CXR-ACGAN

Auxiliary Classifier GAN for Conditional Generation of Chest X-Ray Images (Pneumonia, COVID-19 and Healthy patients).

Dataset: COVIDx CXR-3

- ❑ Create by COVID-NET team
  - 8 different data sources
  - Last release: 06/02/2022
- ❑ 2 different datasets:
  - Training Set
  - Test Set
- ❑ 3 classes: COVID-19, Pneumonia, Normal
- ❑ Two .txt file (train, test) containing metadata
  - Patient ID
  - File name
  - Class
  - Data Source

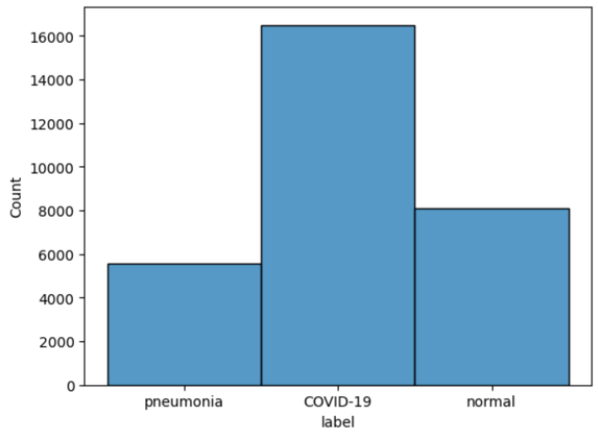
Patient ID	101
filename	pneumocystis-jirovecii-pneumonia-3-1.jpg
class	pneumonia
Data source	cohen

Data exploration

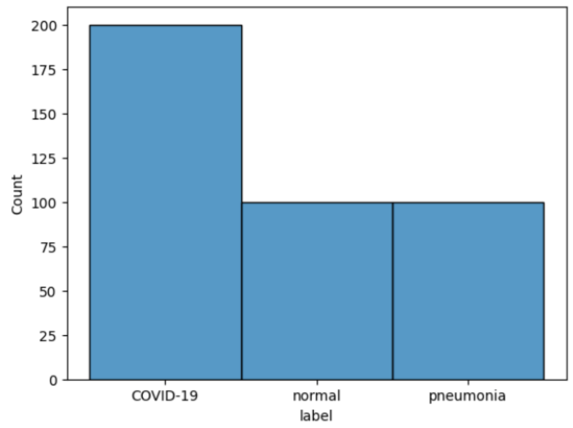
- ❑ Training set: 29.404 CXR images:
    - COVID-19: 15.774 images
    - Normal (no pathology) : 8.085 images
    - Pneumonia: 5.545 images

- ❑ Test set: 400 CXR images:
    - COVID-19: 200 images
    - Normal (no pathology) : 100 images
    - Pneumonia: 100 images

- ❑ The dataset is **imbalanced**



Training Set Distribution

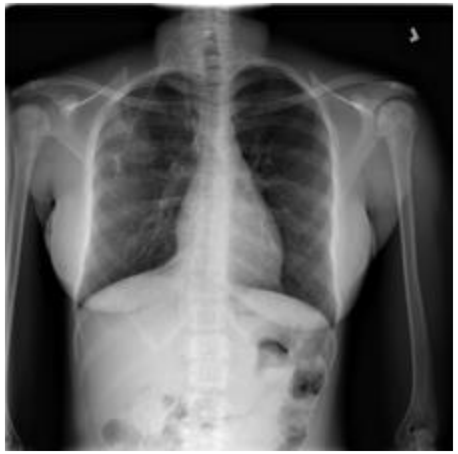


Test Set distribution

Images Exploration

- ❑ Images are 1024x1024 pixels with 3 channel:
- ❑ Only Posterior-Anterior (PA) CXR
- ❑ Many images contain:
  - Noise
  - Undesirable parts
- ❑ Preliminary operations:
  - Resized to 112x122x3
    - Reduced computational cost
  - Data Splitted
  - Data Normalization

CXR «Pneumonia»



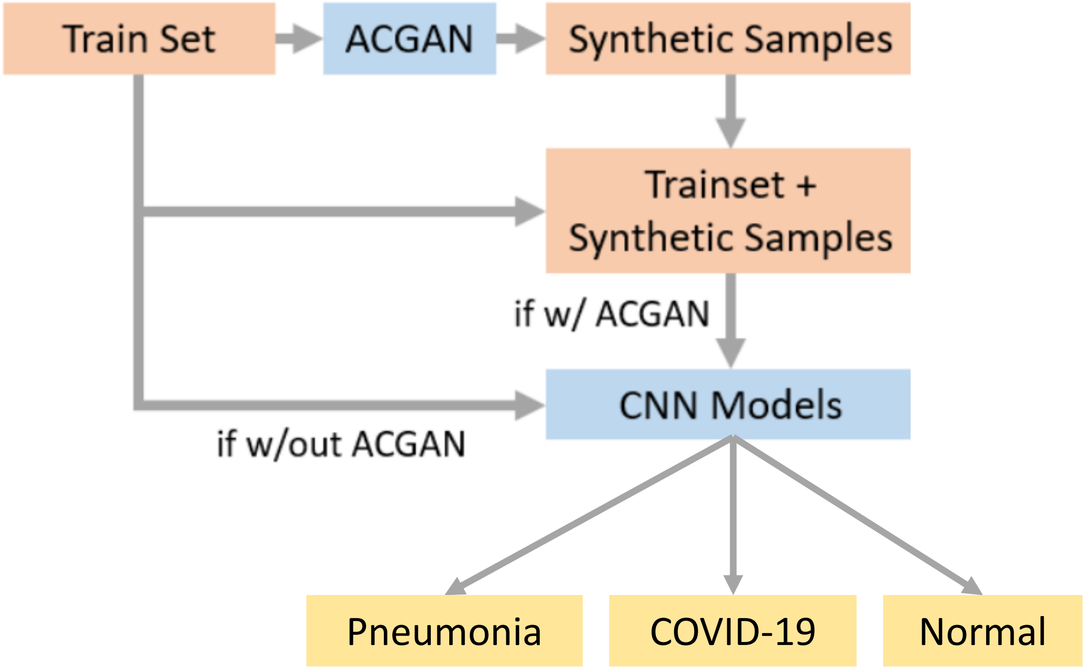
CXR «Normal»

CXR «COVID-19»

Conditional Generation of Synthetic Chest X-Ray Images

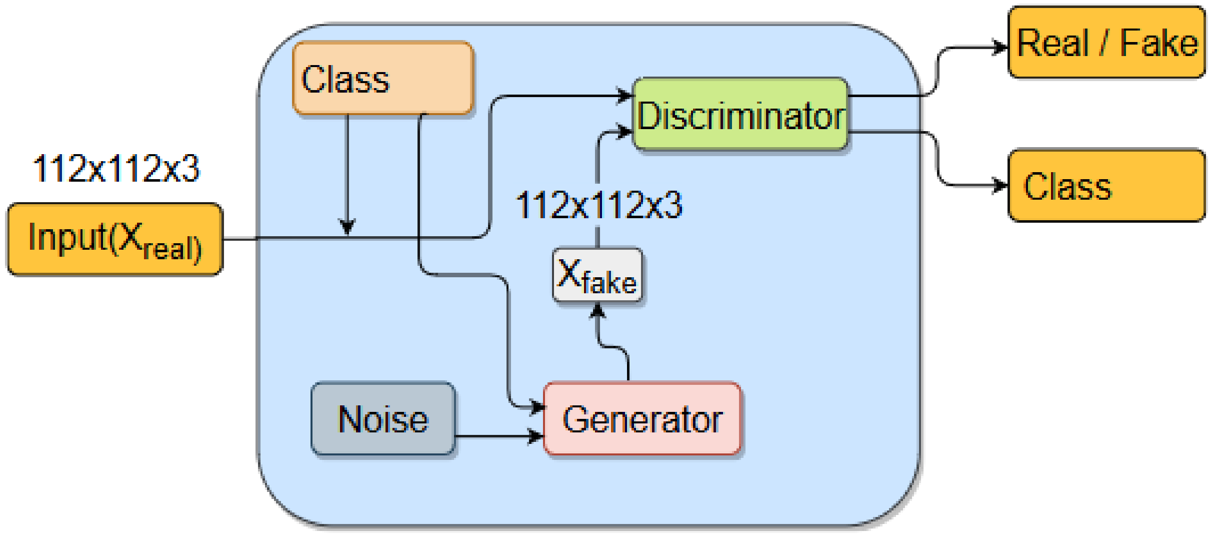
- Objectives:
  - Train an **AC-GAN** to synthesize **chest x-rays images**
  - **Conditional** generation of **healthy**, **covid-19** and **pneumonia** patients x-rays
  - **Data augmentation** on the class-imbalanced **COVIDx** dataset to improve classification performances

