PROJECT-1 *(ECD-416)*

COVID 19 Detection from radiograph using Computer Vision Deep Learning

Submitted By-

Sunil Kumar 17mi438
Divyanshu Bhaik 17mi446
Prayas Thakur 17mi437
Rahul Sharma 17mi426

ECE Dual Degree (BTech & MTech)

Under the Guidance of

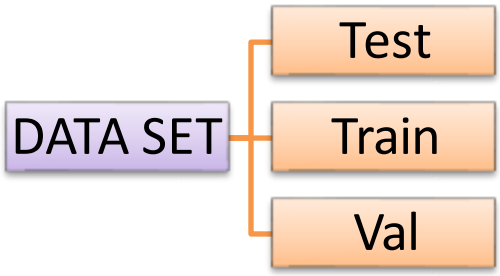
Dr. Abhijit Bhattacharyya, Assistant Professor,

Department of Electronics and Communication Engineering

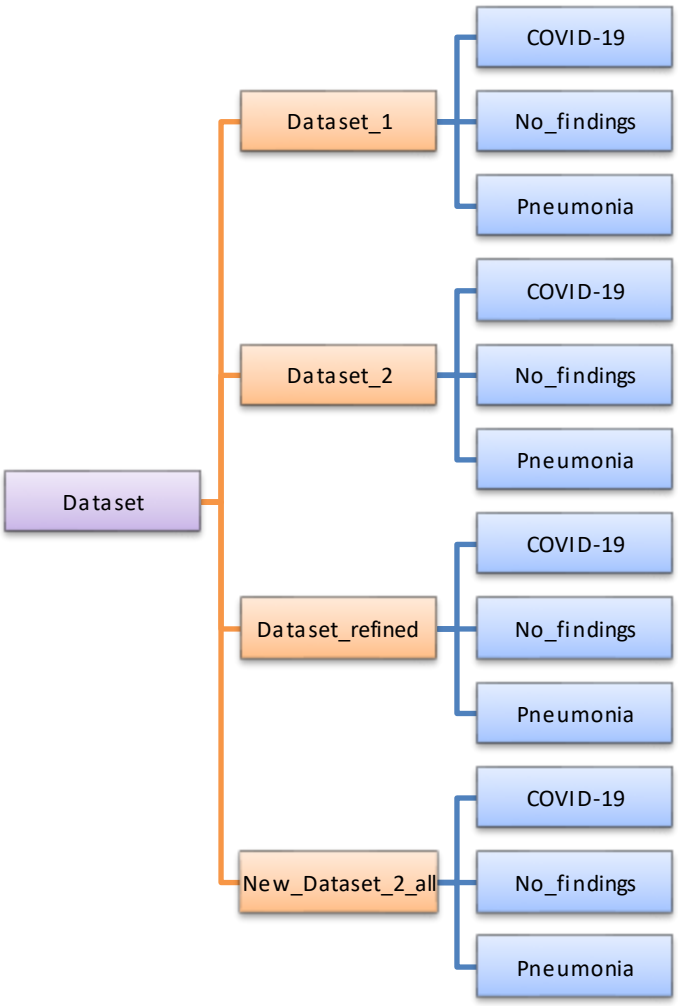
National Institute of Technology Hamirpur, 177005 Himachal Pradesh

Data Set

Image Pre-processing Dataset



Classification Dataset



We have used four dataset for this research from given sources
https://arxiv.org/abs/2003.11597
https://github.com/ieee8023/covid-chestxray-dataset
https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia
we further created folders of data set to attain higher accuracy.

Dataset_1`

Dataset_1 is taken from Cohen. Which contains 1347 images.

Dataset_refined

Dataset_refined is the subset of Dataset_1 which contains 1000 best images from previous dataset.

