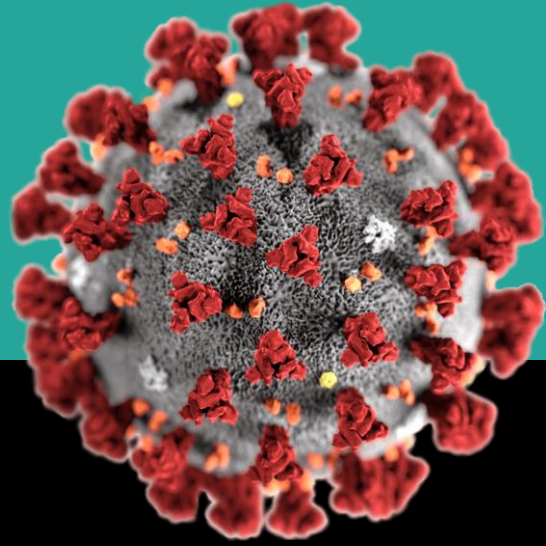


# Covid-19 diagnosis using CNN

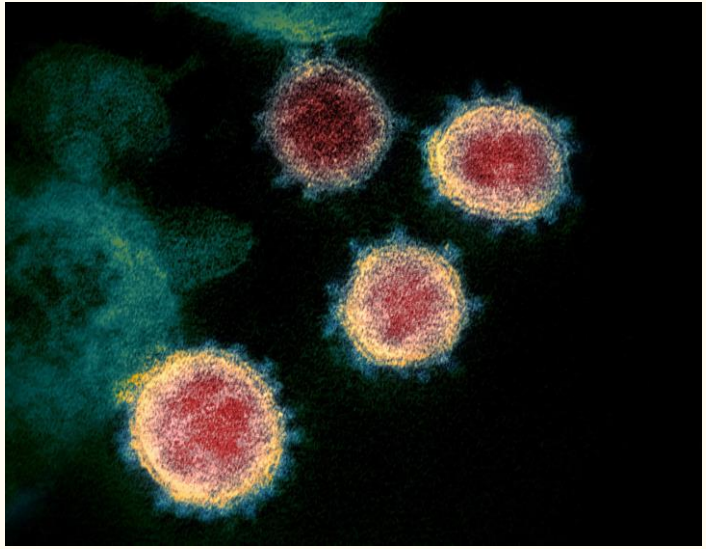
Manos Koutoulakis & Manos Markodimitrakis