- Dataset → **COVIDx**
  - **Simple image pre-processing** →$112x112$ resizing and $[0,1]$ pixel scaling
  - **Data augmentation** → shearing and zooming



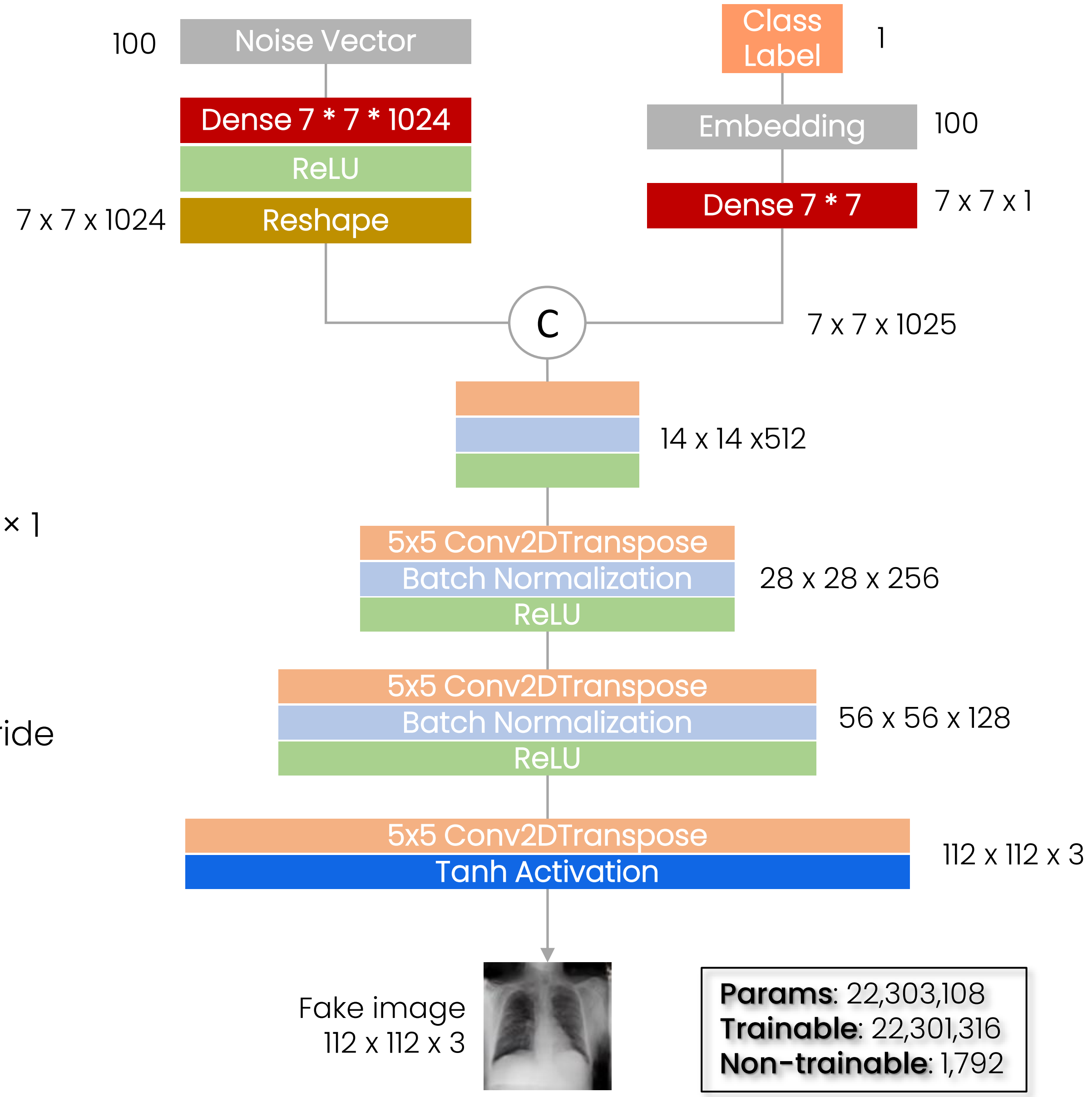
Auxiliary Classifier Generative Adversarial Network (AC-GAN)

- ❑ **AC-GAN** → extension of the GAN architecture
- ❑ The **generator** is **class conditional** as with **cGANs**
  - ❑ Input → randomly sampled **100-dimensional noise vector** and a **label**,
  - ❑ Output → conditionally generating a **112x112x3 image**
  - ❑ The **classes** → coded by integers (**0,1,2**).
- ❑ The **discriminator** → comes with an **auxiliary classifier**
  - ❑ trained to reconstruct the input image **class label**.
  - ❑ Input → 112x112x3 image (real or synthesised)
  - ❑ Output → **predicts its source** (real/fake) and **class** (0,1,2)



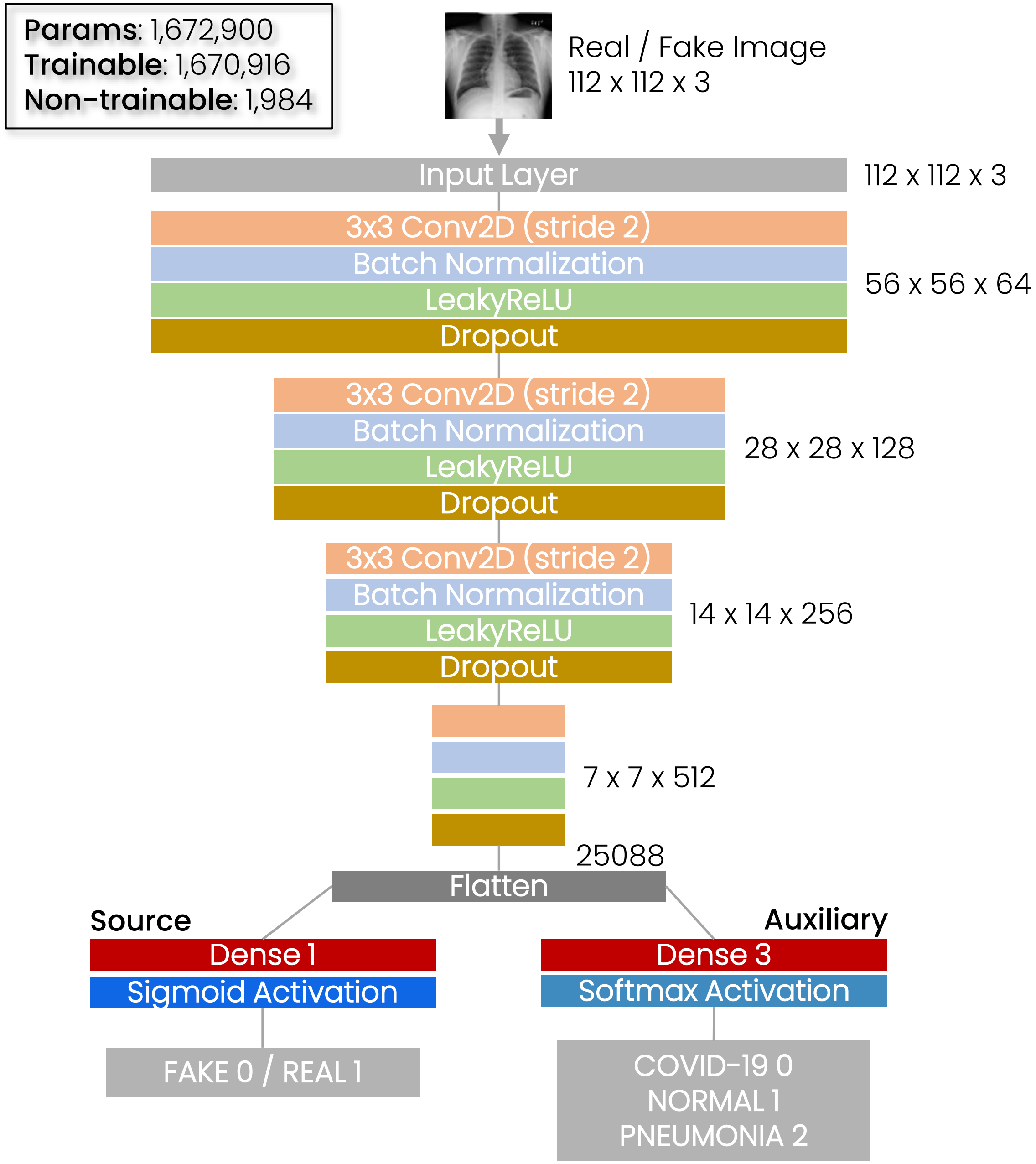
Generator

1. Two **inputs**:
    1. random 100-dimensional **noise vector**
    2. integer **class label** c (0, 1, 2)
2. **Class label** → **embedding layer** → **dense layer** → 7 × 7 × 1
3. **Noise vector** → **dense layer** → 7 × 7 × 1024
4. These two tensors are then **concatenated** → 7 × 7 × 1025
5. **Four** transposed **convolutional layers** (kernel size = 5, stride = 2) → 112 × 112 × 3
    - The first three are paired with **batch normalization** and a **Rectified Linear Unit (ReLU)** activation
    - Last one with **tanh activation**
6. Output: **fake image** with size 112 × 112 × 3