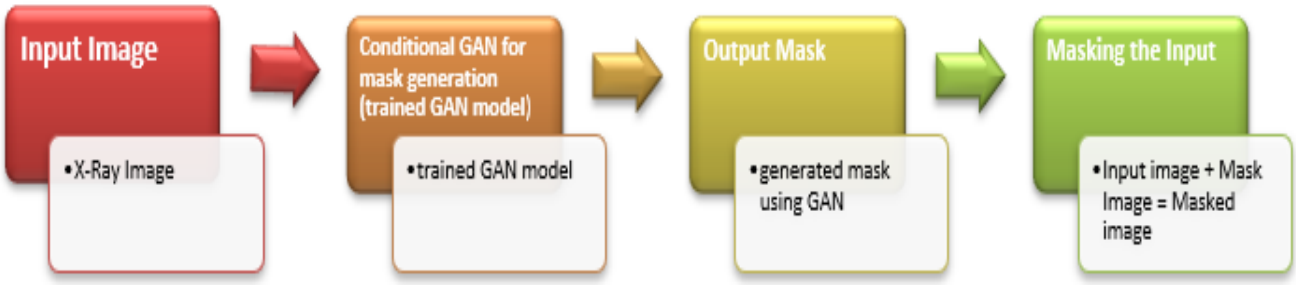
New_Dataset_2_all

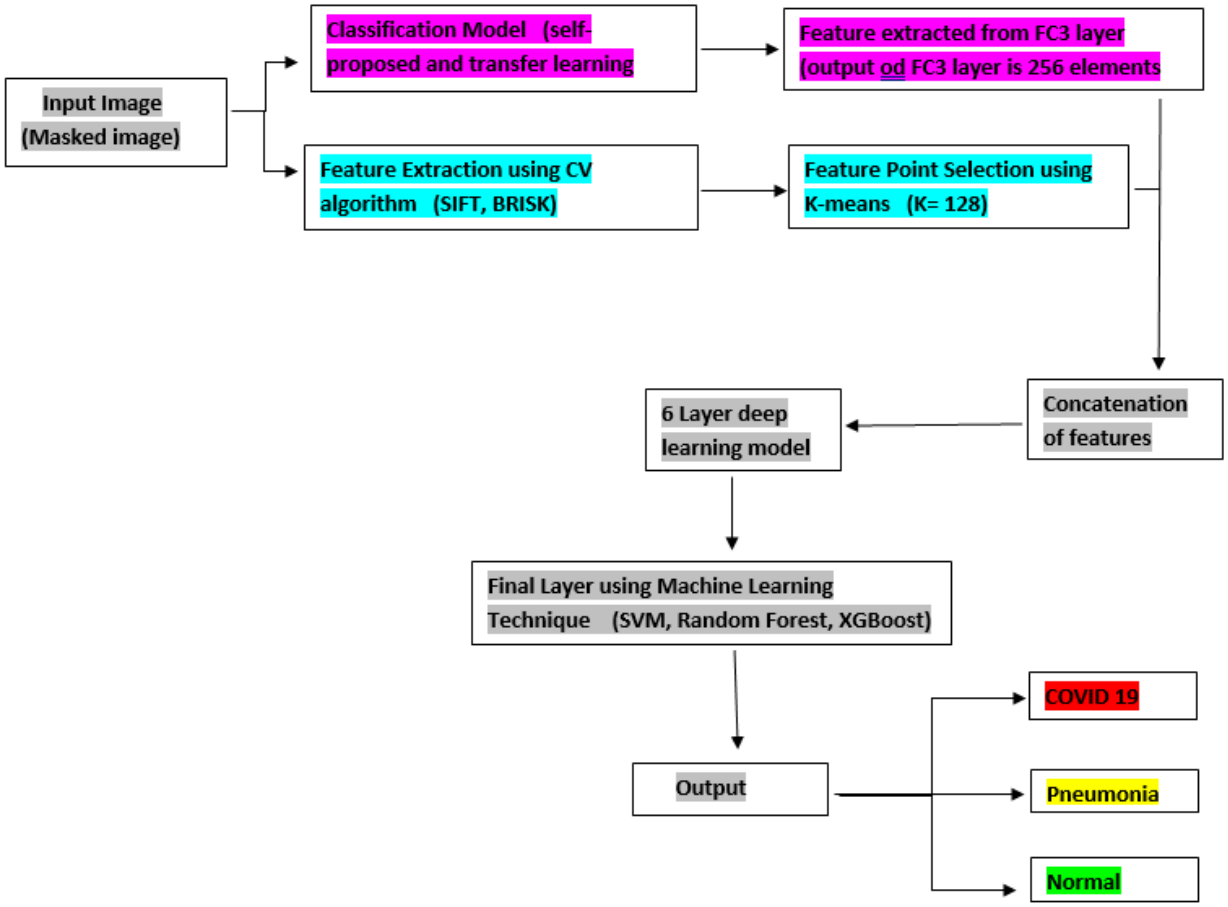
This dataset contains data from all above given sources.
And as per categories it contains COVID-19 – 722 images
Pneumonia – 1500 images
Normal – 1500 images

Dataset_2

This dataset contains best images from above all datasets.

