

000
001
002
003
004
005

Lung Disease Classification - Applied AI Final Report

Group-Q

006 Abstract

1. Introduction

Early diagnosis of respiratory diseases like pneumonia and COVID-19 leads to decreased mortality rate [9] and is a powerful way to manage a pandemic [49]. These diseases can be diagnosed using a variety of tests like pulse oximetry, chest x-ray, CT scan [40], PCR [2] however chest X-rays are by far the most accessible [13] to low and middle income countries. Furthermore, the scan is available in minutes making it one of the fastest ways of diagnosis [41]. However, the bottleneck with this method is the need for an expert radiologists to evaluate the scan [31]. Many researchers have tried to solve this problem by creating a deep learning based lung disease classification system [45] but haven't been able to come up with models that can replace radiologists. Small [16] and highly imbalanced data [45], along with varying specifications of X-ray scanners leading to low inter-hospital accuracy [38] are the biggest problems that researchers have faced. Another issue with using deep neural networks in medical settings is its black-box nature [5], doctors and patients will not trust a model that cannot explain its results [32].

This project is an attempt to compare three CNN backbone architectures namely, ResNet-34, MobileNet V3 Large and EfficientNet B1 along with three lung disease datasets to identify the type of architecture that works best for lung disease classification. Two

of the datasets used presented a multiclass classification problem with 3 classes while the third dataset presented a multiclass, multilabel classification problem. A total of 12 models were trained in this study, four for each of the three datasets. The first three models for each dataset was trained from scratch and the fourth model was trained using transfer learning. Transfer learning was performed by deep-tuning ImageNet weights and the performance was evaluated to check improvement over the models trained from scratch. The small dataset problem and the issue of different radiographic contrast [33] is mitigated using data augmentation. Imbalanced data problem is handled by undersampling the majority class. The hyperparameters were fixed across models and the F1 scores and cross entropy loss have been used to compare models and select the best overall model. All the models were optimized using the Adam optimizer [24] with default parameters and the cosine annealing [30] learning rate scheduler was used to decrease the learning rate as training progressed. Further, an ablation study was performed to find the best learning rate for the selected model. Finally, GradCAM [15] and T-SNE were used to visualize the trained models and understand model predictions better. An F1 score of 0.8 and 0.98 was achieved for the two multiclass datasets, whereas the maximum F1 for the multilabel dataset with 7 classes was 0.46.

Related Works: Li *et al.* [27] were among the first to use CNNs in a medical setting. They used a single convolutional layer to classify interstitial lung diseases using CT scans, achieving better performance than existing approaches. Since then there has been a dramatic increase in application of CNNs in healthcare, deep neural networks have been used to perform various tasks like segmenting regions of interest in MRI [10, 22], classifying X-Ray [37], MRI [12], and CT [3] scans. Further, GANs have been used to generate high quality scans [29] when there is a lack of available data due to either privacy reasons or availability of subjects. GANs have also been used to generate high quality CT scans from MRI scans [28]. Apart from radiographic scans, deep CNNs have also been used to detect malarial parasite in blood smear images [44] with an accuracy of 99.96%. Another interesting application is the use of 1-D convolutions to detect heart anomalies using ECG data [25]. Researchers have also used archi-

108	Dataset	No. of Images	Classes	Size
109	COVID [7, 8, 36]	3.6k:3.6k:1.3k	3	299 ²
110	Pneumonia [23, 43]	3k:1.5k:1.5k	3	224 ²
111	Chest X-Ray8 [46, 47]	7.2k:7k:7k:4.1k :3.9k:3.5k:2.9k	7	1024 ²

Table 1. Shortlisted Datasets.

lectures like the Inception V3 to perform dermatologist level skin-cancer detection using skin lesion images [11] using transfer learning.

In the recent years, many researchers have tried to predict lung diseases using deep CNNs, Wang *et al.* [46] used state of the art backbone architectures to train a lung disease classifier for multilabel data by training only the prediction and transition layers from scratch and leaving pre-trained ImageNet weights freezed while training. They achieved a high AUC of over 0.6 for most of the classes in the dataset with this technique. Rajpurkar *et al.* [37] created a 121 layer deep CNN - CheXNet to detect pneumonia using chest X-rays with radiologist level accuracy. Labhane *et al.* [26] used transfer learning with state of the art backbone architectures like VGG16, VGG19 and InceptionV3 to predict pneumonia in pediatric patients and achieved an F1 score of 0.97. Islam *et al.* combined CNN and LSTM to create a COVID-19 detector [20]. The CNN extracted complex features from scans and the LSTM was used as a classifier. This method resulted in an improvement over a vanilla CNN network and an F1 score of 98.9% was achieved. Abbas *et al.* [1] created the DeTraC network to detect COVID in chest X-rays that improved performance of existing backbone models significantly with the highest accuracy of 98.23% using the VGG19 architecture. Guefrechi *et al.* [16] on the other hand used data augmentation techniques like random rotation, flipping and noise with transfer learning on backbone architectures like ResNet50, InceptionV3 and VGG16 to achieve a high accuracy of 98.3%.