# Outline

- COVID-19
- Scope of this project
- Dataset
- Tools
- References

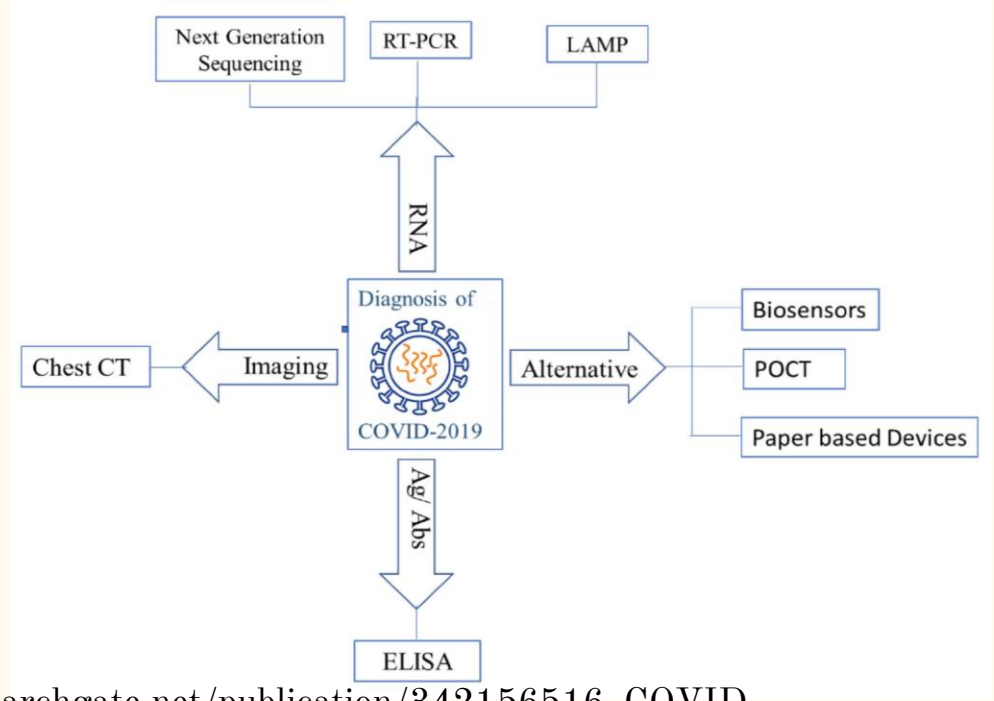
# COVID-19

- **Coronavirus Disease 2019** (COVID-19) is an infectious disease derived from Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2).
- The first case was identified in Wuhan, China in December 2019
- From December 2019 until now the virus has spread all over the world



Source : https://el.wikipedia.org/wiki/COVID-19

# Current diagnostic approaches

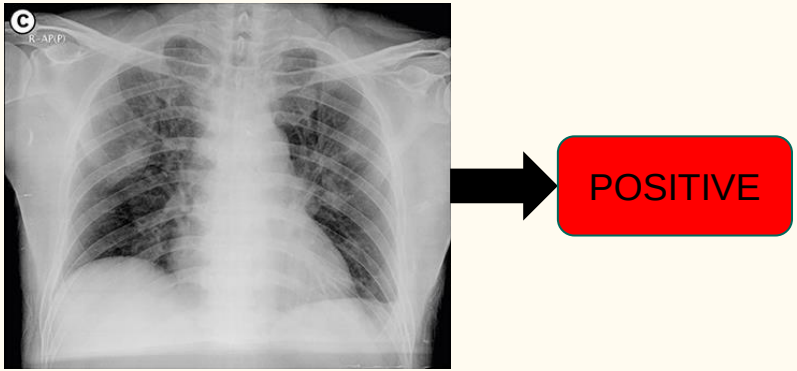


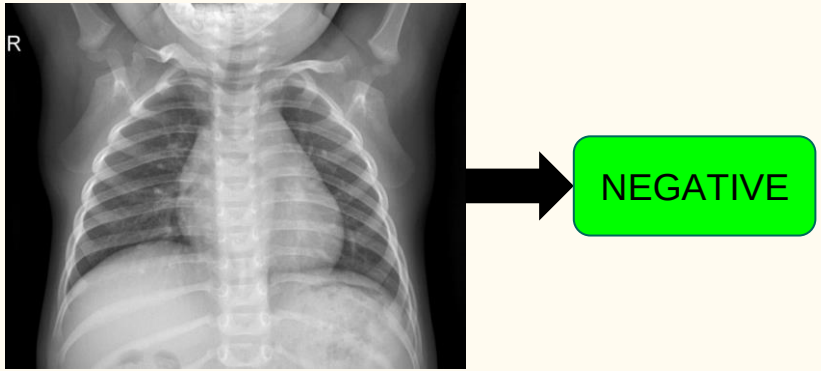

Source: https://www.researchgate.net/publication/342156516_COVID-19_diagnostic_approaches_different_roads_to_the_same_destination

# Scope of this project

# Scope

1. To diagnose if the patient has Covid-19 through chest X-ray using neural networks
2. Develop and improve the current deep learning approaches
3. Validate our results



# Dataset

# Dataset (1/2)

- The dataset contains 950 patients
- Each of them has their own X-Ray image
- There is a wide range of viruses and bacterias in this dataset

| | |
|---|---|
| COVID-19 | 584 |
| Pneumonia | 81 |
| SARS | 16 |
| Pneumocystis | 30 |
| Streptococcus | 22 |
| No Finding | 22 |
| Chlamydophila | 3 |
| E.Coli | 4 |
| Klebsiella | 10 |
| Legionella | 10 |
| Unknown | 1 |
| Lipoid | 13 |
| Varicella | 6 |
| Bacterial | 4 |
| Mycoplasma | 11 |
| Influenza | 5 |
| todo | 83 |
| Tuberculosis | 18 |
| H1N1 | 2 |
| Aspergillosis | 2 |
| Herpes | 3 |
| Aspiration | 1 |
| Nocardia | 8 |
| MERS-CoV | 10 |
| MRSA | 1 |

