Constrained and unconstrained deep image prior optimization models with automatic regularization

International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. , No. (Mon-20..)

http://dx.doi.org/10.12785/ijcds/XXXXXX

Transferred cascade CNNs for COVID-19 classification from chest x-ray images

Nassima DIF[1] and Zakaria ELBERRICHI[2]

[1]*Ecole Superieure en Informatique, LabRI-SBA Lab, Sidi Bel Abbes, Algeria*
[2]*Computer science Department, Djillali Liabes University, EEDIS Laboratory, Sidi Bel Abbes, Algeria*

Received Mon. 20, Revised Mon. 20, Accepted Mon. 20, Published Mon. 20

Abstract:
In light of the recent COVID-19 pandemic, identifying effective techniques for detecting the virus has become crucial. One cost-effective approach is to analyze X-ray images. Deep learning systems have been suggested as an alternative diagnostic tool to assist doctors. However, available datasets only classify X-ray images into Normal, Pneumonia, and COVID-19 categories. It is essential to distinguish between bacterial and viral pneumonia due to their different treatment forms. This paper presents three new cascade systems to differentiate between COVID-19 and non-COVID-19 pneumonia and classify bacterial and viral pneumonia based on a newly compiled dataset. The proposed cascade system (TCCNN) allows the model to quickly identify complex concepts from the data by combining different types of convolutional neural networks in two or three stages. Additionally, TCCNN introduces transfer learning within the cascade system, allowing the convolutional neural network in the current stage to exploit the trained model from the previous stage. The comparative analysis revealed the efficiency of the proposed systems, where the two-stage system PN_CBV achieved an accuracy of 96.27% based on the DenseNet201_DenseNet121 combination. This study validates the efficiency of the proposed two-stage cascade systems compared to the three-cascade system and advantage transfer learning within these strategies.

Keywords: Covid-19, Deep learning, Chest x-ray images, Classification, Transferred cascade CNNs

1. Introduction

Since December 2019, new pneumonia has been detected in China and has affected a large number of people. This disease had a similar behaviour as SARS. In March 2020, WHO declared this pandemic as COVID-19 [1]. The symptoms of COVID-19 include fever, cold, dry cough, breathing difficulties, and acute respiratory syndrome [2]. Because of its high transmissibility, controlling the spread of the virus has become urgent.

For the diagnosis of COVID-19, there are three main clinical tools in use: Real-time polymerase chain reaction (RT-PCR), computerized chest tomography (CT), and chest X-Rays (CXR) scans. RT-PCR tests risk missing positive cases due to various technical problems. Moreover, testing kits and the long processing time (4–6 hours [3]) can result in a rapid spreading rate of COVID-19. As an alternative and to control the Covid-19 spreading, radiological images such as X-rays and CT-sans have been exploited. While CT imaging presents several disadvantages, such as high radiation doses and sensitivity to patient movements [4], X-ray imaging is patient-friendly, fast, cheap, and can detect the disease early. Additionally, X-ray scanners are largely available. However, X-ray scans take a long time to detect COVID-19 and require an expert radiologist. Moreover, their manual analysis is time-intensive and can be influenced by doctors' subjectivity. To reduce the doctor's workload and avoid their subjective decisions, researchers have proposed exploiting computer-aided diagnostic systems (CAD) for COVID-19 detection from X-rays.

Recent advances in machine learning (ML) and deep learning (DL) techniques have enabled the development of CAD systems for X-ray image analysis [5], [6]. Convolutional neural networks (CNNs) are the most popular architectures due to their advantages for image analysis [7], [8], [9], [10]. However, these architectures risk overfitting due to the lack of data. To address this problem, many efforts have been made to create large datasets from different sources [11], [12], [13], [14]. To the best of the authors' knowledge, all voluminous datasets categorize X-rays into three classes: Covid-19, Normal, and Pneumonia due to the difficulty of multi-class classification systems.

E-mail address: author 1, author 2, author 3

The purpose of this contribution is to propose a four-classification system that distinguishes between COVID-19, normal, and viral and bacterial pneumonia. This classification helps to avoid misclassifying COVID-19 samples due to their similar characteristics with other viral pneumonia [15]. To simplify the multi-class classification task, we proposed a cascade system that divides the classification according to annotation complexity, starting with the easiest and progressing to the most complex. TCCNN helps to quickly identify more complex concepts from data by stacking CNNs sequentially. The main contribution is that each model in our cascade system exploits the experience of the previous model through transfer learning. This was motivated by the advantages of transfer learning between similar and non-distant classification tasks [16].

This paper presents several contributions, including the creation of the largest dataset that is composed of four classes: COVID-19, normal, bacterial, and viral. Additionally, we propose three types of cascade systems based on six CNN architectures: two types of two-stage cascade strategies and a three-stage cascade system for covid-19 classification. Our study integrates a transfer learning strategy within the proposed systems. Furthermore, we compare one-stage, two-stage, and three-stage cascade strategies, and between transfer learning from ImageNet and transfer learning from x-rays in cascade systems. This intense comparison provides an interesting reference on the best method for classifying x-rays into four labels.

The remaining parts of this paper proceed as follows: the first section details the related works to deep learning methods for COVID-19 classification. Section 2 explains the proposed method. Section 3 presents and discusses the obtained results. Finally, the last section concludes this work.

2. Related works

To address the COVID-19 epidemic, many efforts have been made to design deep learning applications for COVID-19 detection based on X-rays [17], [12], [18], and CT scans [19] of the chest. Various deep learning architectures have been exploited, such as convolutional neural networks (CNNs) [20] and Long short-term memory (LSTM) [21]. CNNs attracted much interest in detecting COVID-19 from X-rays due to their advantages for image processing. Predictive models for these architectures can be generated by either training from scratch [17] or using transfer learning techniques [12], [13], [18], [22], [20]. In [22], the comparative analysis demonstrated the efficiency of transfer learning over training from scratch based on the VGG16 architecture. Similarly, Nayak et al. [20] highlighted the important results of transfer learning from the ResNet−50 compared to the other six CNN architectures. Chowdhury et al. [12] used a transfer learning technique from 7 imageNet models and one pre-trained on X-ray images CheXNet. Their experimental study revealed that CheXNet was more efficient for binary classification, whereas, DenseNet201 was more promising in the three-class classification scheme. In another investigation [18], using the fine-tuned ResNet50 architecture as a feature extractor with SVM was more efficient than other pre-trained models and a trained CNN architecture from scratch. In [17], a new residual architecture was proposed to extract features at different abstraction levels. This approach uses two parallel convolutions with different filter sizes to capture multi-scale features. Another approach, suggested by Öksüza et al. [23], proposes using fine-grained, coarse-grained, and coarser-grained maps generated from three different networks: SqueezeNet, ShuffleNet, and EfficientNet-B0. This ensemble approach achieved promising results in detecting COVID-19 from chest x-ray images.