Discriminator

1. Input: **112 × 112 × 3 image** → dataset (real) or synthetic (fake)
2. **Four blocks:**
   - Sequence of: **convolutional** layer, **batch normalization** layer, **LeakyReLU** activation (slope = 0.2) and **dropout** layer (p = 0.5).
   - Image size: 112 × 112 × 3 → 7 × 7 × 512
3. The tensor is **flattened** → fed into two dense layers
4. First **dense layer + sigmoid** activation
   - **Binary classifier** → outputs a probability indicating whether the image is from the original dataset (as "real") or generated by the generator (as "fake").
5. Second **dense layer** + softmax activation
   - **Multiclass classifier** → outputs a 1D tensor of probabilities of each class



Training and regularization

- **Adam optimizer** → both the generator and the discriminator
- **Two loss functions**, one for each output layer of the **discriminator**
  - First output layer → binary cross-entropy loss (**source loss L_s**)
  - Second output layer → sparse categorical cross entropy (**auxiliary classifier loss L_c**)
- **Minimize the overall loss $L = L_s + L_c$** → during the generator training as well as the discriminator training
  - **Label flipping** (generator training) → all the fake (0) images generated are passed to discriminator labelled as real (1)
- **Labels smoothing** (discriminator training) → applied to the binary vectors describing the origin of the image (0/real – 1/fake) as a **regularization method**

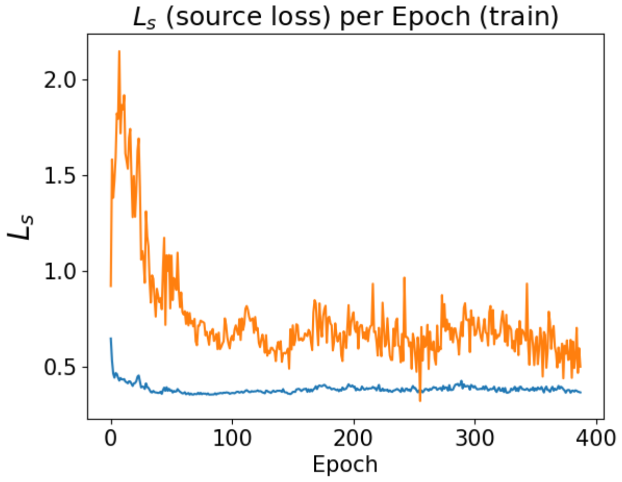
Parameters	Value
Max Epoch	388
Optimizer	Adam
Learning rate	0.0002 (fixed)
Adam β_1	0.5 (fixed)
Batch Size	64
Steps per epoch	460

