standard deviation of [0.229, 0.224, 0.225]. The size of each batch was experimented from 2^5 to 2^7 .

AlexNet, ResNet50 and SqueezeNet1_1 were used as pretrained on ImageNet, and ProposedNet, NovelNet, CovidNet and DarkCovidNet was coded and experimented from scratch.

Cross-entropy loss was chosen as the objective function, and SGD with Momentum, Adam and Adam with decoupled weight decay were experienced to optimize the loss. Various learning rates, momentum values, betas and weight decay parameters were experimented to obtain the optimal results. The learning rate update was also experienced as reducing the learning rate by 0.1 for each quarter of total epoch size.

Final hyperparameter settings for our proposed CNN model are as in [Table 4](#).

Table 4: Final hyperparameters for ProposedNet
(Scheduler refers the update LR algorithm)

Optimizer	Learning Rate	Scheduler	Betas	Weight Decay	Momentum
SGD Momentum	1e-4	True	(0.9, 0.999)	1e-4	-
Adam	1e-4	True	(0.99, 0.99)	1e-4	-
AdamW	1e-3	False	-	-	0.9

Each experiment was run in 200 epochs with test data validation once in 10 epochs. In comparison, the accuracy was compared with the previously stored results for each validation and the final test and saved as a new model weight file if it is higher than the older one.

Grid Search was used to get the best settings on SVM. In total, 720 possibilities were experienced. The final hyper-parameters of SVM are:

- Regularization parameter: ‘5.0’ over [0.01, 0.05, 0.1, 0.5, 1.0, 5.0]
- Kernel type: ‘RBF (Radial Basis Function)’ over [linear, rbf, sigmoid]
- Gamma: ‘scale’ over [scale, auto];
i.e., $(1 / (\text{number of features} * \text{variance}(\text{feature data})))$ is used as the value of gamma
- Decision_function_shape: ‘one-vs-one’ over [one-vs-one, one-vs-rest] and
- Seed: ‘4’ over [1, 2, 4, 13, 17, 23, 25, 100, 999, 1773]

10-fold cross validation for ML algorithms was used with the seed as 4.

Feature extraction from the ProposedNet model weights on Fully-Connected-2 layer was applied to whole dataset. 1024 features were obtained for each sample, and these features were saved as *.numpy files to use in experiments. Then by using 10-fold cross validation, SVM were performed on extracted feature sets.

2.6. Performance Metrics

On the experiments, we made measurements and comparisons considering the number of classes and the unbalanced data label distribution. During the training, we observed train & evaluation accuracies and losses. On testing, we inspected the test loss, accuracy, area under curve (AUC) score, sensitivity, specificity, precision and F1 score for the computed confusion matrix as below:

- Sensitivity = $\text{True Positive} / (\text{True Positive} + \text{False Negative})$
- Specificity = $\text{True Negative} / (\text{True Negative} + \text{False Positive})$
- Precision = $\text{True Positive} / (\text{True Positive} + \text{False Positive})$
- F1 Score = $2 * \text{Precision} * \text{Sensitivity} / (\text{Precision} + \text{Sensitivity})$

3. RESULTS AND DISCUSSION

In this study, we proposed a novel CNN architecture for the diagnosed of pneumonia over four classes from Chest X Ray images. To compare the efficiency of ProposedNet, we obtained results from recently proposed models NovelNet, CovidNet, DarkCovidNet and pretrained models AlexNet, ResNet50, SqueezeNet1_1 as explained in [Section 2.6](#). At the end, we achieved the most successful results on ProposedNet with Adam optimizer.

Different models reached their best accuracies on different hyperparameters settings and with different optimizers. This detailed review can be found on [Table 5](#).

The accuracy and loss curves of training and validation stages are shown in [Figures 1-4](#) below.

Table 5: Prediction performance and results

Pretrained	Model	Accuracy		
		Adam	AdamW	SGD Momentum
FALSE	*ProposedNet	87.21 (40 epochs)	87.21 (120 epochs)	87.21 (190 epochs)
FALSE	DarkCovidNet	84.88	86.05	83.33
FALSE	CovidNet	86.04	84.88	82.55
FALSE	NovelNet	79.84	81.00	82.55
TRUE	AlexNet	82.64	84.88	86.04
TRUE	ResNet50	83.72	84.49	86.82
TRUE	SqueezeNet1_1	83.72	84.88	75.88