To combine the decisions of several learners, Win et al. [24] proposed an ensemble deep learning technique. Their approach was evaluated using 11 types of CNNs. The authors combined the five best models using soft and hard voting techniques. Similarly, Brunese et al. [25] proposed an approach that combines various VGG16 models in a threefold binary classification framework. The first stage classified X-rays into normal or pulmonary cases, while the second stage differentiated COVID-19 from non-COVID-19 pneumonia. Finally, the third stage used the Gradient Class Activation Map (Grad-CAM) to localize suspected COVID-19 areas.

Our proposed contribution shares some similarities with the approach presented in [26], which proposed a two-cascade network. Their method first categorizes x-ray images into normal, pneumonia, or tuberculosis, and then distinguishes between normal pneumonia and covid-19 using the most efficient network for each level. In contrast, our contribution proposes three cascade strategies, including two types of two-stage cascade systems and a three-stage cascade strategy, with transfer learning applied within each stage. Furthermore, our approach also has the capability to classify other pneumonia diseases, such as bacterial and viral infections.

3. Proposed method

A. *Cascade CNN systems*

The purpose of cascade systems in COVID-19 detection from X-rays is to distinguish between normal and pneumonia classes or between pneumonia sub-classes at each level. In these systems, the classification is divided based on the complexity of annotations, starting from the easiest to the most complex. Discriminating between normal and pneumonia is less challenging than classifying pneumonia sub-classes, such as COVID-19, viral, and bacterial. Dividing the classification process into several stages reduces the model's classification load and can also help to reduce the error rate. These cascade systems are sequential ensemble learning strategies that combine the decisions of multiple learners to improve generalization. The originality of our ensemble learning system lies in the ability of each model to use the previous model's experience through a transfer

learning method within the cascade system. Figure 1 illustrates the general structure of the proposed cascade system.

For the two-stage cascade system, X-ray images are classified into n categories using a pre-trained CNN on the ImageNet dataset. Subsequently, the first class is partitioned into m sub-categories, and the resulting dataset is passed through a second CNN. This network can be either a pre-trained CNN on ImageNet or the model developed in the first training stage. The purpose of transfer learning from stage 1 is to exploit the extracted features from dataset 1 to classify the sub-dataset which contains m sub-classes (class1.1, class1.2, ..., class1.m). The three-stage cascade strategy includes an additional step compared to the two-stage cascade method, where subclass 1.1 of class 1 is further divided into k sub-categories (class1.1.1, class1.1.2, ..., class1.1.k). Finally, the resulting dataset is passed through a pre-trained CNN from ImageNet or the previously fine-tuned model from the second stage, and the generated model is fine-tuned on this new dataset.

The first two-stage scheme denoted as PCN_BV, involves the classification of three distinct classes, namely non-COVID-19 pneumonia, COVID-19, and normal. Among these, the non-COVID-19 pneumonia class is further divided into a sub-group of m classes consisting of bacterial and viral categories. On the other hand, the second system (PN_CBV) deals with the classification of pneumonia and normal classes, and the group of pneumonia sub-classes consists of COVID-19, bacterial, and viral. The main difference between the two-stage schemes is the classification level of the COVID-19 class. The primary objective of this study is to determine whether it is challenging to distinguish between COVID-19, pneumonia, and normal classes in the first stage or to classify COVID-19, viral, and bacterial images in the second stage.

In the three-stage cascade system (PN_PC_BV), the group of n classes includes normal and pneumonia. The pneumonia class is further divided into non-pneumonia COVID-19 and COVID-19 subcategories. Finally, the last group of k classes presents subcategories of non-pneumonia COVID-19, which include bacterial and viral pneumonia.

In this study, we employed the transfer learning technique from ImageNet due to its extensive use in the literature and its efficiency. Additionally, we proposed to perform transfer learning between different models in the cascade system. This method aims to reuse previously extracted features by the CNN in the previous stage, as the label "i" in stage k groups images that share similar features and morphology to images in its subcategories in stage k+1. In the cascade system, the trained network on dataset k was used as a source model for transfer learning to dataset k+1.

The transfer learning method consists of three main steps. First, the target CNN is initialized by the weights ($P(X_s \mid Y_s)$) of the source CNN, which was previously trained on ImageNet or X-ray images. Then, the last fully connected layers are removed and replaced by two fully connected layers and one softmax layer. The fully connected layers are composed of 1024 and 512 neurons, respectively, while the softmax layer contains C neurons, where C represents the number of labels in T_T. Finally, the new network is fine-tuned on the target task.In this study, We used six CNN architectures: VGG16, VGG19, Inception, Xception, DenseNet201, DenseNet121. The fine-tuning process was performed according to CNN's nature.

4. Experimental study

The experiments were conducted on a computer with an Intel i5-core processor, 8 GB RAM, and NVIDIA GeForce GTX 1060 graphics processing unit (GPU), running on a 64-bit Ubuntu 16.04 operating system with Python.

We trained all CNNs based on the transfer learning strategy in 20 epochs with a batch size of 64. We used the Adam optimizer with a learning rate of 0.001. For evaluation, we used the stratified hold-out method: 60% for training, 20% for validation, and 20% for the test. To validate the efficiency of the proposed systems, we used four evaluation metrics, namely: accuracy, recall, precision, and F1-score.