## Medical Images


Normal


Covid-19

Source: https://github.com/ieee8023/covid-chestxray-dataset
https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

# Dataset (2/2)

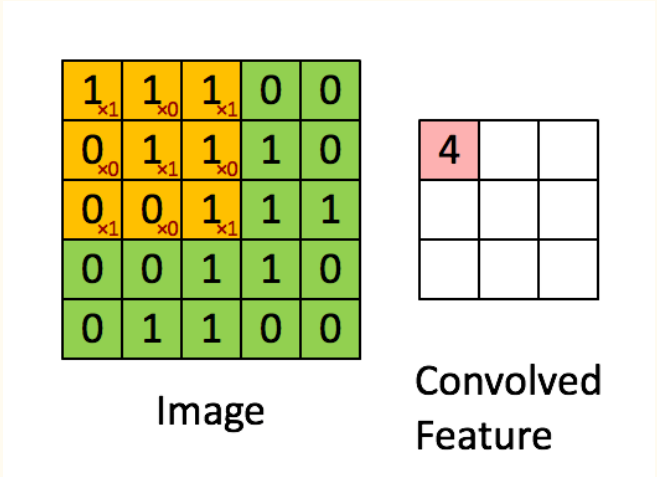
| Feature | Number of subjects |
|---|---|
| Sex | 870 |
| Age | 713 |
| Finding | 950 |
| Survival | 361 |
| Went ICU | 397 |
| In ICU | 335 |
| Needed O2 support | 90 |

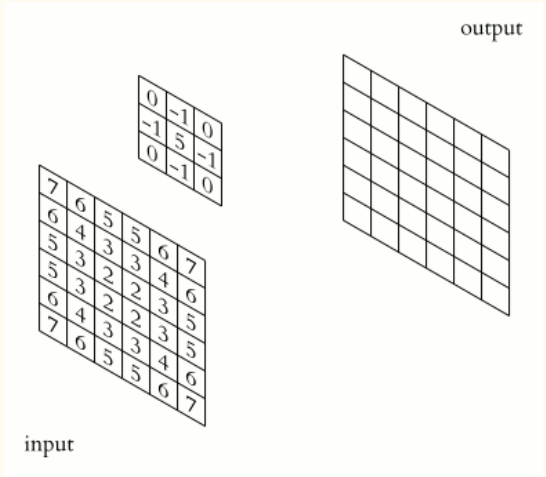
| Feature | Number of subjects |
|---|---|
| Temperature | 78 |
| pO2 Saturation | 119 |
| Modality | 950 |
| View | 950 |
| Location | 894 |
| Filename | 950 |
| Date | 661 |