Training

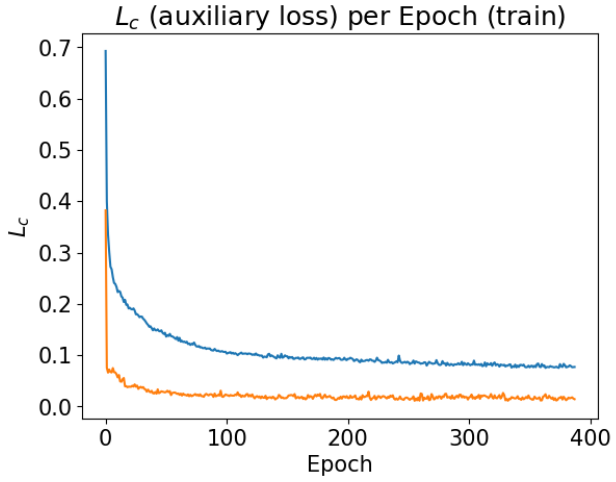
Discriminator

Generator

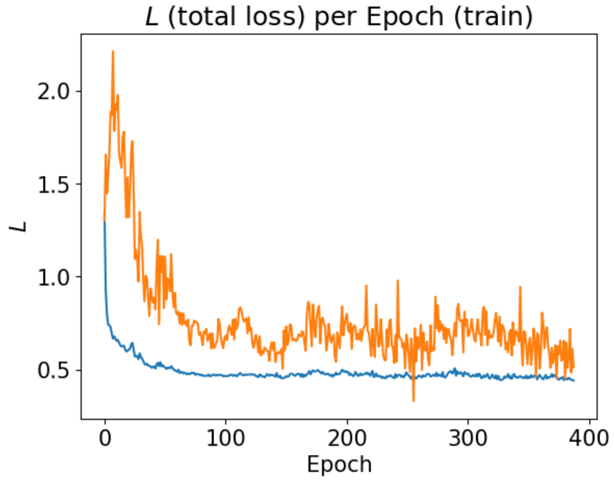
Source Loss L_s



Auxiliary Loss L_c



Total Loss L

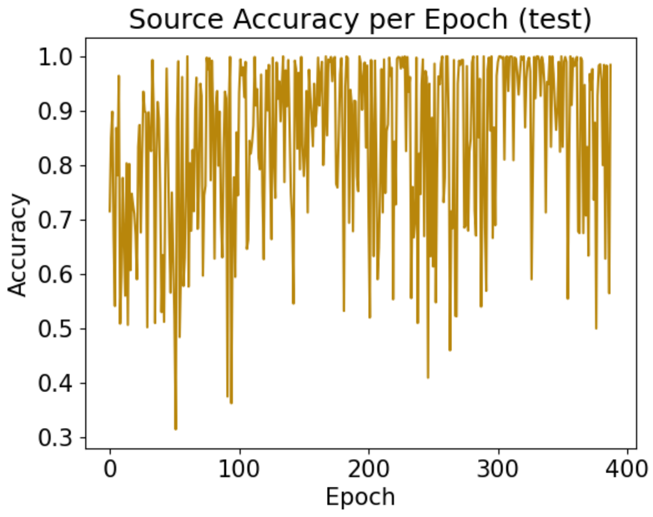


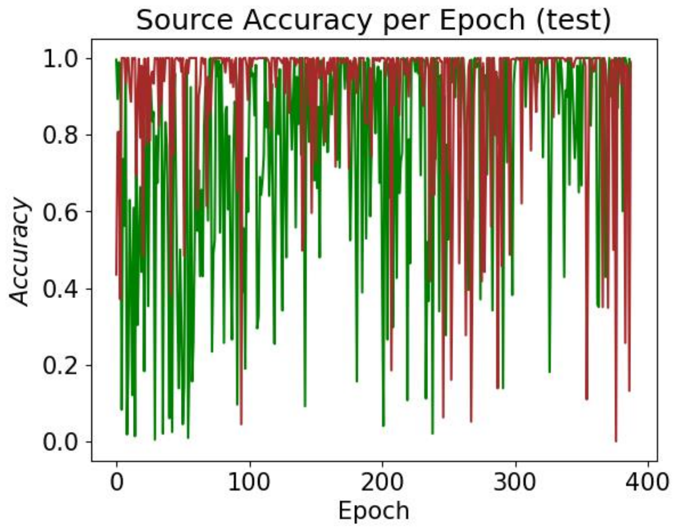
Testing Discriminator

Overall Accuracy

Real Accuracy

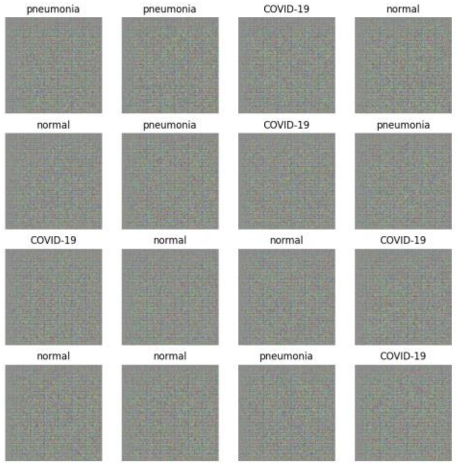
Fake Accuracy



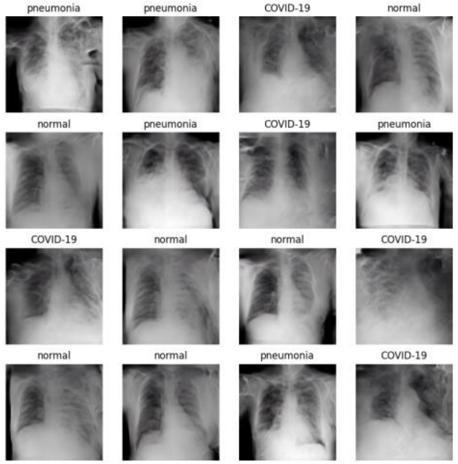


Choosing the best AC-GAN model weights

Epoch 0



Epoch 288



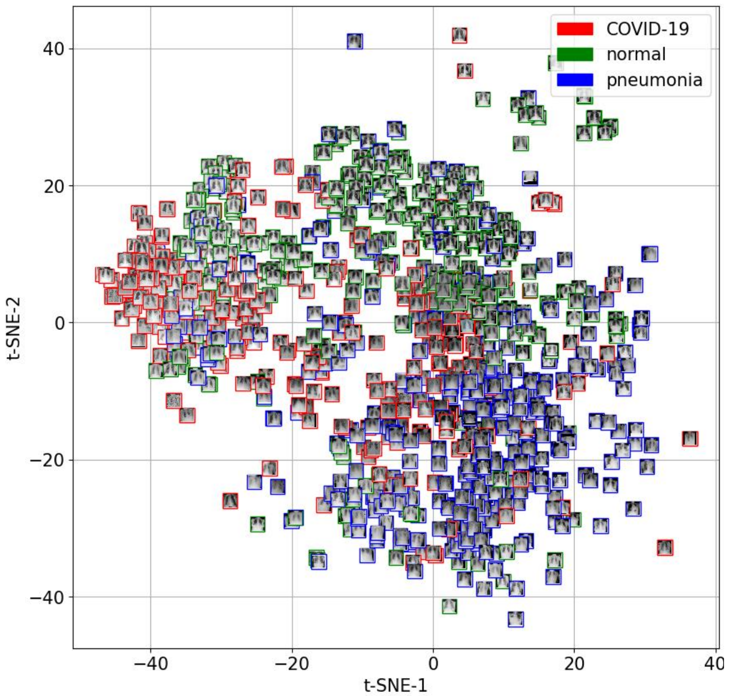
1. First set of models selection based on:
   - ❑ ↑ **visual quality qualitative evaluation** of sample images generated during each epoch
   - ❑ ↓ **generator losses**
   - ❑ ↓ **discriminator accuracy** in correctly classifying fake images as fake.
2. Trained a **classifier** on synthetic images only → evaluated the classification accuracy on real COVIDx images
   - ❑ **epoch 288** → best model
3. Generated Images Quality Evaluation
   - ❑ ↓ **FID**, ↓ **Intra-FID** and ↑ **Inception Score** (**IS**) → InceptionV3
4. **2D t-SNE embedding visualization** of generated and real images

Evaluation

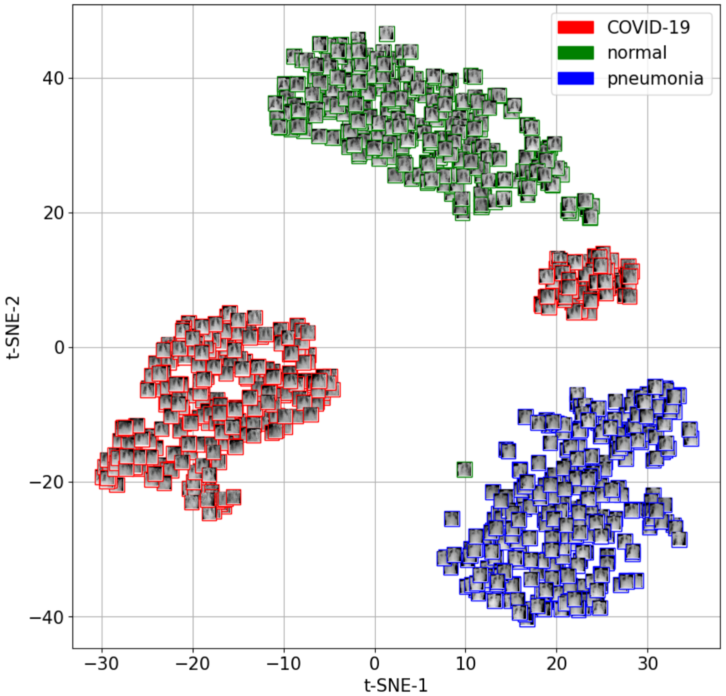
Metric	Value
Generator loss L	0.44
Discriminator accuracy (fake images)	0.13
Qualitative appearance	Realistic
CNN Accuracy (on real images)	0.63

	Our AC-GAN	Paper AC-GAN [6]
IS ↑	2.71 (± 1.70)	2.51 (± 0.12)
FID ↓	123.26 (± 0.02)	50.67 (± 8.13)
Intra FID ↓	136 (± 0.02)	

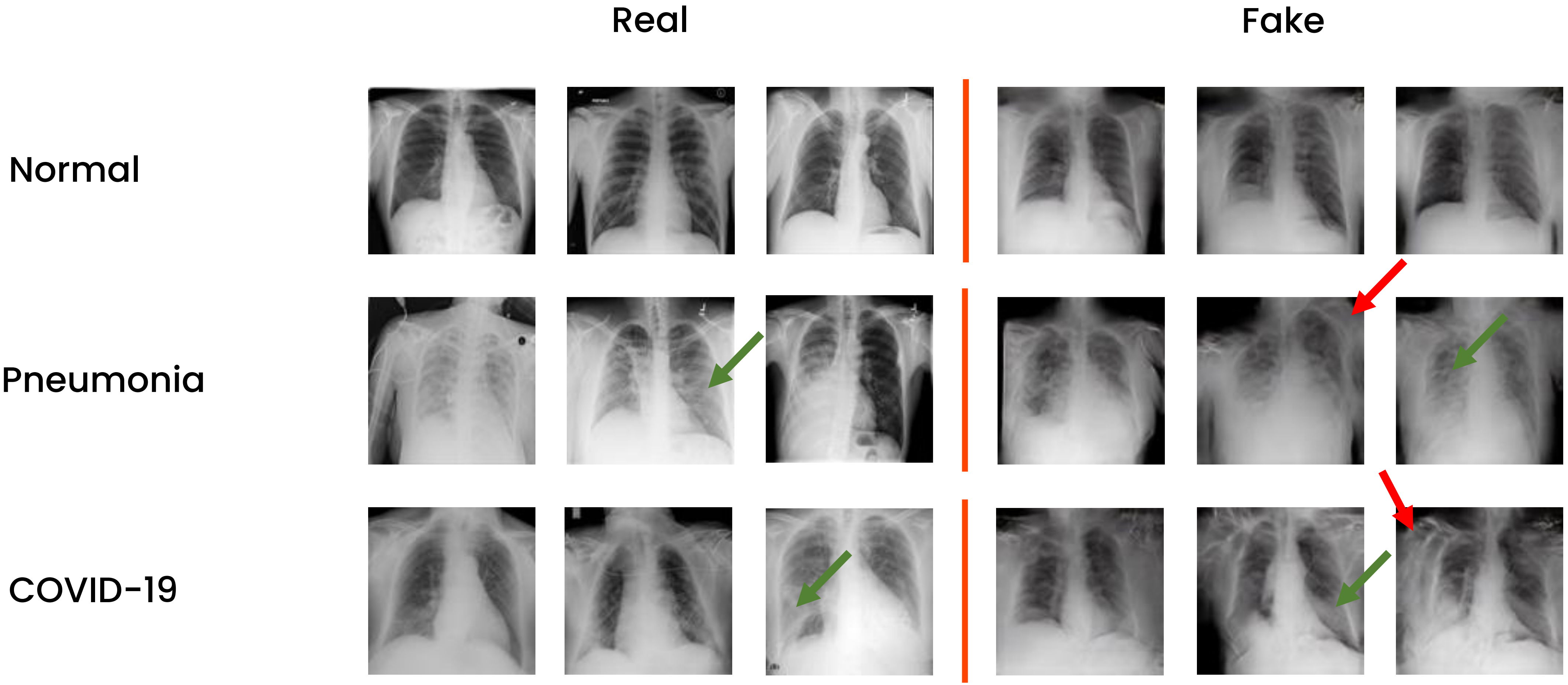
Real t-SNE



Synthetic t-SNE



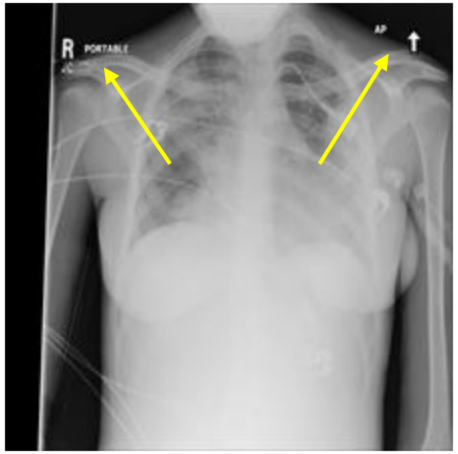
Real and Synthetic chest x-ray sample



CHEST X-RAY CLASSIFICATION

Image Pre-Processing

- **Image Enhancement**:
  - Techniques used to improve the information interpretability in images
    - For radiologists and automated systems
- **Pre-Processing**
  - Removal of textual information commonly embedded in CXR images