A. Data compilation

The emergence of the COVID-19 pandemic has led researchers to propose voluminous datasets. To the best of the authors' knowledge, all voluminous datasets such as COVID-19 Radiography Database[1], COVID-QU-Ex [27], and COVIDx [2] classify CXR images into three classes: Normal, Pneumonia, and COVID-19. In this context, the pneumonia class groups viral, bacterial, and other pneumonia variants.

It is crucial to distinguish between bacterial and viral pneumonia since they require different forms of treatment. Bacterial pneumonia can be treated with antibiotics, while viral pneumonia requires supportive care [28]. Therefore, our study aims to distinguish between COVID-19 and non-COVID-19 pneumonia and classify bacterial and viral pneumonia. To generate our multi-source dataset COVID-QU-Ex_4C, we used three publicly available datasets: COVID-QU-Ex [3] [27], Chest-Xray-Pneumonia [4] [28], and Qatacov19 [5].

Table I provides an overview of the proposed COVID-QU-Ex_4C dataset. The dataset includes normal and COVID-19 instances selected from the COVID-QU-Ex dataset, as well as bacterial and viral classes obtained by

[1] https://www.kaggle.com/tawsifurrahman/covid19-radiography-database
[2] https://github.com/lindawangg/COVID-Net/blob/master/docs/COVIDx.md
[3] https://www.kaggle.com/datasets/anasmohammedtahir/covidqu
[4] https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia
[5] https://www.kaggle.com/datasets/aysendegerli/qatacov19-dataset

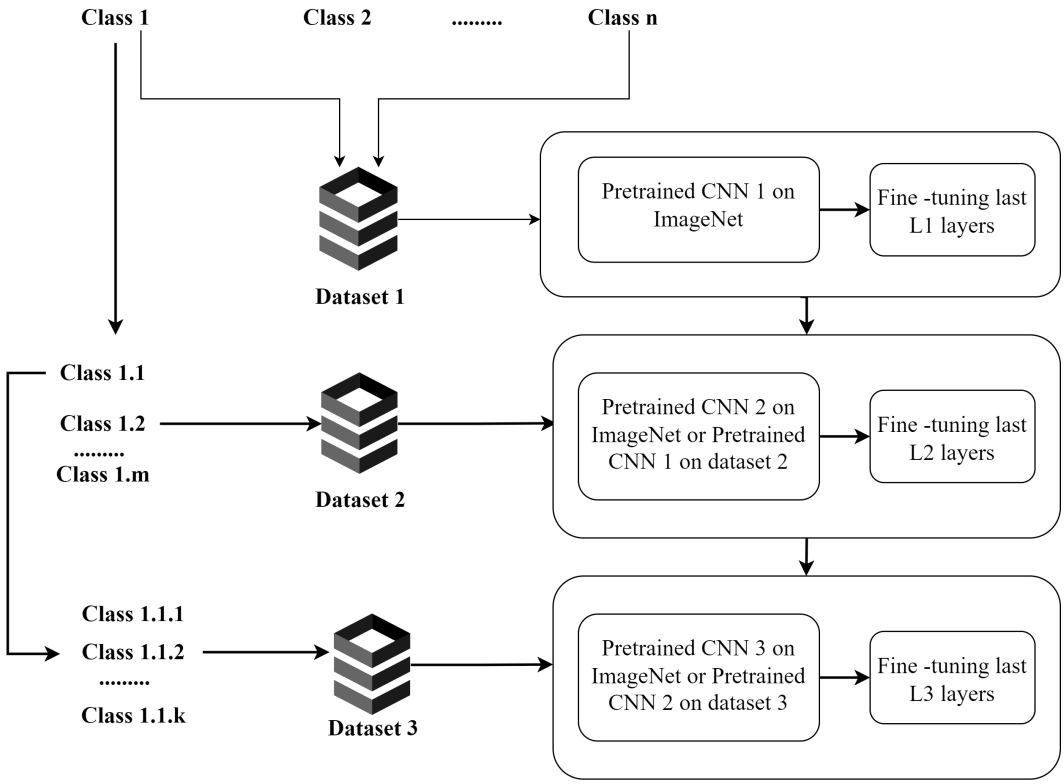

Figure 1. The proposed cascade strategy.

merging bacterial and viral images from the Chest-Xray-Pneumonia and Qata-cov19 (Control Group II) datasets.

TABLE I. The compiled COVID-QU-Ex_4C dataset.

Source	Class	Number	Total
COVID-QU-Ex	Covid-19 Normal	11 956 10 701	31 175
Chest-Xray-Pneumonia + Qata-cov19 (Controle Group 2	Bacterial Viral	5540 2978	

Table II illustrates the structure of the sub-datasets used for training the three cascade systems, namely PCN_BV, PN_CBV, and PN_PC_BV. The first two-stage cascade system, PCN_BV, requires two sub-datasets, namely PCN_BV_D1 and PCN_BV_D2, where PCN_BV_D2 contains sub-categories (viral and bacterial) of the Non-Covid-19 Pneumonia class of PCN_BV_D1. Similarly, the second system PN_CBV also uses two sub-datasets, namely PN_CBV_D1 and PN_CBV_D2. PN_CBV_D2 is a subset of PN_CBV_D1 and contains the pneumonia sub-classes (COVID-19, bacterial, and viral). Beside, the three-stage cascade system PN_CBV_D1 requires three sub-datasets: PN_CBV_D1, PN_PC_BV_D2, and PCN_BV_D2. PN_CBV_D1 and PCN_BV_D2 are also present in the first two cascade systems, while PN_PC_BV_D2 is a new sub-dataset that differentiates COVID-19 from other non-COVID-19 pneumonias.

B. Results of the two-cascade strategies

This section presents the results of the two-stage cascade schemes: PCN_BV and PN_CBV. Firstly, we present the results obtained on the sub-datasets, namely PCN_BV_D1 and PCN_BV_D2 for PCN_BV, and PN_CBV_D1 and PN_CBV_D2 for PN_CBV. Secondly, we perform a comparative study between PCN_BV and PN_CBV to identify the most efficient architecture.