Final Pipeline of Operation-





Pipeline of processing-

Xray image + mask

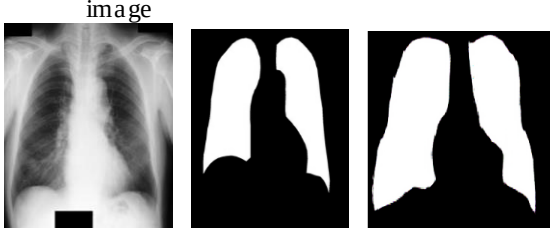
Concatenate

Masked Image

input image

Ground

Predicted



Generation of masked images

Predicted NAND/OR Xray Masked Image

Classification

We have proposed a particular pipeline for classification of Xray Images of a patient into three types. Namely COVID-19, Pneumonia, Normal. For this particular pipeline we have tested best performing pretrained classification models from ImageNet available in Keras Library. Also we have tested a self-defined Simple Convolution Network for the same purpose.

Name of the training model used

- Simple Convolution Network (self-defined)
- VGG-16
- VGG-19
- Xception
- DenseNet169
- DenseNet201
- InceptionResNetV2
- NASANetLarge

Some Predefined Variables

We have taken some fixed value of variables for training purpose

DATA type of trained images

- Maximum number of images for particular class for training is **700** due to RAM restriction.
- Test Size for validation data is **0.1%** of total data.
- For transfer learning models we have used include top = true and weights are downloaded from ImageNet.
- Fully connected dense layer used after flatten layer are **FC1, FC2, FC3** and **Output.**

	FC1	FC2	FC3	Output
Number of Perceptron	**4096**	**1024**	**256**	**3**
Activation Function	**relu**	**relu**	**relu**	**SoftMax**

- Optimizer - Adam Optimizer Learning rate=0.001

- Loss Function – Categorical Cross Entropy
- Matrix of evolution during training is accuracy.
- We have used pre-processed input functions related to every transfer learning model used ion this research.
- Call Back used
  Model Check Point
  ReduceLROnPlateau
- Epoch range (100-300)
- Feature Extraction using Computer Vision Libraries
  Feature extraction algorithm used SIFT, BRISK
  Threshold value = 50
  Clustering using K-means.
  Value of k = 128
- Combining all Processes
  Combining the outputs from K-means clustering and FC3 layer of trained models earlier .
  Total feature = 384
  Maximum input data for training = 2100
  Containing maximum of 700 training data per class.
  Total size = 0.1% of total data.

DL model contain total no. of 7 layers

Namely layer	Output Shape	Activation Function
Dense_1	**1024**	**Relu**
Dense_2	**800**	**Relu**
Dense_3	**512**	**Relu**
Dense_4	**300**	**Relu**
Dense_5	**256**	**Relu**
Dense_6	**128**	**Relu**
Output_layer	**3**	**SoftMax**

- Optimizer- Adam optimizer

  Learning rate- 0.001

  Loss function – Categorical cross entropy

  Matrix of evolution- accuracy

- Call-backs
  Model check point
  ReduceLROnPlateau
- Epoch range – (100-300)
- ML method use
  Output from Dense_6 layer = 128 are used as the input for the machine learning algorithms namely-
  1. SVM- Max_iter = 5000
  2. XGboost- Max_iter = 5000
  3. Random Forest- n_estimators = 1000

Observations- Observed accuracy percentage for different models is as given below:

S.No.	Models	Dataset_refined		Dataset_1		Dataset_2		New_dataset_2_all	
		BRISK	SIFT	BRISK	SIFT	BRISK	SIFT	BRISK	SIFT
1	DenseNet169	68%	73%	70%	67%	81%	85%	77%	75%
2	DenseNet201	71%	74%	75%	72%	77%	84%	73%	77%
3	InceptionResnetV2	79%	77%	70%	67%	78%	85%	73%	84%
4	NASANetLarge	75%	67%	69%	65%	66%	81%	66%	79%
5	Simple_Conv_Network	77%	80%	75%	72%	92%	90%	95%	95%
6	VGG16	90%	88%	72%	72%	96%	94%	89%	92%
7	VGG19	85%	86%	80%	78%	96%	96%	93%	94%
8	Xception	74%	73%	71%	72%	70%	81%	71%	81%

Fig.- Accuracy percentage for different models