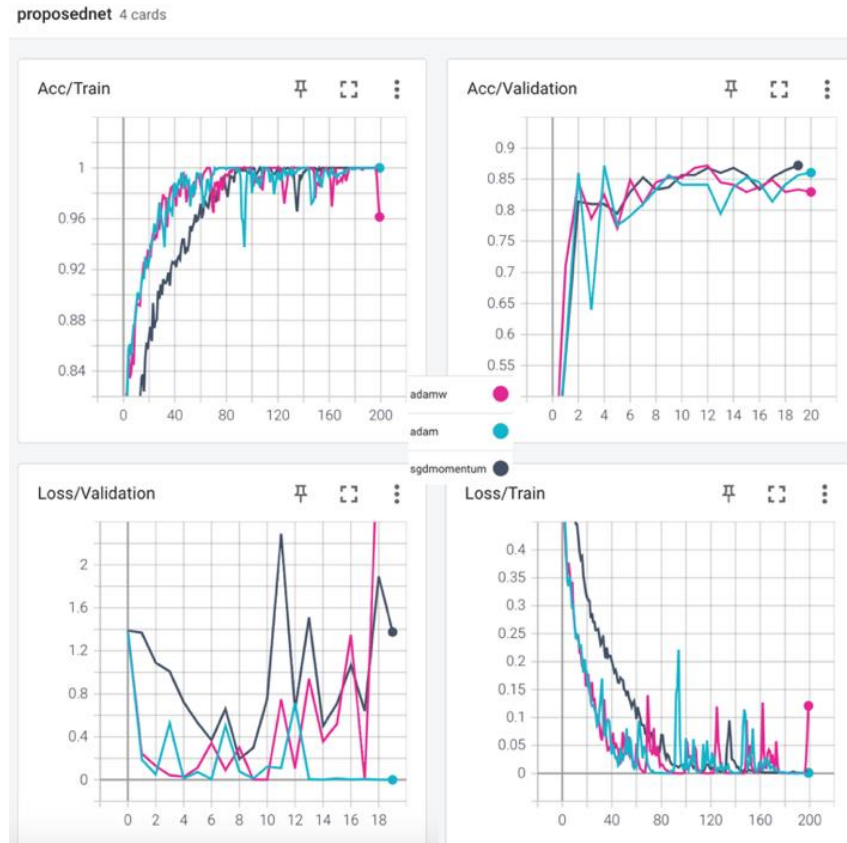
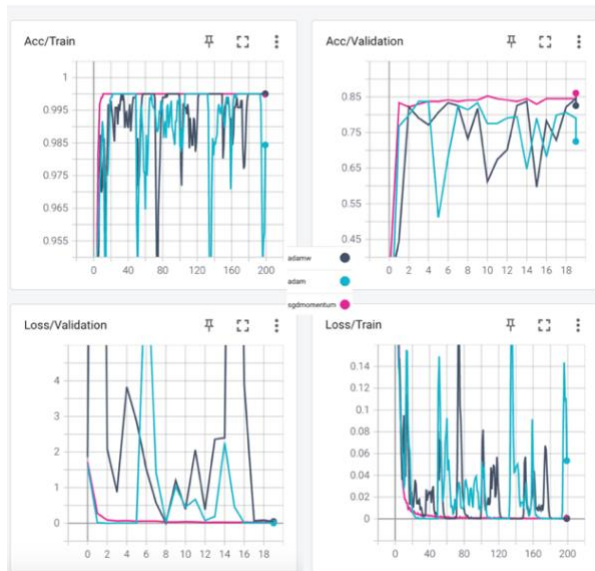