Table III illustrates the results obtained on the PCN_BV_D1 and PCN_BV_D2 datasets. We used transfer learning from the ImageNet dataset as an initial step for both PCN_BV_D1 and PCN_BV_D2, then, we performed transfer learning from PCN_BV_D1 to

TABLE II. The generated sub-datasets for cascade systems.

Scheme	Dataset	Class	Number
PCN_BV	PCN_BV_D1	Non Covid-19	8518
		Pneumonia	
		Covid-19	11 956
		Normal	10 701
	PCN_BV_D2	Bacterial	5540
		Viral	2978
PN_CBV	PN_CBV _D1	Pneumonia	20 474
		Normal	10 701
	PN_CBV_D2	Covid-19	11 956
		Bacterial	5540
		Viral	2978
PN_PC_BV	PN_CBV_D1	-	-
	PN_PC_BV_D2	Non Covid-19	8518
		Pneumonia	
		Covid-19	11 956
	PCN_BV_D2	-	-

PCN_BV_D2.

For the second cascade system PN_CBV, obtained results on both PN_CBV_D1 and PN_CBV_D2 as presented in table IV.

Table V presents the results obtained from the two-stage cascade methods: PCN_BV and PN_CBV. It highlights all possible combinations between the models generated in the previous step. At each level, we fixed the first model in the first stage and varied the second model in the second stage. For instance, VGG16_VGG16 indicates that the VGG16 network was used in both levels, while VGG16_VGG16t indicates that the VGG16t model was obtained through transfer learning from the used VGG16 model in the first stage.

The comparative study demonstrates that transfer learning from the first-stage models in the cascade strategy is generally more efficient than transfer learning from ImageNet, except for DenseNet201 and DenseNet121. Besides, for transfer learning from ImageNet, DenseNet201 was more efficient for both cascade techniques, while DenseNet121 was more promising for the PN_CBV method.

The results indicate that the models were less efficient when used independently compared to their combination with other second-stage models. These findings highlight the advantages of using cascade strategies, except for a few cases (22 out of 84). Surprisingly, when combined with other second-stage models using the PN_CBV method, all results decreased for Xception, which we believe is due to its low accuracy (96.31%) on the PN_CBV_D1 dataset.

The comparative study between the two-stage cascade methods PCN_BV and PN_CBV demonstrated the efficiency of PCN_BV for Xception, VGG16, InceptionV3, and DenseNet121. Specifically, accurate results were obtained for VGG16, InceptionV3, and DenseNet121 using PCN_BV, except when combined with DenseNet121 in all cases and with DenseNet201 for DenseNet121 and InceptionV3. In constract, PN_CBV was more accurate for VGG19 and DenseNet201, except when combining VGG19 with VGG16 or VGG19t and DenseNet201 with VGG16.

C. Results of the three-cascade strategy

For the two-stage cascade strategies, we generated 42 combinations for each method. However, for the three-stage cascade method, a high number of combinations can be generated between the CNN architectures (294), which can be computationally expensive. Therefore, to reduce the number of combinations, we employed two strategies to select the appropriate model at each level. The first strategy combines CNNs based on transfer learning from the previous stage in a cascade strategy, where a unique architecture is used for each combination. The second strategy selects the two best models on the appropriate dataset at each level. For the first stage, we selected DenseNet121 and DenseNet201 due to their high accuracy on the PN_CBV_D1 dataset. In the second stage, we chose DenseNet121 and Xception, and in the third stage, we selected DenseNet201 and InceptionV3.

Table VI presents the obtained results based on the two strategies. In the first technique, DenseNet201 demonstrated its efficiency compared to other combinations. However, the results obtained by Xception were less promising. Besides, for the second strategy, the comparative study, shows that using Densenet201 in the first stage is more promising than DenseNet121. Additionally, in the second stage, DenseNet121 is more efficient than Xception. Overall, the ensemble's efficiency depends on the models' performance at the first and second stages. In general, the best networks ensure the best combination.

D. Comparison and discussion

In the previous section, we conducted a comprehensive comparative study between obtained results on the COVID-QU-Ex_4C and its sub-datasets. Table VII highlights the best strategy for each dataset, and overall, the DenseNets architectures yielded the best results. The obtained results demonstrate the challenges of distinguishing between pneumonia (92.31%) in both binary (bacterial and viral) and multi-class (COVID-19, bacterial, and viral) systems. Whereas, CNNs accurately classified almost all images for datasets that classify pneumonia and COVID-19 (99.88%) or pneumonia and normal images (98.80%). These results confirm the advantages of using a cascade system to separate the four-class classification task based on annotation complexity, starting from the easiest to the most complex.

The comparative analysis of two-stage cascade systems indicates that PCN_BV is more suitable for Xception, VGG16, InceptionV3, and DenseNet121 models in the

TABLE III. The obtained results on the PCN_BV_D1 and PCN_BV_D2 datasets in terms of accuracy (Acc), precision (P), recall (R), and F-measure (F1).

Dataset	PCN_BV_D1 (%)				PCN_BV_D2 (%)							
	Transfer from ImageNet				Transfer from ImageNet				Transfer from PCN_BV_D1			
	Acc	P	R	F1	Acc	P	R	F1	Acc	P	R	F1
VGG16	97.88	97.91	97.91	97.91	91.43	91.29	89.68	90.39	92.25	91.53	91.40	91.46
VGG19	97.23	97.11	97.39	97.24	89.67	89.17	87.86	88.45	92.31	92.05	90.90	91.43
InceptionV3	97.23	97.29	97.36	97.31	91.96	91.80	90.36	91.01	92.31	91.61	91.45	91.53
Xception	97.50	97.41	97.66	97.52	91.78	91.24	90.57	90.89	91.84	91.38	90.54	90.94
DenseNet121	98.11	98.06	98.19	98.12	90.79	90.23	89.34	89.75	91.84	90.97	91.12	91.04
DenseNet201	97.96	97.92	98.04	97.98	92.14	91.60	91.00	91.29	91.37	90.83	90.06	90.42

TABLE IV. The obtained results on the PN_CBV_D1 and PN_CBV_D2 datasets in terms of accuracy (Acc), precision (P), recall (R), and F-measure (F1).