# Tools

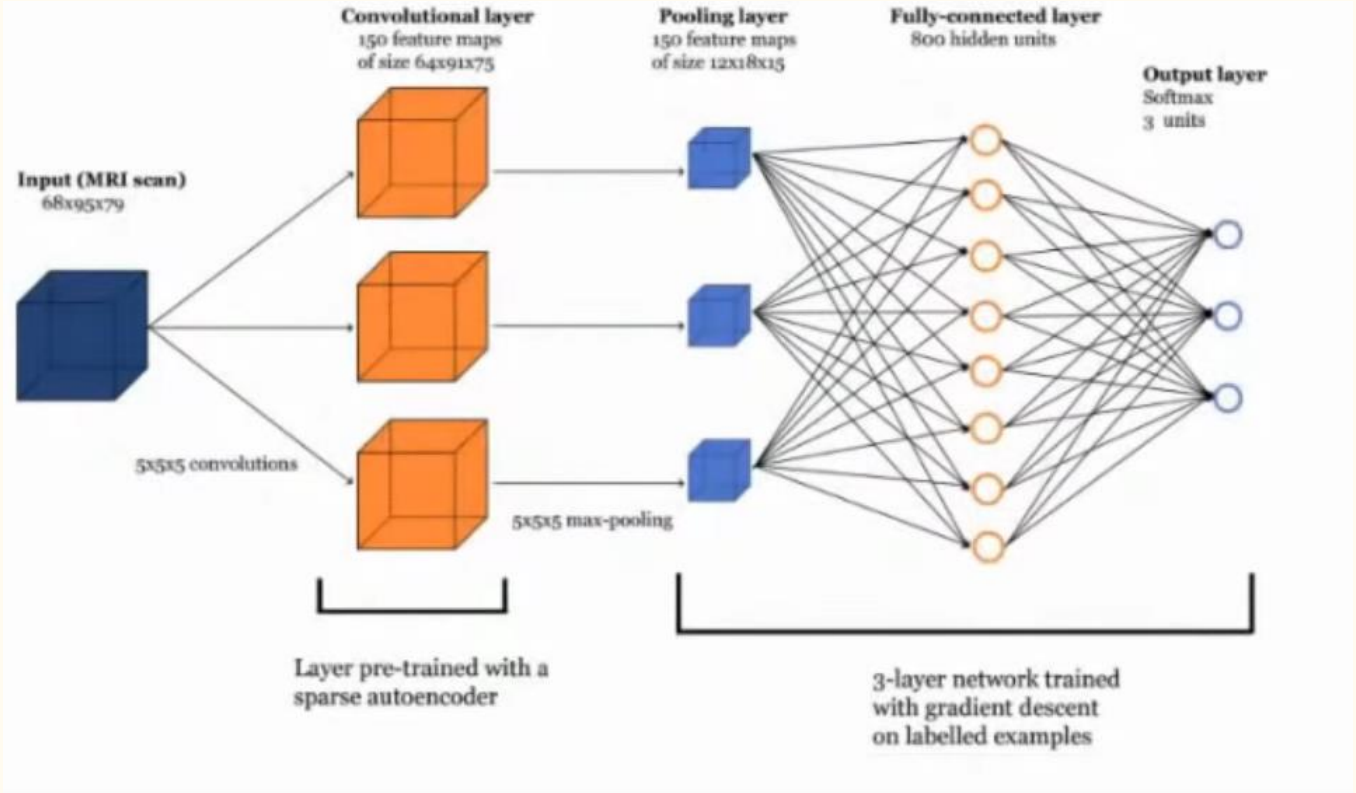
- Tensorflow
- Matplotlib
- Pandas
- Sklearn.skimage
- Keras