Figure 1: The performance of ProposedNet

resnet50 4 cards



alexnet 4 cards

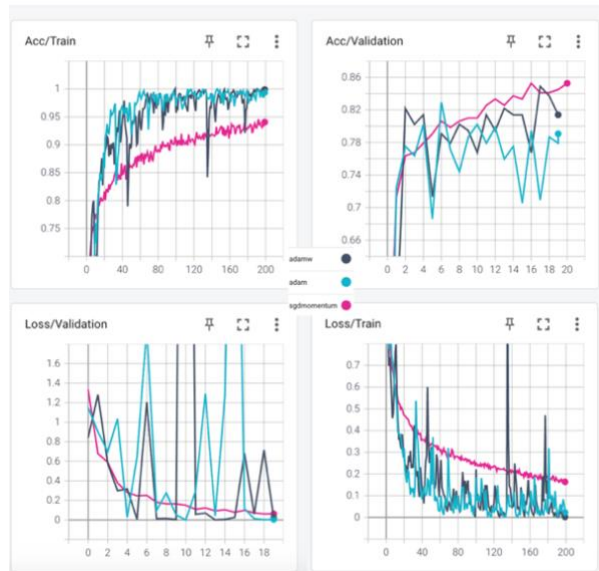
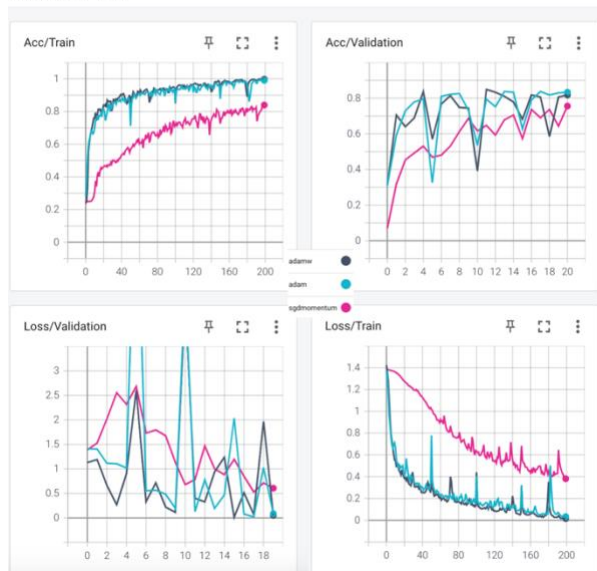


Figure 2: The performances of pretrained Resnet50 and pretrained AlexNet

squeezeNet1_1 4 cards



covidnet 4 cards

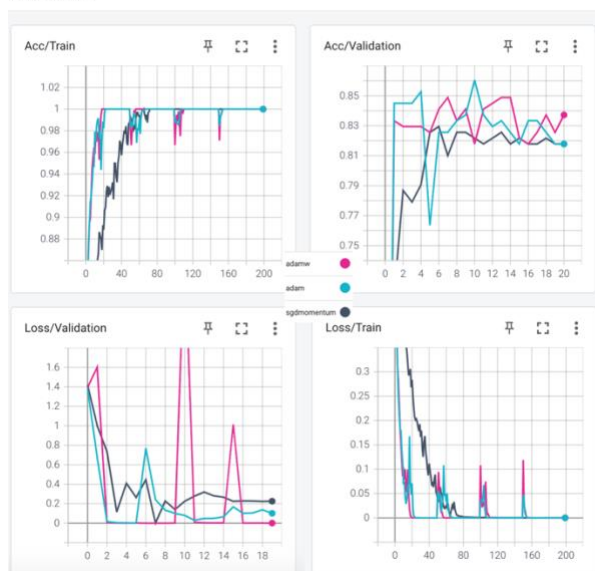
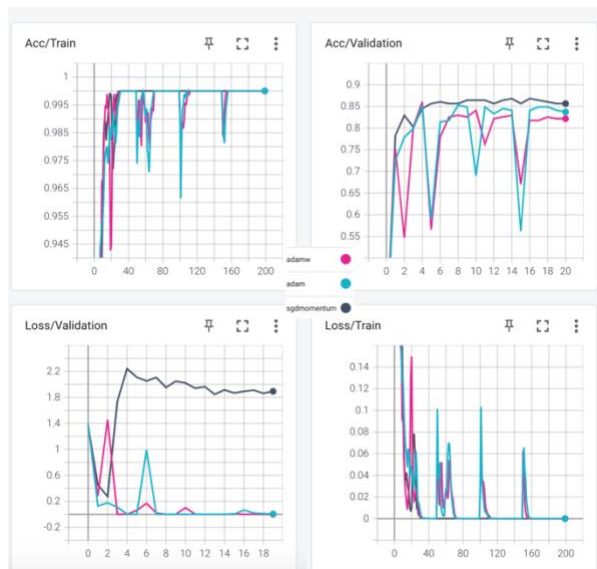


Figure 3: The performances of pretrained SqueezeNet1_1 and CovidNet

darkcovidnet 4 cards



novelnet 4 cards

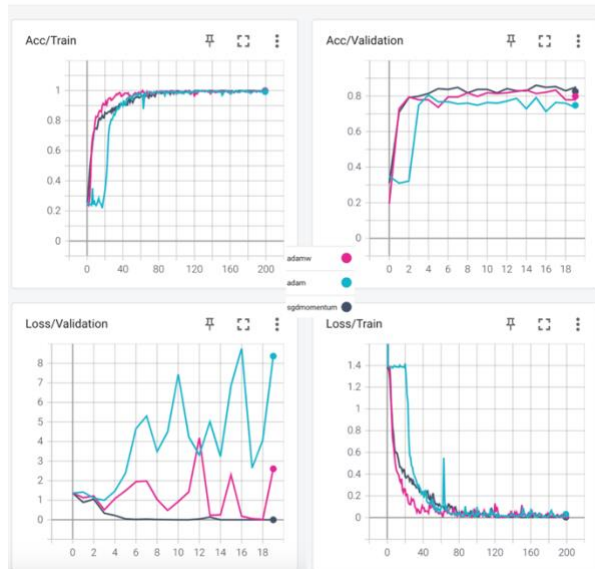


Figure 4: The performances of DarkCovidNet and NovelNet

It can be seen from the [Table 5](#) that the best result was obtained faster by using Adam in ProposedNet model with the accuracy of 87.21%. Moreover, it also has the highest AUC Score as 0.967805.