Dataset	PN_CBV_D1 (%)				PN_CBV_D2 (%)							
	Transfer from ImageNet				Transfer from ImageNet				Transfer from PN_CBV_D1			
	Acc	P	R	F1	Acc	P	R	F1	Acc	P	R	F1
VGG16	97.76	97.55	97.47	97.51	95.78	93.67	91.50	92.37	95.95	93.25	92.82	93.03
VGG19	97.53	97.45	97.06	97.25	95.76	92.45	93.10	92.75	95.76	92.68	92.87	92.77
InceptionV3	97.11	96.96	96.62	96.79	96.29	93.97	93.13	93.51	96.71	94.60	93.86	94.19
Xception	96.31	96.75	95.08	95.84	96.24	93.30	93.68	93.48	96.34	93.83	93.26	93.52
DenseNet121	98	97.99	97.56	97.77	96.73	94.02	94.84	94.40	95.76	93.19	92.01	92.50
DenseNet201	98.22	98.28	97.76	98.02	96.80	95.05	93.81	94.34	96.10	93.22	93.15	93.18

first stage, where it was less accurate in 16 among 42 cases. On the other hand, PN_CBV was more accurate for both VGG19 and DenseNet201. This method achieved the best result by DenseNet201_DenseNet121 architecture. Overall, these findings suggest that we cannot assume that PCN_BV presents the best strategy, as each technique has its advantages depending on the specified architecture.

Figure 2 compares the results obtained from transfer learning from ImageNet and transfer learning within the three cascade systems: PCN_BV, PN_CBV, and PN_PC_BV. In general, transfer learning within the cascade systems based on PCN_BV was more promising, except for DenseNets, where transfer learning from ImageNet based on PN_CBV yielded accurate results. The comparative study between the used methods for transfer learning within the cascade systems also highlights the efficiency of PCN_BV over both PN_CBV and PN_PC_BV. In conclusion, the transfer learning strategy within the two-stage cascade system was not suitable for DenseNets and was less efficient for the three-stage cascade systems.

Figure 3 presents a comparative analysis of the best combinations for each network and strategy. For the PCN_BV strategy, the most accurate results were obtained by merging models with their transferred versions and employing DenseNet201 in the second stage. The transfer learning within the cascade systems was observed to be more promising for the PCN_BV strategy. Conversely, for PN_CBV, combining models with DenseNet121 was more accurate. Based on these findings, we propose that using transfer learning from multi-class classification problems in two-stage cascade systems is more promising.

Figure VIII displays the best results for each strategy, highlighting the efficiency of two-stage cascade systems compared to the three-stage cascade system. It's worth noting that while PCN_BV and PN_PC_BV achieved the same result, PCN_BV is more storage-efficient as it only requires storing two models instead of three for prediction. . According to the best results of two-stage cascade systems, we can conclude that the performance of the first-stage models influences the accuracy of the two-stage cascade strategies. For PCN_BV, DenseNet121_DenseNet121t achieved the best result due to the high performance of DenseNet121 in the first stage on the PCN_BV_D1 dataset. Similarly, for PN_CBV, the interesting results of DenseNet201 in the first stage generated the best combination (DenseNet201_DenseNet121).

5. Conclusion

In this study, we proposed three cascade systems based on six CNN architectures for COVID-19 classification: two types of two-stage cascade strategies (PCN_BV and PN_CBV) and a three-stage cascade system (PN_PC_BV). To improve results, we integrated transfer learning strategies within the proposed systems. We validated the proposed methods on a newly generated dataset (COVID-QU-Ex_4C) that contains four classes: COVID-19, normal, bacterial, and viral. The comparative study showed that the two-stage cascade systems were more efficient than the three-stage cascade system. Furthermore,