In the following sections methodology of the approach and the results will be discussed.

2. Methodology

Datasets: (Tab. 1) with varying disease types were chosen to ensure model robustness. Other criteria included the *number of images per class* and *image quality* as noisy scans can lead to mis-diagnosis [39].

The **COVID** dataset was created by a team of researchers from Qatar University, Doha, Qatar, and the University of Dhaka, Bangladesh along with collabora-

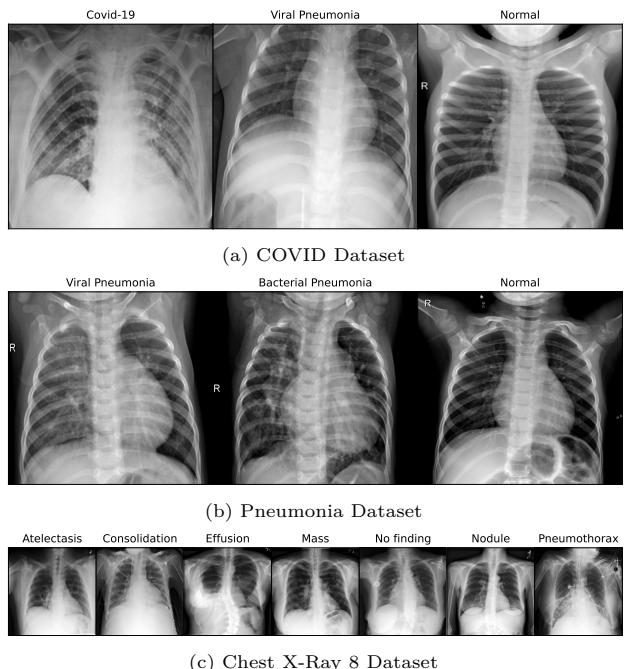


Figure 1. Sample Chest X-rays from the datasets used.

tors from Pakistan and Malaysia in collaboration with medical doctors from the Italian Society of Medical and Interventional Radiology database using 43 different publications [7, 8, 36]. It is a multiclass data with three classes, COVID, viral pneumonia and normal. X-rays with widespread, hazy, and irregular ground glass opacities are of the COVID-19 class [21]. Whereas, the ones with haziness only in the lower regions [50] are viral pneumonia cases as shown in Fig. 1. Chest X-rays of normal lungs provide a clear view of the lungs. The normal class was undersampled to use only 3.6k scans and reduce the data imbalance.

The **Pneumonia**, dataset contains scans from pediatric patients of one to five year olds collected as part of patients' routine clinical care at the Guangzhou Women and Children's Medical Center, Guangzhou, China. [23, 43] This dataset is multiclass with three classes, viral pneumonia, bacterial pneumonia and normal. Scans with one white condensed area affecting only one side of the lungs are tagged as bacterial pneumonia [4] as bacteria tends to aggressively attack one part of the lungs causing inflammation to replace the cells that were otherwise filled with air. On the other hand, X-rays which show bilateral patchy areas of consolidation are classified as viral pneumonia [17] as viruses attack both sides of the lungs producing a homogeneous inflammatory reaction causing mucus and cellular debris. Normal scans here as well produce a clear view of the lungs.

216
217
218
219
220
221
222
223

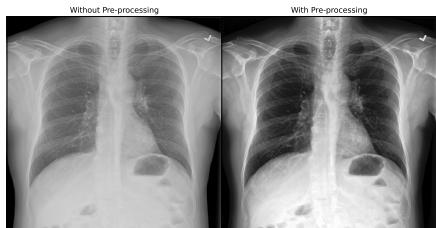


Figure 2. Effect of pre-processing on Chest X-ray images.

NIH [48] released over 100k anonymized chest X-ray images along with their radiological reports from over 30k patients. Wang *et al.* [46] used this data to create the **Chest X-ray 8** dataset by generating disease labels through NLP from the radiological reports. [47] The dataset contains 15 classes but only 7 Fig. 1 were chosen for this study. This dataset is significantly different from the other two as it is a multi-label dataset. Classes were iteratively removed, ensuring that they are not highly imbalanced to finally reach the 7 classes. With over 29,000 images of size 1024 x 1024, this dataset was the biggest and thus had to be resized down to 384 x 384 to reduce training and processing times. Furthermore, normal class images were undersampled by first choosing one scan per patient and then selecting 7000 scans out of this subset randomly. The data consists of multiple scans from the same subject which could lead to data leakage between the train, val and test sets if a random train-test-val split was performed. This was prevented with the use of GroupShuffleSplit from the scikit library.