To solve the problem of 4-class classification on pneumonia disease with the highest and useful result, we collected the deep features of dataset from the FC-2 layer of weighted ProposedNet. Then SVM was experienced and performed as explained in [Section 2.6](#). The satisfying result was obtained as the accuracy of 97.95% and the AUC score of 0.995995. The overall final results and comparison can be found in [Table 6](#).

Table 6: Performance metrics (See [Table 7](#) and [Section 2.7](#) for computations)

Method	Accuracy (%)	Sensitivity (%)	Specificity (%)	Precision (%)	F1-Score (%)
SVM	84.65	84.84	84.86	84.43	84.49
Pretrained Squeezenet1_1 (AdamW)	84.88	84.88	84.98	84.65	84.65
Pretrained AlexNet (SGD Momentum)	86.04	86.04	86.26	86.08	86.80
Pretrained ResNet50 (SGD Momentum)	86.82	86.68	87.02	86.93	86.70
NovelNet (SGD Momentum)	82.55	82.55	82.72	82.55	82.36
CovidNet (Adam)	86.04	86.04	86.26	85.88	85.76
DarkCovidNet (AdamW)	86.05	86.43	86.64	86.02	86.17
ProposedNet (Adam)	87.21	87.21	87.40	87.34	87.03
SVM with features extracted from ProposedNet weights	97.95	97.94	97.97	97.94	97.94

Table 7: Confusion matrices for ProposedNet and SVM with deep features

ProposedNet (Adam)		Actual Class			
Predicted Class	Class	Bacterial	Normal	Viral-Covid	Viral-Other
	Bacterial	61	1	0	6
	Normal	6	77	0	4
	Viral-Covid	1	1	17	0
	Viral-Other	12	1	1	70

SVM with deep features		Actual Class			
Predicted Class	Class	Bacterial	Normal	Viral-Covid	Viral-Other
	Bacterial	30,9	0,1	1	0,8
	Normal	0,1	31,7	0	0,2
	Viral-Covid	0,1	0,1	25,7	0
	Viral-Other	0,9	0,1	0,1	31

4. CONCLUSION

Pneumonia is a common and serious lung disease caused by viruses and bacteria. Although it can be treated on time with early diagnosis and detection play an important role in the treatment of the disease in cases where the human cannot diagnose. Since the emergence of COVID-19 and before, it has been widely used to quickly and efficiently diagnose pneumonia caused by COVID-19, bacteria and other viruses using chest X-rays and deep learning techniques [17].

In the study, we defined our problem as 4-class classification on pneumonia disease, then construct our dataset. After researching on literature, we draw our roadmap and proposed a novel CNN model named as ProposedNet regarding to the defined problem. To obtain the best comparison results, we made too many experiments with different optimizers and hyperparameters. After all experiments, we reached the best results with our proposed CNN model, ProposedNet, with Adam optimizer and hyperparameters stated in [Table 4](#). The results are **the accuracy of 87.21%, the AUC Score of 0.96, the sensitivity of 87.21%, specificity of 87.40%, precision of 87.34% and the F1-score of 87.03%.**

To show the power of ProposedNet on solving the main task, we collected the deep features of dataset extracted from the FC2-Layer, and used them on the experiments of SVM. By the help of grid search, we experimented 720 probabilities and find the optimal settings. Finally, with the seed as 4 and the determined settings stated on [Section 2.6](#), we concluded the determined problem with **the accuracy of 97.95%, the AUC score of 0.995995, the sensitivity of 97.94%, specificity of 97.97%, precision of 97.94% and the F1-score of 97.94%.**

The detailed information for running and reproducing the experiments are stated in the [ReadMe File of project](#). The constructed dataset can be found on the [GitHub Link for dataset](#). For the reproducibility, the model weights of ProposedNet can be found on [GitHub Link for weights](#), and the extracted features and labels can be found on [GitHub Link for features](#) and [GitHub Link for labels](#).