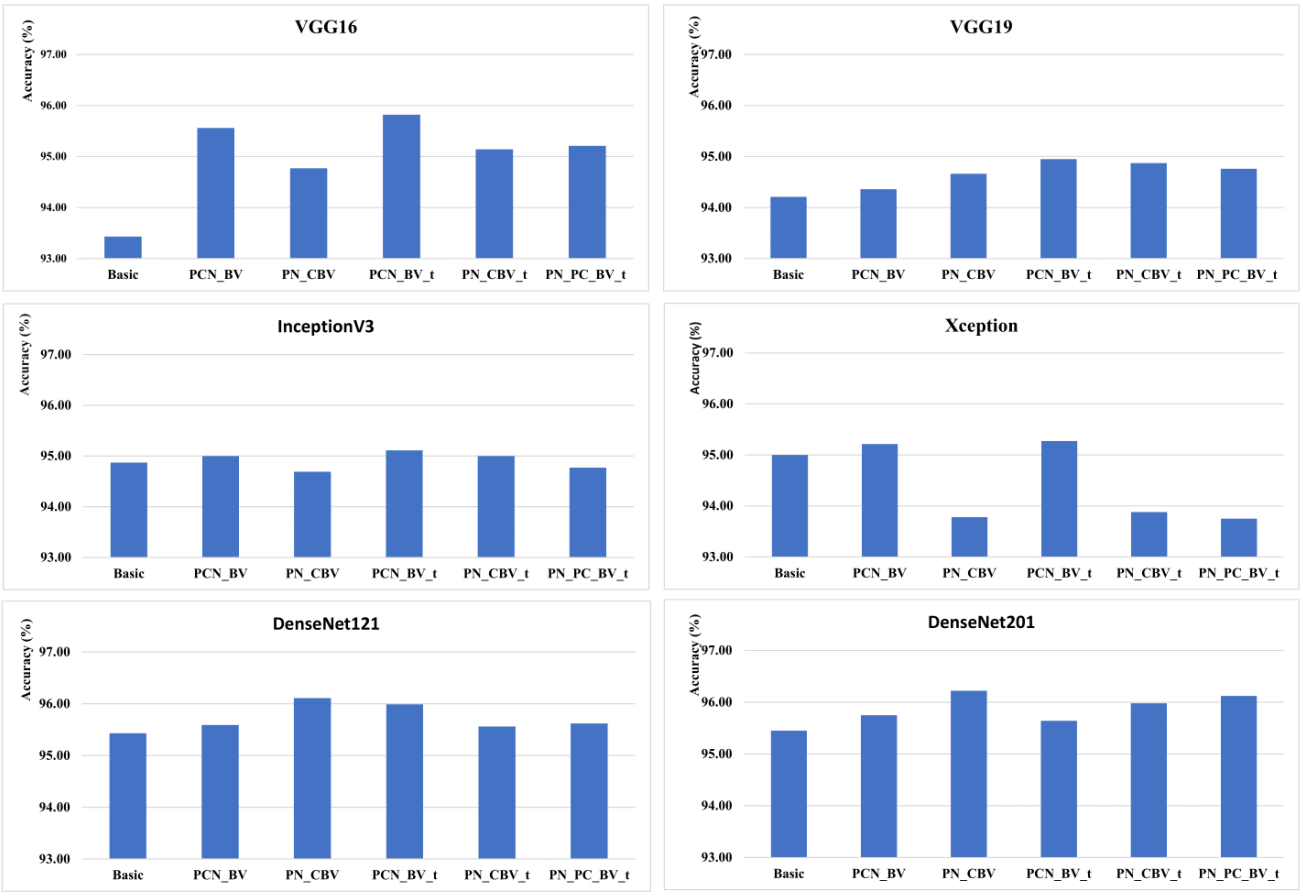

Figure 2. Comparison between transfer learning from ImageNet and transfer learning within the three cascade systems.

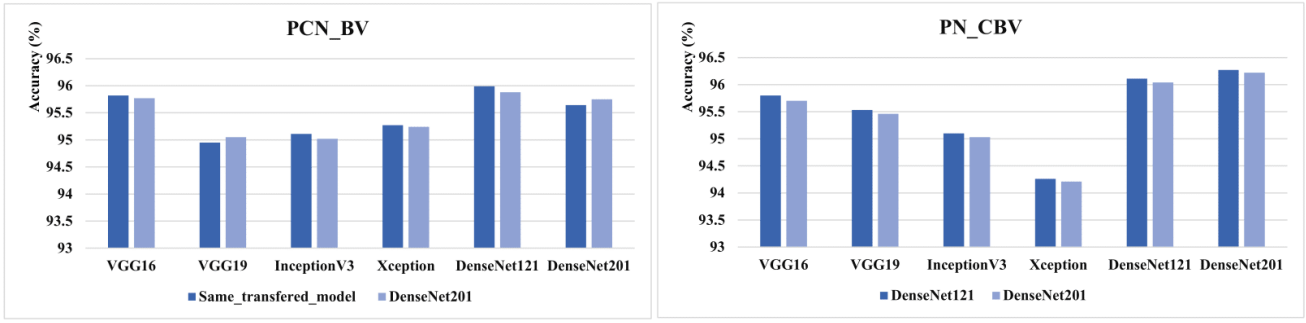

Figure 3. The two best combinations for each network and strategy.

TABLE V. The obtained results on the two-stage cascade methods (PCN_BV, PN_CBV) in terms of accuracy (Acc), precision (P), recall (R), and F-measure (F1).

First Network	Hybridization	PCN_BV (%)				PN_CBV (%)			
		Acc	P	R	F1	Acc	P	R	F1
VGG16	VGG16	93.43	89.01	90.74	89.80	-	-	-	-
	VGG16_VGG16	95.56	93.81	92.68	93.16	94.77	93.05	91.09	91.80
	VGG16_VGG16t	95.82	93.93	93.60	93.76	95.14	93.12	92.57	92.82
	VGG16_VGG19	95.13	92.72	91.95	92.29	94.87	92.19	92.74	92.45
	VGG16_InceptionV3	95.72	94.11	93	93.48	95.35	93.34	92.89	93.08
	VGG16_Xception	95.61	93.64	93	93.29	95.27	92.81	93.13	92.97
	VGG16_DenseNet121	95.37	93.10	92.47	92.76	95.80	93.69	94.47	94.06
	VGG16_DenseNet201	95.77	93.92	93.44	93.66	95.70	94.25	93.55	93.85
VGG19	VGG19	94.21	90.50	92.48	91.26	-	-	-	-
	VGG19_VGG19	94.36	91.18	91.69	91.41	94.66	91.82	92.71	92.23
	VGG19_VGG19t	94.95	92.23	92.95	92.57	94.87	92.38	92.59	92.48
	VGG19_VGG16	94.87	92.49	92.62	92.51	94.53	92.67	90.94	91.54
	VGG19_InceptionV3	95	92.50	92.86	92.65	95.11	92.86	92.77	92.78
	VGG19_Xception	94.92	92.20	92.91	92.54	95.05	92.49	93.03	92.75
	VGG19_DenseNet121	94.68	91.78	92.41	92.08	95.53	93.03	94.30	93.61
	VGG19_DenseNet201	95.05	92.40	93.26	92.82	95.46	93.67	93.44	93.52
InceptionV3	InceptionV3	94.87	92.80	93.01	92.90	-	-	-	-
	InceptionV3_InceptionV3	95	93.24	92.85	93	94.69	92.49	92.56	92.50
	InceptionV3_InceptionV3t	95.11	93.13	93.46	93.29	95	92.82	93.17	92.97
	InceptionV3_VGG16	94.79	92.96	92.41	92.62	94.10	92.29	90.69	91.23
	InceptionV3_VGG19	94.33	91.75	91.60	91.64	94.28	91.54	92.52	91.99
	InceptionV3_Xception	94.94	92.90	92.94	92.90	94.63	92.07	92.80	92.43
	InceptionV3_DenseNet121	94.68	92.38	92.40	92.37	95.10	92.70	94.07	93.33
	InceptionV3_DenseNet201	95.02	92.98	93.15	93.05	95.03	93.32	93.19	93.22
Xception	Xception	95	93.17	92.92	93	-	-	-	-
	Xception_Xception	95.21	92.73	93.24	92.97	93.78	90.54	92.26	91.31
	Xception_Xceptiont	95.27	92.89	93.29	93.07	93.88	90.13	92.14	91.01
	Xception_VGG16	95.06	92.79	92.70	92.69	93.27	90.84	90.16	90.22
	Xception_ VGG19	94.60	91.62	91.89	91.73	93.36	89.80	91.87	90.69
	Xception_InceptionV3	95.22	92.86	93.03	92.91	93.85	90.77	92.01	91.31
	Xception_DenseNet121	94.94	92.22	92.66	92.43	94.26	90.91	93.53	92
	Xception_DenseNet201	95.24	92.77	93.34	93.04	94.21	91.39	92.69	91.96
DenseNet121	DenseNet121	95.43	92.56	93.05	92.80	-	-	-	-
	DenseNet121_DenseNet121	95.59	93.03	93.14	93.07	96.11	93.76	94.74	94.22
	DenseNet121_DenseNet121t	95.99	93.47	94.27	93.85	95.56	93.18	93.04	93.07
	DenseNet121_VGG16	95.74	93.62	93.23	93.37	95.05	93.18	91.33	91.98
	DenseNet121_VGG19	95.21	92.33	92.27	92.27	95.16	92.28	93.06	92.64
	DenseNet121_InceptionV3	95.86	93.70	93.46	93.54	95.66	93.33	93.23	93.25
	DenseNet121_Xception	95.80	93.47	93.57	93.51	95.62	93.05	93.56	93.30
	DenseNet121_DenseNet201	95.88	93.56	93.82	93.68	96.04	94.34	93.93	94.09
DenseNet201	DenseNet201	95.45	92.75	93.71	93.20	-	-	-	-
	DenseNet201_DenseNet201	95.75	93.50	93.69	93.59	96.22	94.57	93.97	94.21
	DenseNet201_DenseNet201t	95.64	93.09	93.40	93.24	95.98	93.60	93.85	93.72
	DenseNet201_VGG16	95.61	93.55	93.10	93.27	95.26	93.50	91.45	92.18
	DenseNet201_VGG19	95.11	92.29	92.21	92.22	95.42	92.65	93.26	92.93
	DenseNet201_InceptionV3	95.75	93.69	93.38	93.49	95.83	93.57	93.28	93.39
	DenseNet201_Xception	95.69	93.38	93.51	93.43	95.75	93.18	93.49	93.33
	DenseNet201_DenseNet121	95.48	92.94	93.06	92.99	96.27	93.84	94.79	94.28