Before training, all the images were pre-processed using histogram equalization and Gaussian blur with a 5x5 filter as Giełczyk *et al.* [14] showed that this improved the F1 score by about 4% for the chest X-ray classification task. Visually, the contrast of the scan improved and allowed irregularities to stand out as shown in Fig. 2. Next, the scans were divided into train, validation and test with the 70:15:15 split. During training, the scans were augmented using RandomAdjustSharpness and RandomAutocontrast in Pytorch [34] to increase the number of images the model gets to learn from and ensure that the model is robust to scans from different machines. RandomHorizontalFlip was also used to make the models invariant to the direction of the scan as some scans were anterior-posterior while others were posterior-anterior [6].

Backbone Architectures: (Tab. 2) of various configuration and blocks were chosen to ensure that different ideas are tested in this study. Other selection criteria were the *number of trainable parameters*, important to keep track of the total training time, *FLOPS*.

Arch.	Params (Mil.)	Layers	FLOPS (Bil.)	Imagenet Acc.
MobileNet	5.5	18	8.7	92.6
EfficientNet	7.8	25	25.8	94.9
Resnet	21.8	34	153.9	91.4

Table 2. Shortlisted Backbone Architectures.

as models that could easily be deployed on to embedded devices were required and the *top 5 classification accuracy* on the ImageNet 1K benchmark dataset.

ResNet 34 residual learning network with 34 layers that are made possible by skip connections. The 34 layer variant was chosen to decrease training time while not compromising on the accuracy much. This architecture had the highest trainable parameters and FLOPS while the lowest Imagenet accuracy. [18]

MobileNet V3 Large uses depthwise separable convolution from MobileNet V2 along with squeeze-excitation blocks in residual layers from MnasNet. This makes it really quick to train while still performing at par with other architectures. This architecture had the lowest trainable parameters and FLOPS among the three selected. Howard *et al.* [19] also used network architecture search to find the most effective model. The large configuration was chosen to not compromise on the prediction accuracy.

EfficientNet B1 uses compound scaling to scale the model by depth, width and resolution. The B1 version was chosen to have faster training without compromising on the accuracy. [42] This architecture performs the best among the selected on the Imagenet benchmark dataset while having a third of the trainable parameters of Resnet34.

Optimization Algorithm: The Adam optimizer [24] is an adaptive learning rate algorithm which was chosen as the algorithm of choice as it converges faster by integrating benefits of RMSProp and momentum. It is also robust to hyperparameters but, requires tweaking of the learning rate depending on the task at hand. For this study, a learning rate of 0.01 and the author recommend settings for $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$ were used for the first and second order moment estimate as defined in Eq. (1) and Eq. (2) where β_1 and β_2 control the decay rates.

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot q_t \quad (1)$$

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot q_t^2 \quad (2)$$

Further, Cosine annealing [30] learning rate scheduler was used to reduce the learning rate as the training progressed down to a low of 0.001.

324

3. Results

325

Experiment Setup:

First undersampling was performed as described in Sec. 2 on the datasets. Then, the scans were pre-processed using histogram equalization and Gaussian blur before resizing them and storing them in separate directories to make it easier for PyTorch dataloaders. Two datasets in this study presented the multiclass classification problem while the third, chest X-ray 8 dataset presented the multiclass, multilabel classification problem. Thus, the training methodology was separated for these two problems. For the multilabel problem, a softmax layer had to be added before the loss function to get 0 or 1 prediction for all the classes of the data. For this, the BCEWITHLOGITSLOSS function of PyTorch was used as it combines the Sigmoid layer and the BCELoss function in one single class. This makes these operations more numerically stable than their separate counterparts [35]. The backbone architectures were obtained directly from the torchvision library and the final classification layer was modified for the selected datasets. For the models which had to be trained from scratch, the weights were randomly initialized and the entire model was trained for a total of 100 epochs each. For the transfer learning models, the weights were initialized with the IMAGENET1K_V2 weights but the entire model was fine-tuned. The rationale behind performing deep-tuning was that the Imagenet data is very different from chest X-ray scans thus the model would need to learn features from X-ray scans.

356

The batch size was fixed to 32 for all the models. While training the best model by validation loss was saved to prevent the usage of overfit models for test set analysis. The actual and predicted results from each epoch was also stored to calculate the F1 score at each step of training. While calculating the F1 score, macro averaging was used to get an average score across classes. All images were normalized before getting trained with the mean and standard deviation of the training set of each of the selected datasets.

366

Initial runs of the multilabel data training produced 0 F1 score due to the highly imbalanced nature of multilabel classes. To mitigate this, class wise weights were calculated using the training data and used with the loss function. This improved the F1 score considerably.

372

Finally, the best models from each run by validation loss were used to get the test set metrics that are displayed in Tab. 3. Training and validation F1 score and loss are also provided in Fig. 3 and Fig. 4.