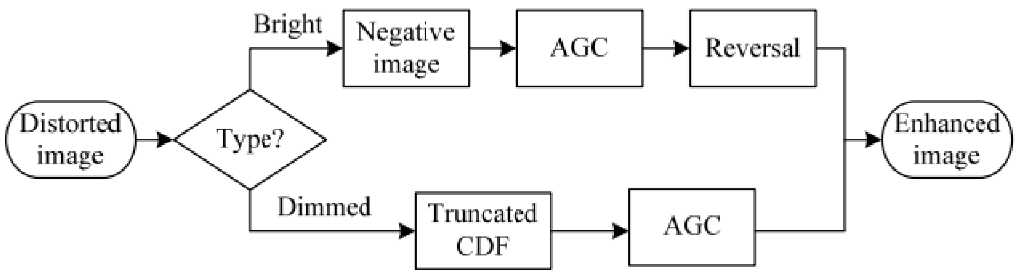


Common textual items

Noisy CXR-image

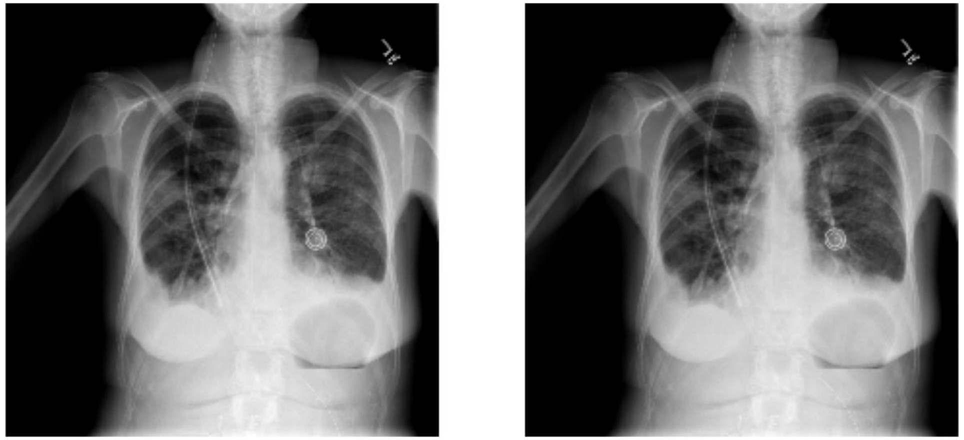
Improved Adaptive Gamma Correction

- Adaptive Gamma Correction tool
  - AGC (Adaptive Gamma Correction) is a tool for image contrast
  - AGC relates the gamma parameter with the cumulative distribution function (CDF) of the pixel gray levels
  - good for most dimmed images, but fails for globally bright images
- **Improved Adaptive Gamma Correction**
  - new AGC algorithm
  - enhance bright images with the use of negative images
  - enhance dimmed images with the use of gamma correction modulated by truncated CDF

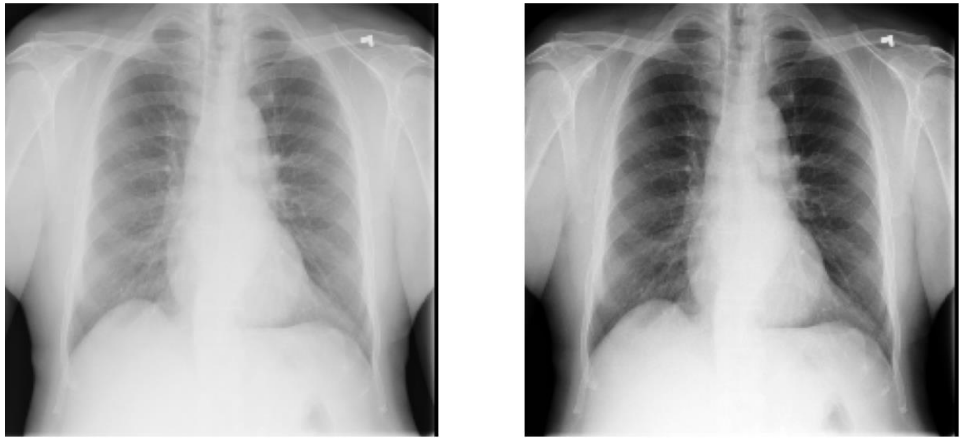


Flowchart of Improved AGC tool

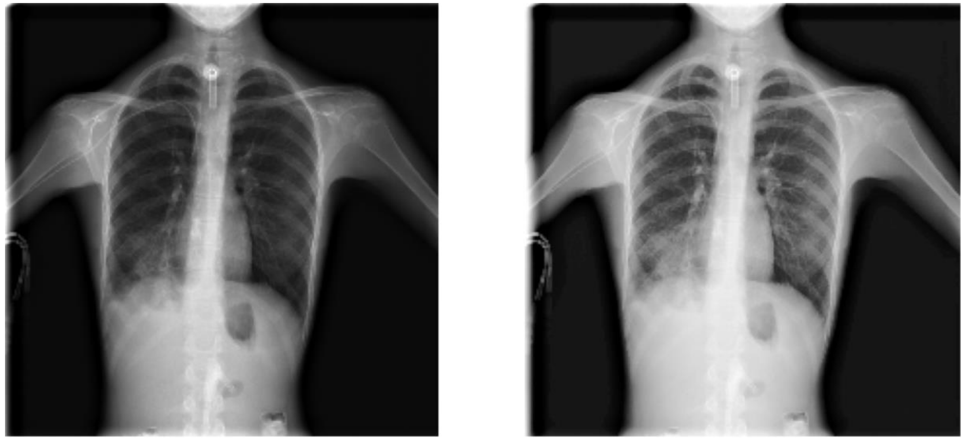
Improved Adaptive Gamma Correction



No ACG applied



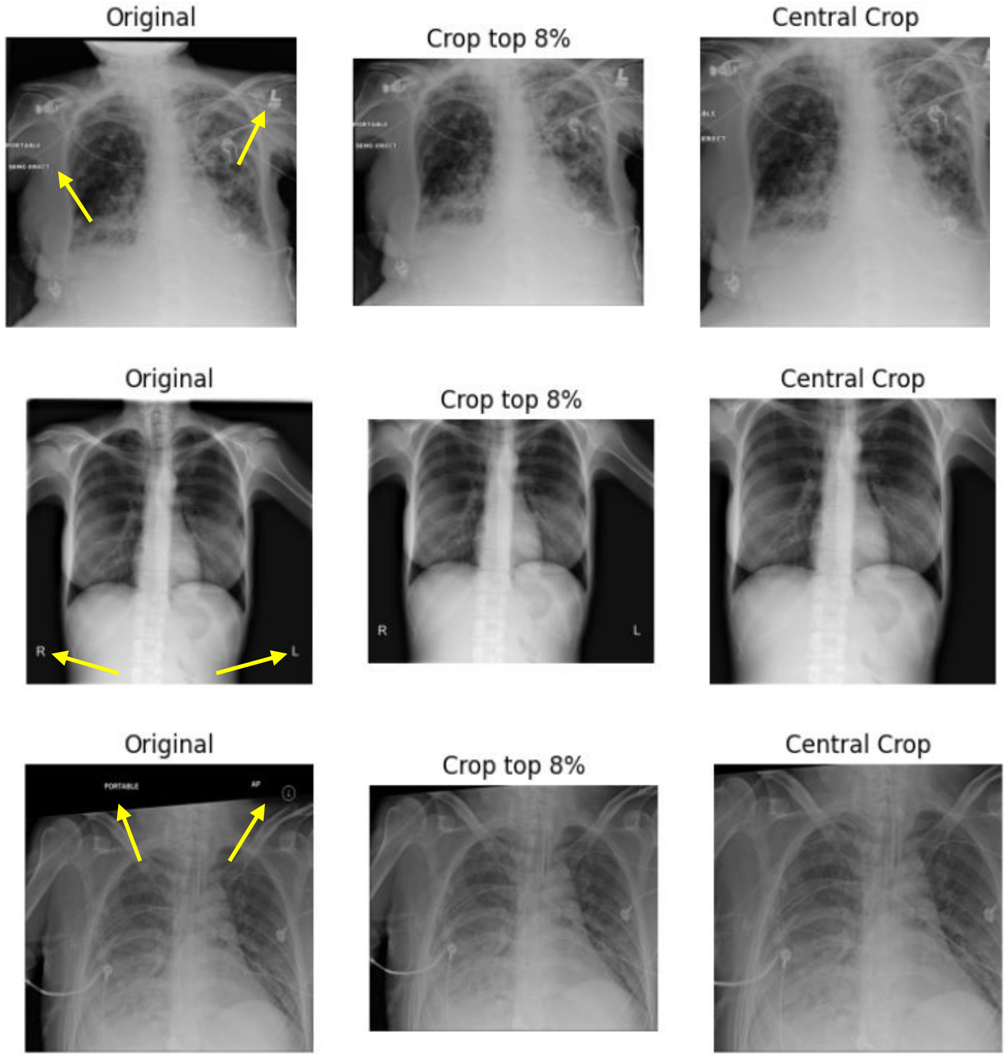
ACG applied (too bright)