# Feature extraction

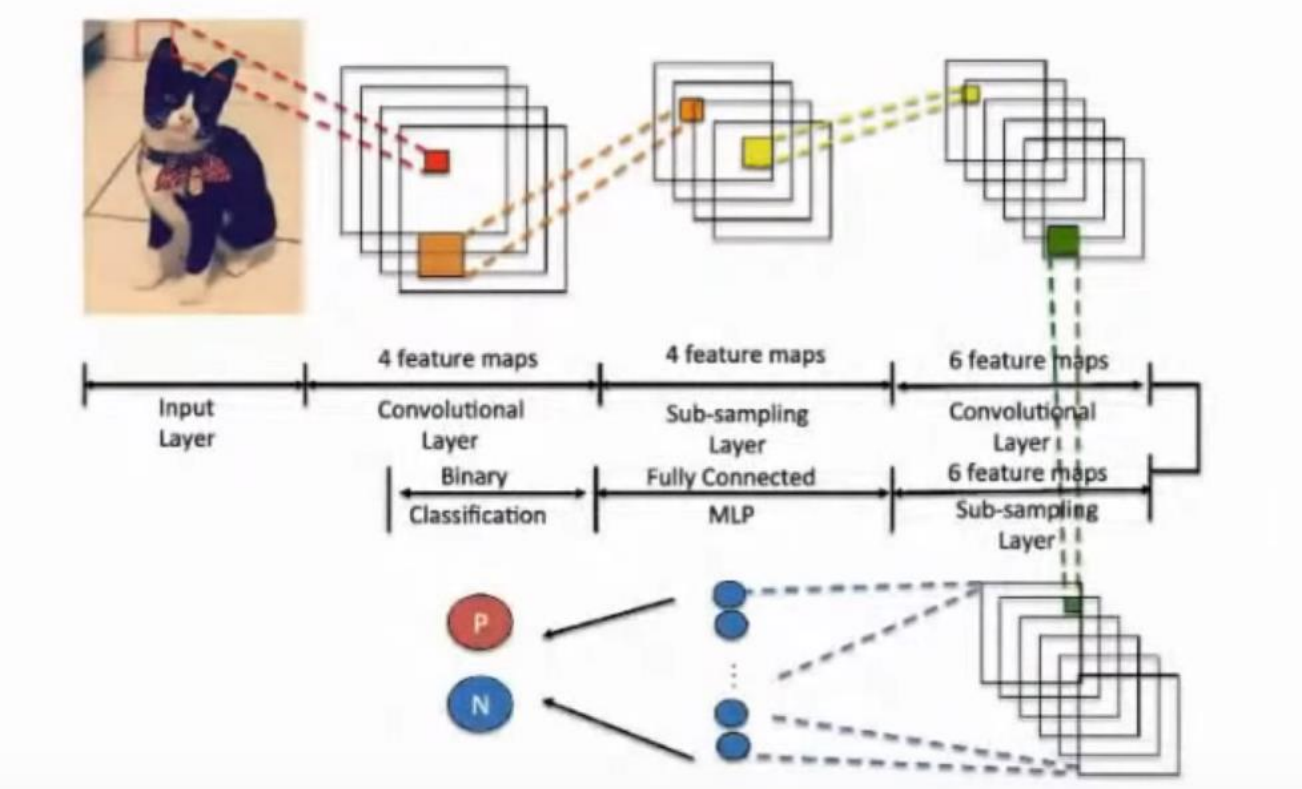



# Classify the Image according to Features



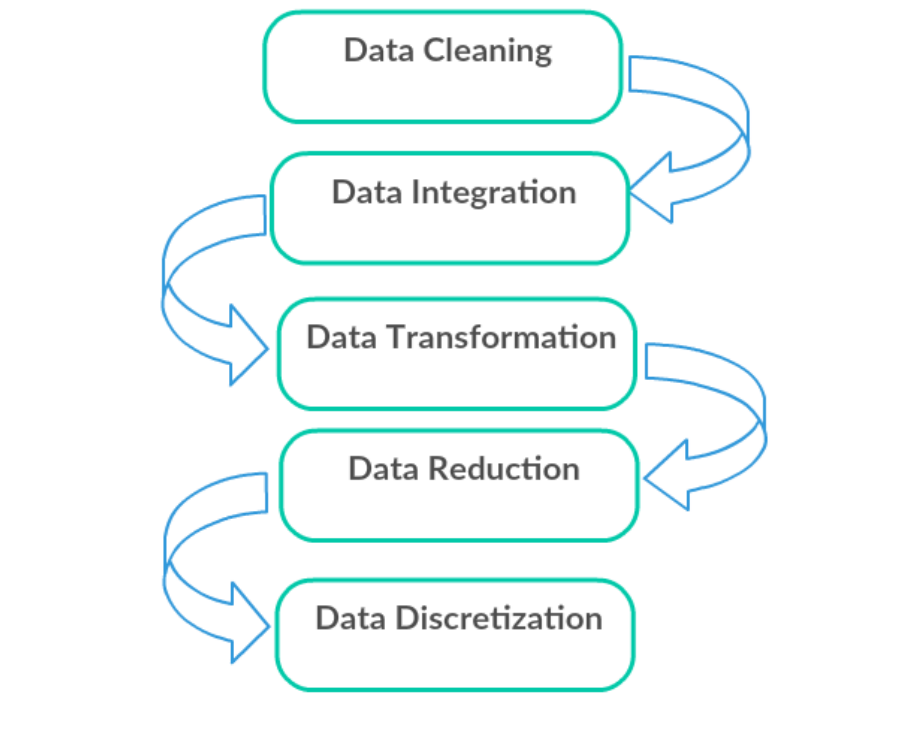
# Feature extraction + Classification = CNN