376

Main Results:

377

From Fig. 3 it is clear that going from a smaller ar-

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

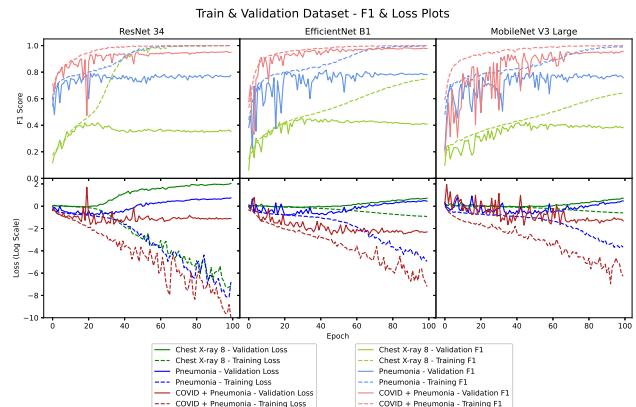


Figure 3. Train & Val, F1 & Loss plots for the 9 models.

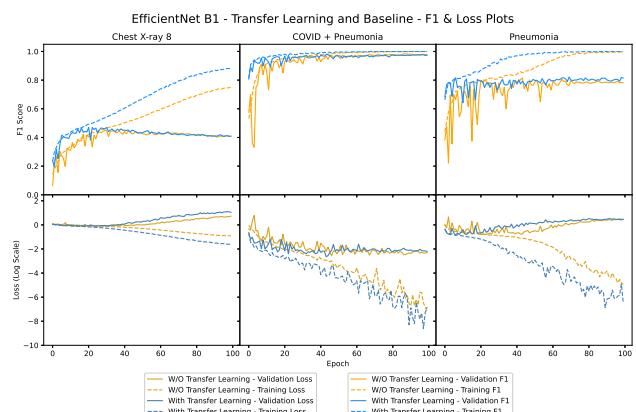


Figure 4. Train & Val, F1 & Loss plots for EfficientNet B1 trained from scratch and with ImageNet weights.

chitecture to a bigger architecture, makes the model start to overfit earlier. The MobileNet model was the most unstable among the three and also took more epochs to reach the minima. The EfficientNet algorithm performs best for the COVID and Chest X-ray 8 dataset and all three architectures performed similar for the pneumonia dataset. This shows that the compound scaline of EfficientNet gives good results for chest X-ray data. The X-ray 8 dataset performed the worst among the three datasets which could be due to the high number of classes, class imbalance and the multilabel nature of the problem. Surprisingly, the pneumonia dataset performed worse than the COVID + pneumonia dataset which indicates that COVID cases are easier to distinguish from pneumonia cases.

In Fig. 4 it can be seen that the transfer learning model had a much better start than the randomly initialized model. It also converged much quicker than the model trained from scratch. For the Pneumonia dataset, the model trained from scratch was highly un-

Model	ResNet			MobileNet			EfficientNet			EN - Transfer Learning		
Dataset	F1	Time	Epoch	F1	Time	Epoch	F1	Time	Epoch	F1	Time	Epoch
Pneumonia	0.784	82	22	0.804	75	42	0.768	110	44	0.782	114	70
COVID	0.959	50	21	0.967	37	44	0.979	56	46	0.978	56	43
X-Ray 8	0.411	11,502	19	0.406	7,275	42	0.445	13,820	31	0.457	13,813	29

Table 3. F1 (higher is better), time per epoch in seconds (lower is better), and number of epochs to reach the best validation loss (lower is better) for the 12 models that were trained.

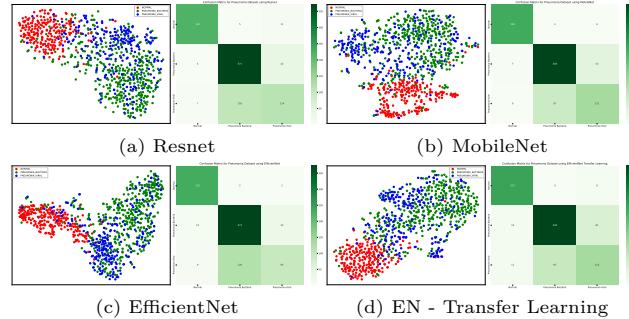


Figure 5. T-SNE and Confusion matrices for the test set of the Pneumonia dataset.

stable at the start and could not catch up to the transfer learning model even after 100 epochs in terms of the F1 score.