the results of the two-stage cascade systems revealed that PCN_BV was more accurate in most cases, and transfer learning within the cascade systems was more effective for PCN_BV than PN_CBV.

As a future perspective, we plan to introduce more variability into the proposed dataset and address the data imbalance issue. We propose to explore additional data augmentation techniques, such as generative adversarial networks (GANs), to tackle this problem. Additionally, we suggest introducing Gradient-weighted Class Activation Mapping (Grad-CAM) to visualize activated features during detection.

TABLE VI. The obtained results based on the three-stage cascade strategy PN_PC_BV in terms of accuracy (Acc), precision (P), recall (R), and F-measure (F1).

Network	Acc (%)	P (%)	R (%)	F1 (%)
VGG16_VGG16t_VGG16t	95.21	92.97	92.74	92.84
VGG19_VGG19t_VGG19t	94.76	92.22	92.14	92.15
InceptionV3_InceptionV3t_InceptionV3t	94.77	92.74	92.70	92.68
Xception_Xceptiont_Xceptiont	93.75	90.94	91.62	91.16
DenseNet121_DenseNet121t_DenseNet121t	95.62	92.90	93.71	93.29
DenseNet201_DenseNet201t_DenseNet201t	96.12	93.86	94.08	93.96
Densenet201_ Densenet121_DenseNet201	95.99	93.63	93.83	93.72
Densenet201_ Xception_DenseNet201	95.96	93.64	93.77	93.70
Densenet201_ Densenet121_InceptionV3	95.98	93.79	93.47	93.59
Densenet201_ Xception_InceptionV3	95.94	93.80	93.42	93.56
Densenet121_ Densenet121_DenseNet201	95.78	93.49	93.71	93.59
Densenet121_ Xception_DenseNet201	95.75	93.47	93.65	93.55
Densenet121_ Densenet121_InceptionV3	95.80	93.67	93.42	93.51
Densenet121_ Xception_InceptionV3	95.77	93.64	93.36	93.46

TABLE VII. The best results for each dataset.

Dataset	Method	Accuracy(%)
COVID-QU-Ex_4C	DenseNet201	95.45
PCN_BV_D1	DenseNet121	98.11
PCN_BV_D2	VGG19, InceptionV3	92.31
PN_CBV_D1	DenseNet201	98.22
PN_CBV_D2	DenseNet201	96.80
PN_PC_BV_D2	VGG16	99.88

TABLE VIII. The best results of the cascade strategies.

Strategy	Accuracy (%)
PCN_BV	95.99
PN_CBV	96.27
PN_PC_BV	95.99

References

[1] D. Cucinotta and M. Vanelli, "Who declares covid-19 a pandemic," *Acta Bio Medica: Atenei Parmensis*, vol. 91, no. 1, p. 157, 2020.

[2] F. Yu, L. Du, D. M. Ojcius, C. Pan, and S. Jiang, "Measures for diagnosing and treating infections by a novel coronavirus responsible for a pneumonia outbreak originating in wuhan, china," *Microbes and infection*, vol. 22, no. 2, pp. 74–79, 2020.

[3] T. Ai, Z. Yang, H. Hou, C. Zhan, C. Chen, W. Lv, Q. Tao, Z. Sun, and L. Xia, "Correlation of chest ct and rt-pcr testing in coronavirus disease 2019 (covid-19) in china: a report of 1014 cases," *Radiology*, 2020.

[4] H. L. Fred, "Drawbacks and limitations of computed tomography: views from a medical educator," *Texas Heart Institute Journal*, vol. 31, no. 4, p. 345, 2004.

[5] L. Brunese, F. Martinelli, F. Mercaldo, and A. Santone, "Machine learning for coronavirus covid-19 detection from chest x-rays," *Procedia Computer Science*, vol. 176, pp. 2212–2221, 2020.