ACG applied (too dim)

Pre-Processing:

- The chest CXR images were cropped
  - top 8% of the image
    - Commonly embedded textual information
  - Central crop
    - To Centre the cropped image



Some pre-processing examples

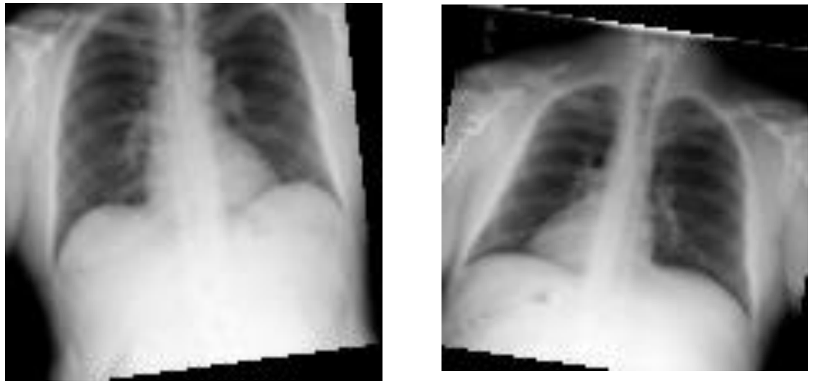
Class imbalance problem

- Different techniques explored to handle unbalanced classes
  - **Under-sampling** of the dataset
    - Rebalancing with respect to the least populated class
  - **Class-weights**
    - Assigns higher weights to samples from underrepresented classes
  - **Over-sampling** of the dataset
    - Data augmentation on minority classes
      - Positional-based Data Augmentation
      - GAN

Classes	Nr. images
COVID-19	15.774
Pneumonia	5.545
Normal	8.085
Total	**29.904**

Data Augmentation

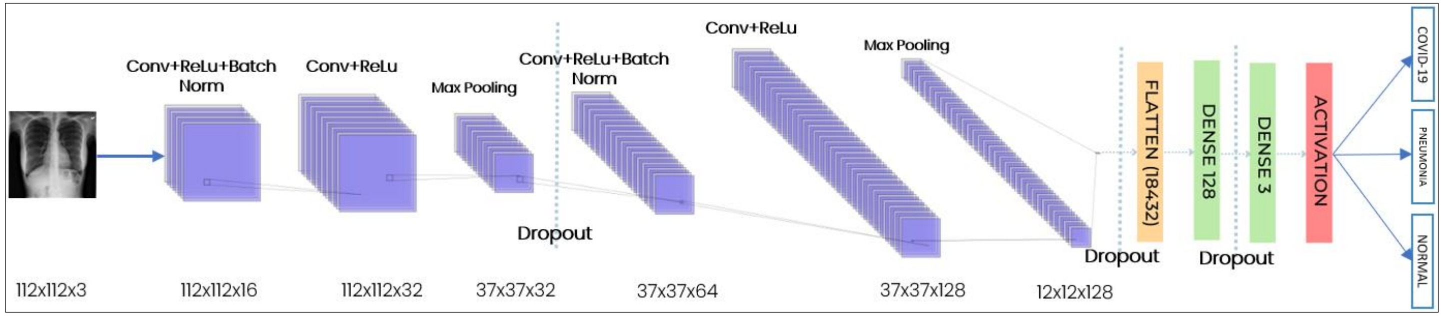
- A data augmentation technique was adopted to balance the classes, in particular was:
  - Implemented **after under-sampling** (performing it on all classes)
  - Implemented **to increase minority classes** (not performing it on the most populated class)
- Data augmentation was exploited with the **following types of augmentation**:
  - Translation (± 10% in x and y directions)
  - Rotation (± 10)
  - Horizontal flip, zoom (± 15%)
  - Intensity shift (± 10%)



Some augmentation examples

CNN: Network Architecture

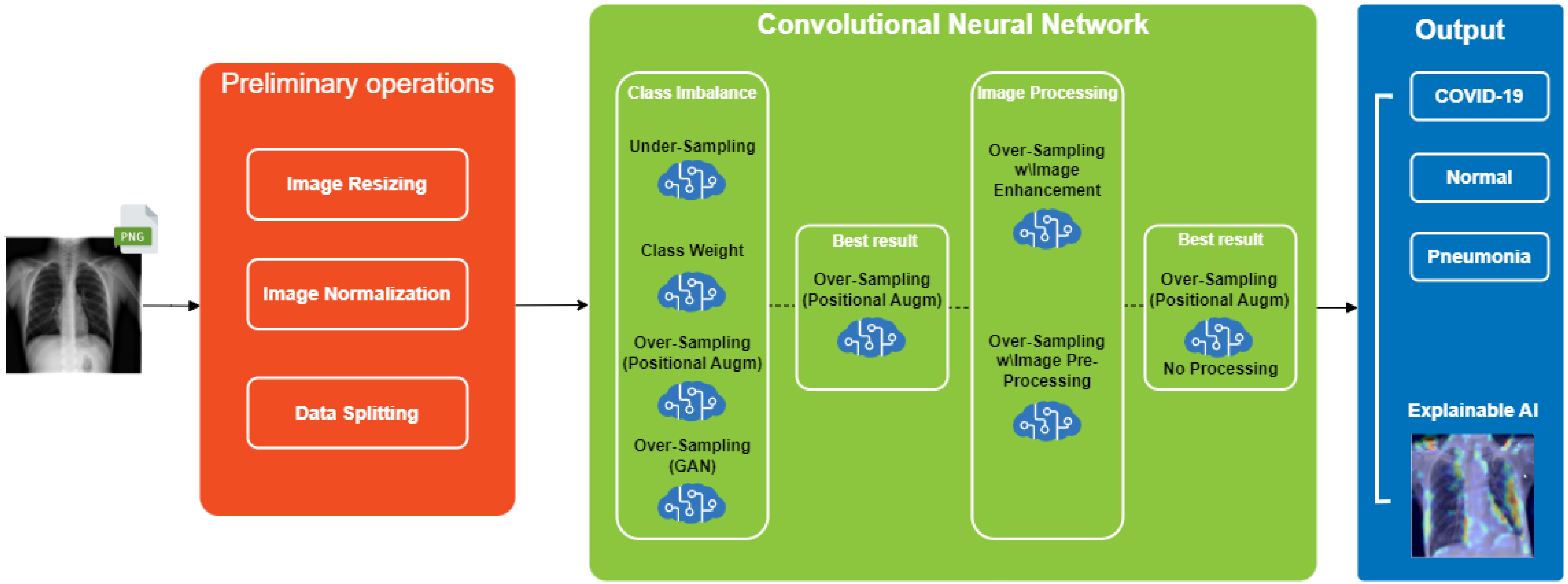
Params: 2,416,611
Trainable: 2,416,451
Non-trainable: 160



- **Input layer** (112x112x3)
- 2 convolutional blocks, with:
  - **Convolutional** layers
  - **Batch Normalization** layers
  - **ReLu**
- 2 convolutional blocks with: **Convolutional** layer, **ReLu**
- 2 **Max Pooling** layers
- 2 **Dropout** layers (rate 0,2)
- Output of feature extractor is passed to **Flatten layer**
- **Fully connected** layer (128 neurons), ReLu
- **Dropout** layer (rate 0,5)
- **Output layer,** 3 neurons, **Softmax** activation function