Finally, looking at Tab. 3 it can be seen that the MobileNet architecture was the fastest to train per epoch. It consistently took less time per epoch but, if number of epochs required to converge is considered, it does not train the fastest all the time. It is also evident that ResNet converged the fastest at half the number of epochs compared with other models. EfficientNet models perform the best in terms of the overall F1 score on the test set with the exception of the Pneumonia dataset where surprisingly MobileNet performed the best. The transfer learning models converged quicker than the other models with the exception of the Pneumonia dataset. Another surprising observation is that the EfficientNet model takes the longest to train per epoch even though the number of trainable parameters is nowhere close to ResNet. Also, MobileNet isn't as fast to train as expected when compared to ResNet even though it has 4 times the learnable parameters. This could be due to two reasons, depthwise convolutions are not optimized in the version of PyTorch and CUDA used and training is getting CPU bound due to the data augmentation before each training run which would take the same amount of time for all the models.

Fig. 5 shows that the models are able to differentiate well between the normal and pneumonia classes but struggle with the viral pneumonia vs bacterial pneumonia classification. MobileNet performs better but

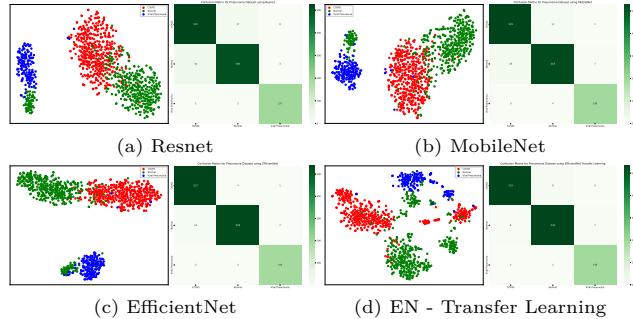


Figure 6. T-SNE and Confusion matrices for the test set of the COVID dataset.

the EfficientNet transfer learning model creates better separation of classes. Thus, even though MobileNet performs better in this case, the EfficientNet transfer learning model would generalize well on new unseen data. This is correlated in the confusion matrix where the transfer learning and MobileNet models perform the best.

Fig. 6 shows that all models do a good job of separating classes to create distinct clusters but, the transfer learning model creates better clusters with separate smaller clusters. These smaller clusters could indicate other factors of the disease, for example the severity and amount of lung damage caused by the disease. This performance of the transfer learning model can be confirmed by looking at the confusion matrix as well.

Fig. 7 shows the gradCAM visualization of the last layer of the convolutional network. Here it can be seen that the ResNet is learning completely different features as compared to the other models. This could be the reason for its low performance. In case of bacterial pneumonia, the network identifies affected area on the right side of the scan. On the other hand, in case of viral pneumonia, models look at both sides of the lungs. This correlates with the actual progression of these diseases as given in Sec. 2.

Fig. 8 shows that MobileNet activates the entire image incase of COVID, this be the reason for its low performance. In case of pneumonia, the EfficientNet models identifies affected areas on the bottom of the lungs. On the other hand, in case of COVID, the mod-

(a) Bacterial Pneumonia

(b) Viral Pneumonia

(c) Normal

Figure 7. GradCAM visualization for the Pneumonia dataset.

(a) COVID

(b) Pneumonia

(c) Normal

Figure 8. GradCAM visualization for the COVID dataset.

els look at a bigger region of the lungs. This correlates with how these diseases impact the lungs as given in Sec. 2.

Ablative Study: For the ablative study, the COVID dataset was chosen along with the EfficientNet B1 architecture trained from scratch. The learning rates chosen for the study are 0.001, 0.005, 0.01, 0.05, and 0.1. From the training and validation F1 score and loss plots given in Fig. 9 it is seen that a very high learning rate of 0.1 is highly unstable and prevents the model from reaching close to global minima. Similarly, learning rate of 0.05 also prevented the model from converging on the validation set even after 100 epochs. The other three learning rates all converged on the validation set but, the learning rate of

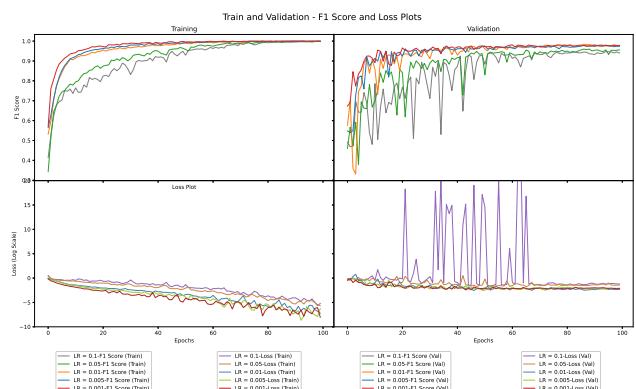


Figure 9. Train & Val, F1 & Loss plots for ablative study models

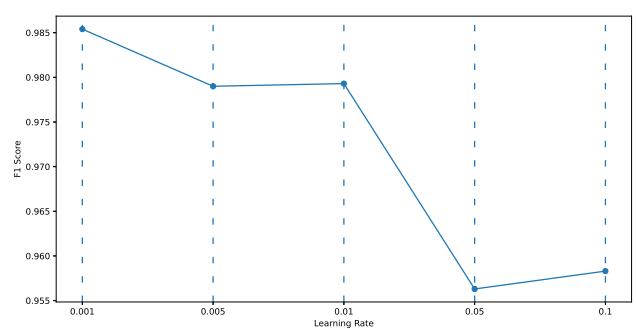


Figure 10. Ablative Study F1 scores (Higher is better).