# Workflow

- Data preprocessing
- Model creation
- Training
- Validation

# Data preprocessing



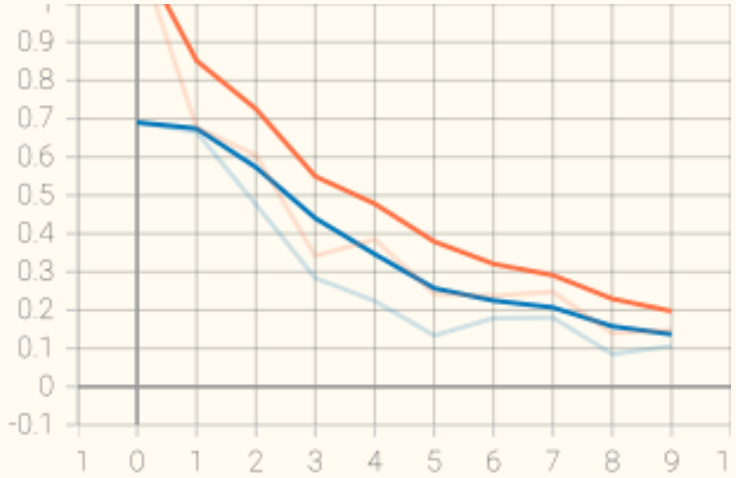
# Model creation

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_8 (Conv2D)            (None, 222, 222, 32)      896
_________________________________________________________________
conv2d_9 (Conv2D)            (None, 220, 220, 64)      18496
_________________________________________________________________
max_pooling2d_6 (MaxPooling2 (None, 110, 110, 64)      0
_________________________________________________________________
dropout_8 (Dropout)          (None, 110, 110, 64)      0
_________________________________________________________________
conv2d_10 (Conv2D)           (None, 108, 108, 64)      36928
_________________________________________________________________
max_pooling2d_7 (MaxPooling2 (None, 54, 54, 64)        0
_________________________________________________________________
dropout_9 (Dropout)          (None, 54, 54, 64)        0
_________________________________________________________________
conv2d_11 (Conv2D)           (None, 52, 52, 128)       73856
_________________________________________________________________
max_pooling2d_8 (MaxPooling2 (None, 26, 26, 128)       0
_________________________________________________________________
dropout_10 (Dropout)         (None, 26, 26, 128)       0
_________________________________________________________________
flatten_2 (Flatten)          (None, 86528)             0
_________________________________________________________________
dense_4 (Dense)              (None, 64)                5537856
_________________________________________________________________
dropout_11 (Dropout)         (None, 64)                0
_________________________________________________________________
dense_5 (Dense)              (None, 1)                 65
=================================================================
Total params: 5,668,097
Trainable params: 5,668,097
Non-trainable params: 0
```

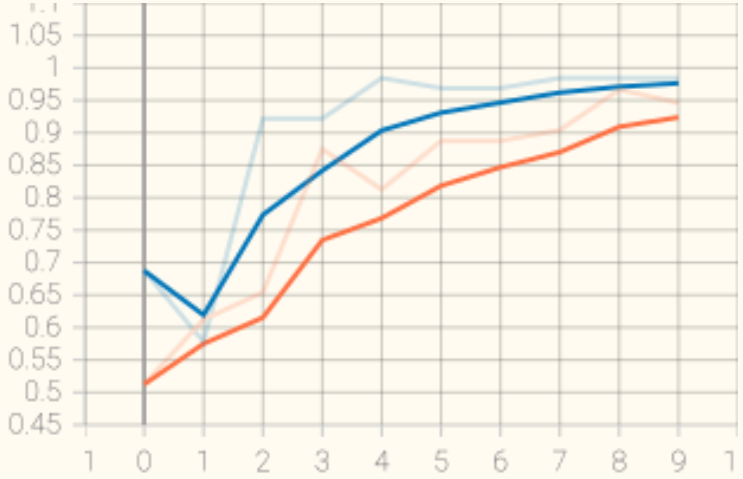
# Training