Parameters	Value
Max Epoch	50
Optimizer	Adam
Learning rate	0.0001 (fixed)
Batch Size	32
Step per epoch	1035

Overview



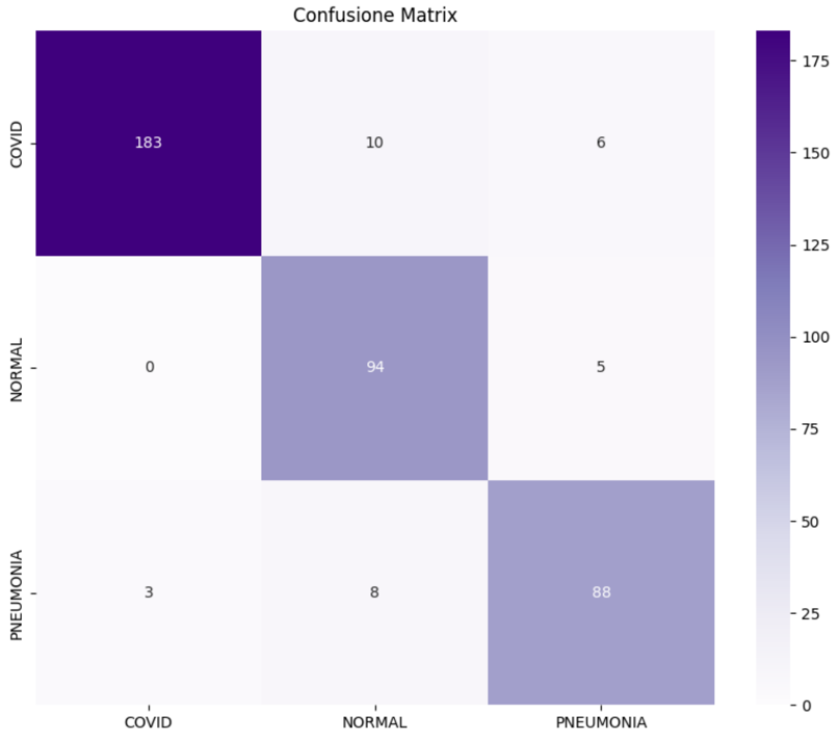
PNG
Preliminary operations
Image Resizing
Image Normalization
Data Splitting
Convolutional Neural Network
Class Imbalance
Under-Sampling
Class Weight
Over-Sampling (Positional Augm)
Over-Sampling (GAN)
Best result
Over-Sampling (Positional Augm)
Image Processing
Over-Sampling w\Image Enhancement
Over-Sampling w\Image Pre-Processing
Best result
Over-Sampling (Positional Augm)
No Processing
Output
COVID-19
Normal
Pneumonia
Explainable AI

Over-Sampling w\Positional Augmentation Results

- The solution that produced the **best results** turned out to be the one:
  - without preprocessing
  - and Over-Sampling of minority classes with **positional augmentation**

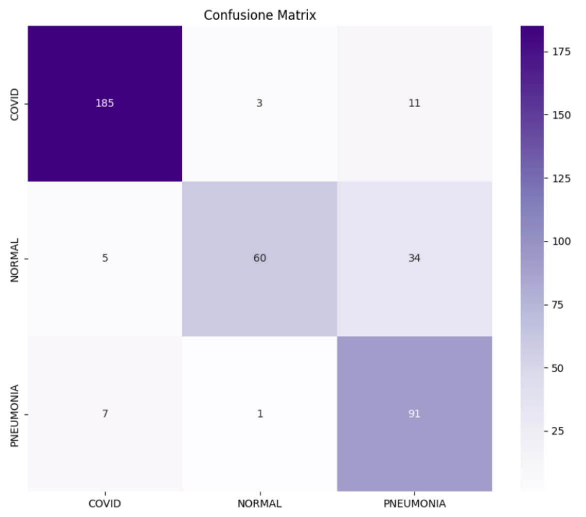
```
...........................................
> Correct Predictions: 365
> Wrong Predictions: 32
...........................................
```

	precision	recall	f1-score	support
COVID	0.98	0.92	0.95	199
NORMAL	0.84	0.95	0.89	99
PNEUMONIA	0.89	0.89	0.89	99
accuracy			0.92	397
macro avg	0.90	0.92	0.91	397
weighted avg	0.92	0.92	0.92	397



Confusion matrix on test set

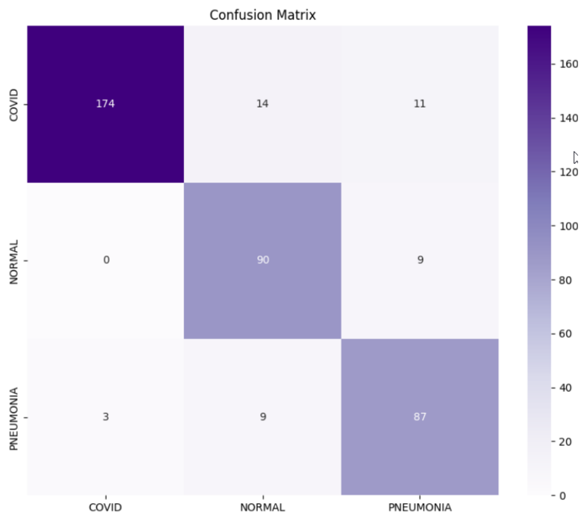
Under-sampling




```
---------------------------------------------
DCCN - Undersampling
> Accuracy on train: 0.87        Loss on train: 0.36
> Accuracy on test: 0.85         Loss on test: 0.45
---------------------------------------------

.............................................
> Correct Predictions: 336
> Wrong Predictions: 61
.............................................
```

	precision	recall	f1-score	support
COVID	0.94	0.93	0.93	199
NORMAL	0.94	0.61	0.74	99
PNEUMONIA	0.67	0.92	0.77	99
accuracy			0.85	397
macro avg	0.85	0.82	0.82	397
weighted avg	0.87	0.85	0.85	397

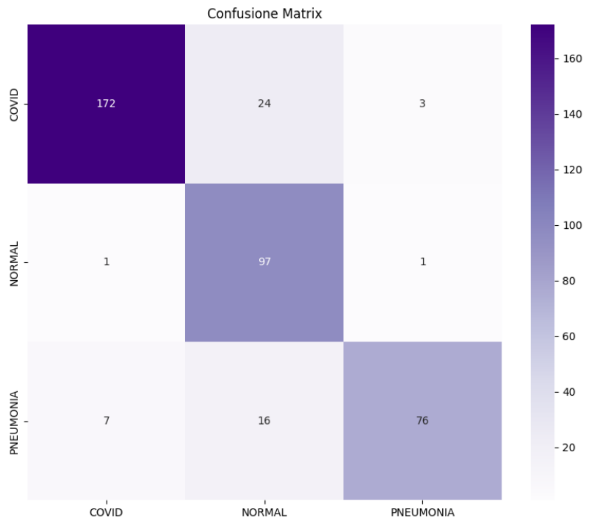
Class-Weights




```
---------------------------------------------
DCCN - Class Weights
> Accuracy on train: 0.96        Loss on train: 0.11
> Accuracy on test: 0.88         Loss on test: 0.46
---------------------------------------------

.............................................
> Correct Predictions: 351
> Wrong Predictions: 46
.............................................
```

	precision	recall	f1-score	support
COVID	0.98	0.87	0.93	199
NORMAL	0.80	0.91	0.85	99
PNEUMONIA	0.81	0.88	0.84	99
accuracy			0.88	397
macro avg	0.86	0.89	0.87	397
weighted avg	0.89	0.88	0.89	397

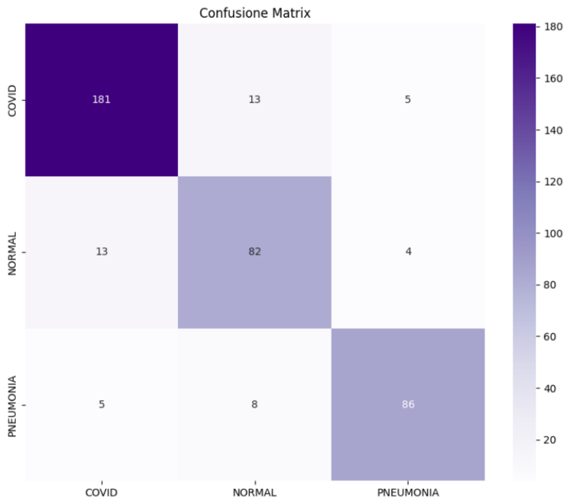
AC-GAN Augmentation




```
---------------------------------------------
DCCN - Oversampling w\AC-cGAN
> Accuracy on train: 0.99        Loss on train: 0.02
> Accuracy on test: 0.87         Loss on test: 0.6
---------------------------------------------

.............................................
> Correct Predictions: 345
> Wrong Predictions: 52
.............................................
```