0.001 was the most stable and reached the highest F1 score earliest. On the other hand, learning rate of 0.01 performed marginally better on the loss plot. From Fig. 10 it can be seen that the best performing learning rate is 0.001 on the F1 score of the test set with 0.005, 0.01 close second and 0.05, 0.1 performing the worst. This matches the results of the validation set on Fig. 9. Thus, a learning rate of 0.001 performs the best on the COVID dataset with the transfer learning EfficientNet model.

648

649
References

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701

- [1] Asmaa Abbas, Mohammed M Abdelsamea, and Mohamed Medhat Gaber. Classification of covid-19 in chest x-ray images using detrac deep convolutional neural network. *Applied Intelligence*, 51(2):854–864, 2021. 2
- [2] Noorullah Akhtar, Jiyuan Ni, Claire Langston, Gail J Demmler, and Jeffrey A Towbin. Pcr diagnosis of viral pneumonitis from fixed-lung tissue in children. *Biochemical and molecular medicine*, 58(1):66–76, 1996. 1
- [3] Wafaa Alakwaa, Mohammad Nassef, and Amr Badr. Lung cancer detection and classification with 3d convolutional neural network (3d-cnn). *International Journal of Advanced Computer Science and Applications*, 8(8), 2017. 1
- [4] How Drugs are Made and Product List. Viral vs. bacterial pneumonia: Understanding the difference. https://www.pfizer.com/news/articles/viral_vs_bacterial_pneumonia_understanding_the_difference, 2020. 2
- [5] Paul J. Blazek. Why we will never open deep learning’s black box. <https://towardsdatascience.com/why-we-will-never-open-deep-learnings-black-box-4c27cd335118>, 2022. 1
- [6] Aleksander Botev, Matthias Bauer, and Soham De. Regularising for invariance to data augmentation improves supervised learning. *arXiv preprint arXiv:2203.03304*, 2022. 3
- [7] Muhammad E. H. Chowdhury, Tawsifur Rahman, Amith Khandakar, Rashid Mazhar, Muhammad Abdul Kadir, Zaid Bin Mahbub, Khandakar Reajul Islam, Muhammad Salman Khan, Atif Iqbal, Nasser Al Emadi, Mamun Bin Ibne Reaz, and Mohammad Tariqul Islam. Can ai help in screening viral and covid-19 pneumonia? *IEEE Access*, 8:132665–132676, 2020. 2
- [8] Muhammad E. H. Chowdhury, Tawsifur Rahman, Amith Khandakar, Rashid Mazhar, Muhammad Abdul Kadir, Zaid Bin Mahbub, Khandakar Reajul Islam, Muhammad Salman Khan, Atif Iqbal, Nasser Al Emadi, Mamun Bin Ibne Reaz, and Mohammad Tariqul Islam. Covid-19 radiography database. <https://www.kaggle.com/datasets/tawsifurrahman/covid19-radiography-database>, 2021. 2
- [9] Priya Daniel, Chamira Rodrigo, Tricia M Mckeever, Mark Woodhead, Sally Welham, and Wei Shen Lim. Time to first antibiotic and mortality in adults hospitalised with community-acquired pneumonia: a matched-propensity analysis. *Thorax*, 71(6):568–570, 2016. 1
- [10] Jose Dolz, Karthik Gopinath, Jing Yuan, Herve Lombaert, Christian Desrosiers, and Ismail Ben Ayed. Hyperdense-net: a hyper-densely connected cnn for multi-modal image segmentation. *IEEE transactions on medical imaging*, 38(5):1116–1126, 2018. 1