```
Epoch 1/10
1/8 [==>...........................] - ETA: 0s - loss: 0.7509 - accuracy: 0.4375WARNING:tensorflow:From /home/manos/anaconda3/envs/tensorEnv/lib/python3.6/site-packages/tensorflow/python/ops/summary_ops_v2.py:1277: stop (from tensorflow.python.eager.profiler) is deprecated and will be removed after 2020-07-01.
Instructions for updating:
use `tf.profiler.experimental.stop` instead.
8/8 [==============================] - 19s 2s/step - loss: 1.1420 - accuracy: 0.5125 - val_loss: 0.6904 - val_accuracy: 0.6875
Epoch 2/10
8/8 [==============================] - 18s 2s/step - loss: 0.6777 - accuracy: 0.6125 - val_loss: 0.6652 - val_accuracy: 0.5781
Epoch 3/10
8/8 [==============================] - 18s 2s/step - loss: 0.6046 - accuracy: 0.6542 - val_loss: 0.4759 - val_accuracy: 0.9219
Epoch 4/10
8/8 [==============================] - 18s 2s/step - loss: 0.3420 - accuracy: 0.8750 - val_loss: 0.2837 - val_accuracy: 0.9219
Epoch 5/10
8/8 [==============================] - 18s 2s/step - loss: 0.3849 - accuracy: 0.8125 - val_loss: 0.2239 - val_accuracy: 0.9844
Epoch 6/10
8/8 [==============================] - 19s 2s/step - loss: 0.2425 - accuracy: 0.8875 - val_loss: 0.1338 - val_accuracy: 0.9688
Epoch 7/10
8/8 [==============================] - 18s 2s/step - loss: 0.2373 - accuracy: 0.8875 - val_loss: 0.1789 - val_accuracy: 0.9688
Epoch 8/10
8/8 [==============================] - 18s 2s/step - loss: 0.2479 - accuracy: 0.9042 - val_loss: 0.1806 - val_accuracy: 0.9844
Epoch 9/10
8/8 [==============================] - 18s 2s/step - loss: 0.1389 - accuracy: 0.9667 - val_loss: 0.0851 - val_accuracy: 0.9844
Epoch 10/10
8/8 [==============================] - 18s 2s/step - loss: 0.1490 - accuracy: 0.9458 - val_loss: 0.1062 - val_accuracy: 0.9844
```

# Validation(1/2)




Epoch Loss




Epoch Accuracy

# Validation(2/2) - Confusion matrix

# References


1. *Gianluca Maguolo*, *Loris Nanni*, “A Critic Evaluation of Methods for COVID-19 Automatic Detection from X-Ray Images ” *Sci. Data*, vol. 5, no. 1, p. 180202, Dec. 2018, Cite as: https://arxiv.org/2004.12823
2. Brownlee J , “How to configure Image Data Augmentation in Keras”, April. 2019, Retrieved from: https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/
3. *Sekeroglu, B., & Ozsahin, I.*, Detection of COVID-19 from Chest X-Ray Images Using Convolutional Neural Networks. SLAS TECHNOLOGY: Translating Life Sciences Innovation, 247263032095837, Sep 2020, doi:10.1177/2472630320958376
4. *Jain R., Gupta M., Taneja S., et al.* “Deep learning based detection and analysis of COVID-19 on chest X-ray images”, Appl Intell (2020), doi: https://doi.org/10.1007/s10489-020-01902-1

# Covid-19 diagnosis using CNN

Source Code : https://github.com/manoskout/covid-19-diagnosis-using-cnn/

Thank you for your attention

Questions?

Manos Koutoulakis & Manos Markodimitrakis