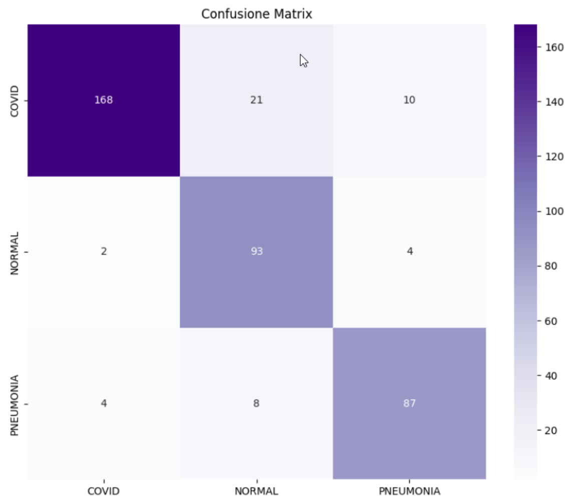
	precision	recall	f1-score	support
COVID	0.96	0.86	0.91	199
NORMAL	0.71	0.98	0.82	99
PNEUMONIA	0.95	0.77	0.85	99
accuracy			0.87	397
macro avg	0.87	0.87	0.86	397
weighted avg	0.89	0.87	0.87	397

Image Processing




```
---------------------------------------------
DCCN - Image Processing
> Accuracy on train: 0.98       Loss on train: 0.05
> Accuracy on test: 0.88        Loss on test: 0.97
---------------------------------------------
.........................................
> Correct Predictions: 349
> Wrong Predictions: 48
.........................................
              precision    recall  f1-score   support

       COVID       0.91      0.91      0.91       199
      NORMAL       0.80      0.83      0.81        99
   PNEUMONIA       0.91      0.87      0.89        99

    accuracy                           0.88       397
   macro avg       0.87      0.87      0.87       397
weighted avg       0.88      0.88      0.88       397
```

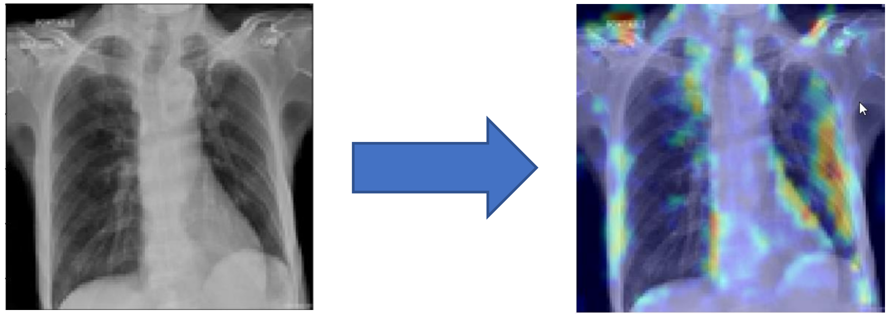
Image Enhancement



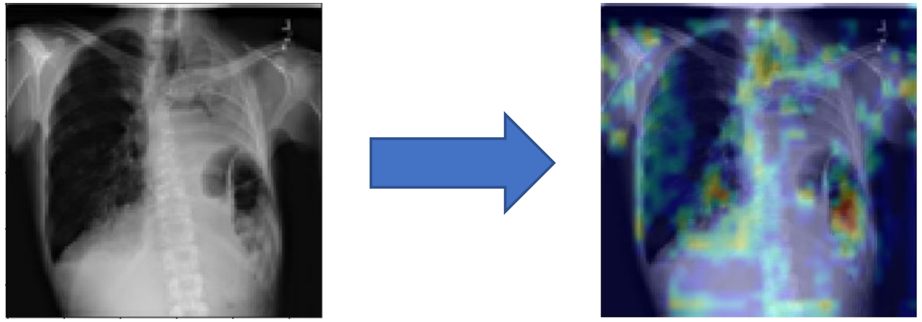

```
---------------------------------------------
DCCN - Oversampling - Image Enanchment
> Accuracy on train: 0.96       Loss on train: 0.1
> Accuracy on test: 0.88        Loss on test: 0.43
---------------------------------------------
.........................................
> Correct Predictions: 349
> Wrong Predictions: 48
.........................................
              precision    recall  f1-score   support

       COVID       0.91      0.91      0.91       199
      NORMAL       0.80      0.83      0.81        99
   PNEUMONIA       0.91      0.87      0.89        99

    accuracy                           0.88       397
   macro avg       0.87      0.87      0.87       397
weighted avg       0.88      0.88      0.88       397
```

Explainable AI: Class activation Heat-Map

- We developed an **explainability algorithm** based on the use of Gradient-weighted Class Activation Mapping (**Grad-CAM**)
  - It provides a visual output of **the most interesting areas** found by the proposed CNN models
  - Grad-CAM uses the gradients of any target concept, flowing into the final convolutional layer to produce a coarse localization map highlighting the important regions in the image for predicting the concept.



COVID-19 CXR, Activation Map



Pneumonia CXR, Activation Map

/ Bibliography

1. Karbhari, Y., Basu, A., Geem, Z. W., Han, G.-T., & Sarkar, R. (2021). Generation of Synthetic Chest X-ray Images and Detection of COVID-19: A Deep Learning Based Approach. Diagnostics, 11(5), Article 5. https://doi.org/10.3390/diagnostics11050895.

2. DeVries, T., Romero, A., Pineda, L., Taylor, G. W., & Drozdzal, M. (2019). On the Evaluation of Conditional GANs (arXiv:1907.08175). arXiv. https://doi.org/10.48550/arXiv.1907.08175

3. Borji, A. (2018). Pros and Cons of GAN Evaluation Measures (arXiv:1802.03446). arXiv. https://doi.org/10.48550/arXiv.1802.03446

4. Goel S, Kipp A, Goel N, et al. (November 22, 2022) COVID-19 vs. Influenza: A Chest X-ray Comparison. Cureus 14(11): e31794. doi:10.7759/cureus.31794

5. Kim, S.-H.; Wi, Y.M.; Lim, S.; Han, K.-T.; Bae, I.-G. Differences in Clinical Characteristics and Chest Images between Coronavirus Disease 2019 and Influenza-Associated Pneumonia. Diagnostics 2021, 11, 261. https://doi.org/10.3390/diagnostics11020261

6. Odena, A., Olah, C., & Shlens, J. (2017). Conditional Image Synthesis With Auxiliary Classifier GANs (arXiv:1610.09585). arXiv. https://doi.org/10.48550/arXiv.1610.09585

7. Christi Florence, C. (2021). Detection of Pneumonia in Chest X-Ray Images Using Deep Transfer Learning and Data Augmentation With Auxiliary Classifier Generative Adversarial Network. 14.

/ Bibliography

8. 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. Sci Rep 10, 19549 (2020). https://doi.org/10.1038/s41598-020-76550-z

9. Gang Cao, Lihui Huang, Huawei Tian, Xianglin Huang, Yongbin Wang, Ruicong Zhi, Contrast enhancement of brightness-distorted images by improved adaptive gamma correction, Computers & Electrical Engineering, Volume 66, 2018, Pages 569-582, ISSN 0045-7906, https://doi.org/10.1016/j.compeleceng.2017.09.012.

10. Ait Nasser, A.; Akhloufi, M.A. A Review of Recent Advances in Deep Learning Models for Chest Disease Detection Using Radiography. Diagnostics 2023, 13, 159. https://doi.org/10.3390/diagnostics13010159

11. Huang, W., Song, G., Li, M., Hu, W., Xie, K. (2013). Adaptive Weight Optimization for Classification of Imbalanced Data. In: Sun, C., Fang, F., Zhou, ZH., Yang, W., Liu, ZY. (eds) Intelligence Science and Big Data Engineering. IScIDE 2013. Lecture Notes in Computer Science, vol 8261. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-42057-3_69

12. Elshennawy, N.M.; Ibrahim, D.M. Deep-Pneumonia Framework Using Deep Learning Models Based on Chest X-Ray Images. Diagnostics 2020, 10, 649. https://doi.org/10.3390/diagnostics10090649

13. Chetoui, M.; Akhloufi, M.A.; Yousefi, B.; Bouattane, E.M. Explainable COVID-19 Detection on Chest X-rays Using an End-to-End Deep Convolutional Neural Network Architecture. Big Data Cogn. Comput. 2021, 5, 73. https://doi.org/10.3390/bdcc5040073