- [11] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639):115–118, 2017. 2
- [12] Ammarah Farooq, SyedMuhammad Anwar, Muhammad Awais, and Saad Rehman. A deep cnn based multi-class classification of alzheimer’s disease using mri. In *2017 IEEE International Conference on Imaging systems and techniques (IST)*, pages 1–6. IEEE, 2017. 1
- [13] Guy Frija, Ivana Blažić, Donald P Frush, Monika Hierath, Michael Kawooya, Lluis Donoso-Bach, and Boris Brkljačić. How to improve access to medical imaging in low-and middle-income countries? *EClinicalMedicine*, 38:101034, 2021. 1
- [14] Agata Gielczyk, Anna Marciniak, Martyna Tarczewska, and Zbigniew Lutowski. Pre-processing methods in chest x-ray image classification. *Plos one*, 17(4):e0265949, 2022. 3
- [15] Jacob Gildenblat and contributors. Pytorch library for cam methods. <https://github.com/jacobgil/pytorch-grad-cam>, 2021. 1
- [16] Sarra Guefrechi, Marwa Ben Jabra, Adel Ammar, Anis Koubaa, and Habib Hamam. Deep learning based detection of covid-19 from chest x-ray images. *Multimedia Tools and Applications*, 80(21):31803–31820, 2021. 1, 2
- [17] W Guo, J Wang, M Sheng, M Zhou, and L Fang. Radiological findings in 210 paediatric patients with viral pneumonia: a retrospective case study. *The British journal of radiology*, 85(1018):1385–1389, 2012. 2
- [18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3
- [19] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1314–1324, 2019. 3
- [20] Md Zabirul Islam, Md Milon Islam, and Amanullah Asraf. A combined deep cnn-lstm network for the detection of novel coronavirus (covid-19) using x-ray images. *Informatics in medicine unlocked*, 20:100412, 2020. 2
- [21] Adam Jacobi, Michael Chung, Adam Bernheim, and Corey Eber. Portable chest x-ray in coronavirus disease-19 (covid-19): A pictorial review. *Clinical imaging*, 64:35–42, 2020. 2
- [22] Baris Kayalibay, Grady Jensen, and Patrick van der Smagt. Cnn-based segmentation of medical imaging data. *arXiv preprint arXiv:1701.03056*, 2017. 1
- [23] Daniel Kermany, Kang Zhang, Michael Goldbaum, et al. Labeled optical coherence tomography (oct) and chest x-ray images for classification. *Mendeley data*, 2(2), 2018. 2