[6] S. Bharati, P. Podder, M. Mondal, and V. Prasath, "Medical imaging with deep learning for covid-19 diagnosis: a comprehensive review," *arXiv preprint arXiv:2107.09602*, 2021.

[7] V. Gupta, N. Jain, J. Sachdeva, M. Gupta, S. Mohan, M. Y. Bajuri, and A. Ahmadian, "Improved covid-19 detection with chest x-ray images using deep learning," *Multimedia Tools and Applications*, vol. 81, no. 26, pp. 37 657–37 680, 2022.

[8] E. Jangam, C. S. R. Annavarapu, and A. A. D. Barreto, "A multi-class classification framework for disease screening and disease diagnosis of covid-19 from chest x-ray images," *Multimedia Tools and Applications*, pp. 1–35, 2022.

[9] P. Sharma, R. Arya, R. Verma, and B. Verma, "Conv-capsnet: capsule based network for covid-19 detection through x-ray scans," *Multimedia Tools and Applications*, pp. 1–25, 2023.

[10] H. Kör, H. Erbay, and A. H. Yurttakal, "Diagnosing and differentiating viral pneumonia and covid-19 using x-ray images," *Multimedia Tools and Applications*, vol. 81, no. 27, pp. 39 041–39 057, 2022.

[11] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks," *Pattern Analysis and Applications*, vol. 24, no. 3, pp. 1207–1220, 2021.

[12] M. E. Chowdhury, T. Rahman, A. Khandakar, R. Mazhar, M. A. Kadir, Z. B. Mahbub, K. R. Islam, M. S. Khan, A. Iqbal, N. Al Emadi *et al.*, "Can ai help in screening viral and covid-19 pneumonia?" *IEEE Access*, vol. 8, pp. 132 665–132 676, 2020.

[13] T. Rahman, A. Khandakar, Y. Qiblawey, A. Tahir, S. Kiranyaz, S. B. A. Kashem, M. T. Islam, S. Al Maadeed, S. M. Zughaier, M. S. Khan *et al.*, "Exploring the effect of image enhancement techniques on covid-19 detection using chest x-ray images," *Computers in biology and medicine*, vol. 132, p. 104319, 2021.

[14] R. Jain, M. Gupta, S. Taneja, and D. J. Hemanth, "Deep learning based detection and analysis of covid-19 on chest x-ray images," *Applied Intelligence*, vol. 51, no. 3, pp. 1690–1700, 2021.

[15] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao, J. Guo, M. Cai, J. Yang, Y. Li, X. Meng *et al.*, "A deep learning algorithm using ct images

to screen for corona virus disease (covid-19)," *European radiology*, vol. 31, no. 8, pp. 6096–6104, 2021.

[16] N. Dif, M. O. Attaoui, Z. Elberrichi, M. Lebbah, and H. Azzag, "Transfer learning from synthetic labels for histopathological images classification," *Applied Intelligence*, vol. 52, no. 1, pp. 358–377, 2022.

[17] 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*, vol. 140, p. 110245, 2020.

[18] A. M. Ismael and A. Şengür, "Deep learning approaches for covid-19 detection based on chest x-ray images," *Expert Systems with Applications*, vol. 164, p. 114054, 2021.

[19] H. Gunraj, A. Sabri, D. Koff, and A. Wong, "Covid-net ct-2: Enhanced deep neural networks for detection of covid-19 from chest ct images through bigger, more diverse learning," *Frontiers in Medicine*, vol. 8, p. 3126, 2022.

[20] S. R. Nayak, D. R. Nayak, U. Sinha, V. Arora, and R. B. Pachori, "Application of deep learning techniques for detection of covid-19 cases using chest x-ray images: A comprehensive study," *Biomedical Signal Processing and Control*, vol. 64, p. 102365, 2021.

[21] M. Z. Islam, M. M. Islam, and A. Asraf, "A combined deep cnn-lstm network for the detection of novel coronavirus (covid-19) using x-ray images," *Informatics in medicine unlocked*, vol. 20, p. 100412, 2020.

[22] N. Dif, A. Arioui, I. Zeblah, and S. M. Benslimane, "Covid-19 classification from x-rays : A comparative study," in *2022 3rd International Conference on Embedded and Distributed Systems (EDiS)*, 2022, pp. 75–80.

[23] C. Öksüz, O. Urhan, and M. K. Güllü, "Ensemble-cvdnet: A deep learning based end-to-end classification framework for covid-19 detection using ensembles of networks," *arXiv preprint arXiv:2012.09132*, 2020.

[24] K. Y. Win, N. Maneerat, S. Sreng, and K. Hamamoto, "Ensemble deep learning for the detection of covid-19 in unbalanced chest x-ray dataset," *Applied Sciences*, vol. 11, no. 22, p. 10528, 2021.

[25] L. Brunese, F. Mercaldo, A. Reginelli, and A. Santone, "Explainable deep learning for pulmonary disease and coronavirus covid-19 detection from x-rays," *Computer Methods and Programs in Biomedicine*, vol. 196, p. 105608, 2020.

[26] A. Shelke, M. Inamdar, V. Shah, A. Tiwari, A. Hussain, T. Chafekar, and N. Mehendale, "Chest x-ray classification using deep learning for automated covid-19 screening," *SN computer science*, vol. 2, no. 4, pp. 1–9, 2021.

[27] A. M. Tahir, M. E. Chowdhury, A. Khandakar, T. Rahman, Y. Qiblawey, U. Khurshid, S. Kiranyaz, N. Ibtehaz, M. S. Rahman, S. Al-Maadeed *et al.*, "Covid-19 infection localization and severity grading from chest x-ray images," *Computers in biology and medicine*, vol. 139, p. 105002, 2021.

[28] D. S. Kermany, M. Goldbaum, W. Cai, C. C. Valentim, H. Liang, S. L. Baxter, A. McKeown, G. Yang, X. Wu, F. Yan *et al.*, "Identifying medical diagnoses and treatable diseases by image-based deep learning," *Cell*, vol. 172, no. 5, pp. 1122–1131, 2018.