In conclusion, the obtained results can be clearly considered as the highest ones in the literature. That is why, the proposed CNN model ProposedNet, with the beneficial usage example observed on this study, is a significant contribution to the literature for the diagnosis of pneumonia disease. With this model and methodology, the pneumonia and its cause can be detected automatically and quickly.

In future studies, the similar multiclass problems on Chest X-Ray can be experimented with the ProposedNet and with the similar methodology used in this paper for final results.

REFERENCES

1. H.A. Rothan , S.N. Byrareddy , The epidemiology and pathogenesis of coronavirus disease (Covid-19) outbreak, *Journal of autoimmunity* (2020) 102433.
2. Wardlaw, T.; Johansson, E.W.; Hodge, M. *Pneumonia: The Forgotten Killer of Children*; United Nations Children’s Fund (UNICEF): New York, NY, USA, 2006; p. 44.
3. Q. Li , X. Guan , P. Wu , X. Wang , L. Zhou , Y. Tong , R. Ren , K.S. Leung , E.H. Lau, J.Y. Wong , et al. , Early transmission dynamics in wuhan, china, of novel coronavirus infected pneumonia, *New England Journal of Medicine* (2020)
4. Al-Falluji R.A., Katheeth Z.D., Alathari B: Automatic Detection of COVID-19 Using Chest X-Ray Images and Modified ResNet18 Based Convolution Neural Networks, 2020.
5. E. Y. P. Lee, M.Y. Ng and P. L. Khong, “COVID-19 pneumonia: What has CT taught us?,” *Lancet Infectious Diseases*, vol. 20, no. 4, pp. 384–385, 2020.
6. Tobias, R.R., De Jesus, L.C.M.; Mital, M.E.G., Lauguico, S.C.; Guillermo, M.A., Sybingco, E., Bandala, A.A. Dadios, E.P. CNN-based Deep Learning Model for Chest X-ray Health Classification Using TensorFlow. In *Proceedings of the 2020 RIVF International Conference on Computing and Communication Technologies, RIVF 2020, Ho Chi Minh, Vietnam, 14–15 October 2020* .
7. Stephen, O., Sain, M., Maduh, U.J., Jeong, D.U. An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare. *J. Healthc. England*, 2019.
8. Elshennawy, N.M., Ibrahim, D.M. Deep-Pneumonia Framework Using Deep Learning Models Based on Chest X-Ray Images. *Diagnostics* 2020, 10, 649.
9. Emhamed, R., Mamlook, A., Chen, S. Investigation of the performance of Machine Learning Classifiers for Pneumonia Detection in Chest X-ray Images. In *Proceedings of the 2020 IEEE International Conference on Electro Information Technology (EIT), Chicago, IL, USA, 31 July–1 August 2020*; pp. 98–104.
10. Wang L., Lin Z.Q., Wong A. COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images, *Scientific Reports* 2020.
11. Ozturk T, Talo M, Yildirim EA, Baloglu UB, Yildirim O, Acharya UR, Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Comput Biol Med* (2020), 121:103792.
12. M. Nour, Z. Cömert, K. Polat: A Novel Medical Diagnosis model for COVID-19 infection detection based on Deep Features and Bayesian Optimization (2020) 106508, <https://doi.org/10.1016/j.asoc.2020.106580>.
13. E. Başaran, Z. Cömert, Y. Çelik, Convolutional neural network approach for automatic tympanic membrane detection and classification, *Biomed. Signal Process. Control* 56 (2020) 101734, <http://dx.doi.org/10.1016/j.bspc.2019.101734>.
14. M. Toğaçar, B. Ergen, Z. Cömert, BrainMRNet: Brain tumor detection using magnetic resonance images with a novel convolutional neural network model, *Med. Hypotheses*. 134 (2020) 109531, <http://dx.doi.org/10.1016/j.mehy.2019.109531>.
15. Sutskever, Ilya; Martens, James; Dahl, George; Hinton, Geoffrey E. (June 2013). Sanjoy Dasgupta and David Mcallester (ed.). On the importance of initialization and momentum in deep learning (PDF). In *Proceedings of the 30th international conference*

- on machine learning (ICML-13). **28**. Atlanta, GA. pp. 1139–1147. Retrieved 14 January 2016.
16. F. M. Graetz; Why AdamW Matters (2018), <https://towardsdatascience.com/why-adamw-matters-736223f31b5d>
 17. Yu X., Wang S.H., Zhang Y.D. CGNet: A graph-knowledge embedded convolutional neural network for detection of pneumonia, Information Processing and Management 58 (2021), <https://doi.org/10.1016/j.cmpb.2020.105608>