- 756 [24] Diederik P Kingma and Jimmy Ba. Adam: A 810
757 method for stochastic optimization. *arXiv preprint* 811
758 *arXiv:1412.6980*, 2014. 1, 3 812
759 [25] Serkan Kiranyaz, Turker Ince, and Moncef Gabbouj. 813
760 Real-time patient-specific ecg classification by 1-d 814
761 convolutional neural networks. *IEEE Transactions on* 815
762 *Biomedical Engineering*, 63(3):664–675, 2015. 1 816
763 [26] Gaurav Labhane, Rutuja Pansare, Saumil Maheshwari, 817
764 Ritu Tiwari, and Anupam Shukla. Detection of 818
765 pediatric pneumonia from chest x-ray images using cnn 819
766 and transfer learning. In *2020 3rd International Conference 820*
767 on Emerging Technologies in Computer Engineering: Machine 821
768 Learning and Internet of Things (ICETCE), pages 85–92. IEEE, 2020. 2 822
769 [27] Qing Li, Weidong Cai, Xiaogang Wang, Yun Zhou, 823
770 David Dagan Feng, and Mei Chen. Medical image 824
771 classification with convolutional neural network. In *2014 825*
772 13th international conference on control automation 826
773 robotics & vision (ICARCV), pages 844–848. IEEE, 827
774 2014. 1 828
775 [28] Yanxia Liu, Anni Chen, Hongyu Shi, Sijuan Huang, 829
776 Wanjia Zheng, Zhiqiang Liu, Qin Zhang, and Xin 830
777 Yang. Ct synthesis from mri using multi-cycle gan for 831
778 head-and-neck radiation therapy. *Computerized Medical 832*
779 Imaging and Graphics, 91:101953, 2021. 1 833
780 [29] Mohamed Loey, Florentin Smarandache, and Nour 834
781 Eldeen M. Khalifa. Within the lack of chest covid-19 835
782 x-ray dataset: a novel detection model based on gan and 836
783 deep transfer learning. *Symmetry*, 12(4):651, 2020. 1 837
784 [30] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic 838
785 gradient descent with warm restarts. *arXiv preprint* 839
786 *arXiv:1608.03983*, 2016. 1, 3 840
787 [31] P Mehrotra, V Bosemani, and J Cox. Do radiologists 841
788 still need to report chest x rays? *Postgraduate medical 842*
789 journal, 85(1005):339–341, 2009. 1 843
790 [32] Aleksandra Mojsilovic. Introducing ai explainability 844
791 360. <https://www.ibm.com/blogs/research/2019/08/ai-explainability-360/>, 2019. 1 845
792 [33] Andrew Murphy. Radiographic contrast. <https://radiopaedia.org/articles/radiographic-contrast>, 2022. 1 846
793 [34] PyTorch. Transforming and augmenting images. 847
794 <https://pytorch.org/vision/stable/transforms.html>. 3 848
795 [35] PyTorch. Bce with logits loss. <https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html>, 2022. 4 849
796 [36] Tawsifur Rahman, Amith Khandakar, Yazan Qiblawey, 850
797 Anas Tahir, Serkan Kiranyaz, Saad Bin Abul Kashem, 851
798 Mohammad Tarique Islam, Somaya Al Maadeed, Susu M. 852
799 Zughayer, Muhammad Salman Khan, and Muhammad E.H. 853
800 Chowdhury. Exploring the effect of image enhancement 854
801 techniques on covid-19 detection using chest x-ray images. 855
802 *Computers in Biology and Medicine*, 132:104319, 2021. 2 856
803 [37] Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon 857
804 Yang, Hershel Mehta, Tony Duan, Daisy Ding, 858
805 [38] Melissa Rohman. Ai performs poorly when 859
806 tested on data from multiple health systems. <https://healthimaging.com/topics/artificial-intelligence/ai-poorly-detects-pneumonia-chest-x-rays>, 2018. 1 860
807 [39] Janaki Sivakumar, K Thangavel, and P Saravanan. 861
808 Computed radiography skull image enhancement using 862
809 wiener filter. In *International Conference on Pattern 863*
810 Recognition, Informatics and Medical Engineering (PRIME-2012), pages 307–311. IEEE, 2012. 2 864
811 [40] Matt Smith. Common lung diagnostic tests. <https://www.webmd.com/lung/breathing-diagnostic-tests>, 2022. 1 865
812 [41] Healthwise Staff. Chest x-ray. <https://www.healthlinkbc.ca/tests-treatments-medications/medical-tests/chest-x-ray>, 2021. 1 866
813 [42] Mingxing Tan and Quoc Le. Efficientnet: Rethinking 867
814 model scaling for convolutional neural networks. In *International conference on machine learning*, pages 868
815 6105–6114. PMLR, 2019. 3 869
816 [43] Tolga. Chest x-ray images. <https://www.kaggle.com/datasets/tolgadincer/labeled-chest-xray-images>, 2020. 2 870
817 [44] Muhammad Umer, Saima Sadiq, Muhammad Ahmad, 871
818 Saleem Ullah, Gyu Sang Choi, and Arif Mehmood. 872
819 A novel stacked cnn for malarial parasite detection in 873
820 thin blood smear images. *IEEE Access*, 8:93782–93792, 874
821 2020. 1 875
822 [45] Guangyu Wang, Xiaohong Liu, Jun Shen, Chengdi 876
823 Wang, Zhihuan Li, Linsen Ye, Xingwang Wu, Ting 877
824 Chen, Kai Wang, Xuan Zhang, et al. A deep-learning 878
825 pipeline for the diagnosis and discrimination of viral, 879
826 non-viral and covid-19 pneumonia from chest x-ray 880
827 images. *Nature biomedical engineering*, 5(6):509–521, 881
828 2021. 1 882
829 [46] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, 883
830 Mohammadkhadi Bagheri, and Ronald M Summers. 884
831 Chestx-ray8: Hospital-scale chest x-ray database and 885
832 benchmarks on weakly-supervised classification and lo- 886
833 calization of common thorax diseases. In *Proceedings 887*
834 of the IEEE conference on computer vision and pattern 888
835 recognition, pages 2097–2106, 2017. 2, 3 889
836 [47] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, 890
837 Mohammadkhadi Bagheri, and Ronald M Summers. 891
838 Nih chest x-rays. <https://www.kaggle.com/datasets/nih-chest-xrays/data>, 2017. 2, 3 892
839 [48] Wang X, Peng Y, Lu L, Lu Z, Bagheri M, and Summers 893
840 RM. Nih clinical center provides one of the largest 894
841 publicly available chest x-ray datasets to scientific 895
842 community. <https://www.nih.gov/news-events/news-releases/nih-clinical-center-provides-one-largest-publicly-available-chest-x-ray-datasets-scientific-community>, 2017. 3 896

864	[49] Lizhou Xu, Danyang Li, Sami Ramadan, Yanbin Li,	918
865	and Norbert Klein. Facile biosensors for rapid de-	919
866	detection of covid-19. <i>Biosensors and Bioelectronics</i> ,	920
867	170:112673, 2020. 1	921
868	[50] Na Zhan, Yingyun Guo, Shan Tian, Binglu Huang, Xi-	922
869	aoli Tian, Jinjing Zou, Qiutang Xiong, Dongling Tang,	923
870	Liang Zhang, and Weiguo Dong. Clinical character-	924
871	istics of covid-19 complicated with pleural effusion.	925
872	<i>BMC Infectious Diseases</i> , 21(1):1–10, 2021. 2	926
873		927
874		928
875		929
876		930
877		931
878		932
879		933
880		934
881		935
882		936
883		937
884		938
885		939
886		940
887		941
888		942
889		943
890		944
891		945
892		946
893		947
894		948
895		949
896		950
897		951
898		952
899		953
900		954
901		955
902		956
903		957
904		958
905		959
906		960
907		961
908		962
909		963
910		964
911		965
912		966
913		967
914		968
915		969
916		970
917		971