

COVID-19 Screening Using Residual Attention Network an Artificial Intelligence Approach

Vishal Sharma

Department of Computer Science
Utah State University, Utah, USA
vishal.sharma@usu.edu

Curtis Dyreson

Department of Computer Science
Utah State University, Utah, USA
curtis.dyreson@usu.edu

Abstract—Coronavirus Disease 2019 (COVID-19) is caused by severe acute respiratory syndrome coronavirus 2 virus (SARS-CoV-2). The virus transmits rapidly; it has a basic reproductive number (R_0) of 2.2–2.7. In March 2020, the World Health Organization declared the COVID-19 outbreak a pandemic. COVID-19 is currently affecting more than 200 countries with 6M active cases. An effective testing strategy for COVID-19 is crucial to controlling the outbreak but the demand for testing surpasses the availability of test kits that use Reverse Transcription Polymerase Chain Reaction (RT-PCR). In this paper, we present a technique to screen for COVID-19 using artificial intelligence. Our technique takes only seconds to screen for the presence of the virus in a patient. We collected a dataset of chest X-ray images and trained several popular deep convolution neural network-based models (VGG, MobileNet, Xception, DenseNet, InceptionResNet) to classify the chest X-rays. Unsatisfied with these models, we then designed and built a Residual Attention Network that was able to screen COVID-19 with a testing accuracy of 98% and a validation accuracy of 100%. A feature maps visual of our model show areas in a chest X-ray which are important for classification. Our work can help to increase the adaptation of AI-assisted applications in clinical practice. The code and dataset used in this project are available at <https://github.com/vishalshar/covid-19-screening-using-RAN-on-X-ray-images>.

Index Terms—COVID-19 screening, Residual Attention Network, Deep Learning, Machine Learning

I. INTRODUCTION

In February 2013, some people in Guangdong province in China became infected with a severe acute respiratory syndrome virus (SARS-CoV) [1]. Eventually, SARS was detected in about 8000 patients across 26 countries, the World Health Organization (WHO) reported 774 deaths due to SARS [2]. In September 2012, a similar incident happened with the Middle East respiratory syndrome virus (MERS-CoV). There were 2494 confirmed cases of infection with 858 deaths due to MERS-CoV [3].

Both SARS and MERS pale in significance to the latest CoV outbreak concerning human health. In November 2019 pneumonia-like cases due to unknown causes started to appear in Wuhan, China killing hundreds of people in the initial weeks. In early 2020, the International Committee on the Taxonomy of Viruses (ICTV) declared the virus as Coronavirus Disease 2019 (COVID-19) caused by the SARS-CoV-2 virus [1]. The reproductive number (R_0) of COVID-19 is 2.2 – 2.7 [4] higher than SARS coronavirus due to a S protein in the RBD region of SARS-CoV-2 [5]. The highly transmissible virus quickly spread globally. By August 10,

2020 COVID-19 had been detected in 213 countries with six and a half million active cases and seven hundred and thirty thousand deaths. Due to a lack of medical supplies and staff, COVID-19 has overpowered the medical system of over 200 countries, and was declared a pandemic by World Health Organization (WHO) on March 11, 2020.

Diagnosing who has COVID-19 can help curb its spread by quarantining those infected. Currently, the most widely used technique for detecting COVID-19 is with viral nucleic acid detection using Reverse Transcription Polymerase Chain Reaction (RT-PCR), which works by detecting viral RNA from sputum or a nasopharyngeal swab [6]. Unfortunately, there is a shortage of RT-PCR test kits [7]. RT-PCR tests are also relatively slow, a test takes, at best, about four hours to complete, even in a highly controlled environment. RT-PCR tests also have a high false positive rate (about 30%) [8]. A common notable symptom of COVID-19 patients is difficulty breathing [9].

Recent advances in the field of computer vision suggest the possibility of a faster, more widely available alternative for detecting COVID-19. A CT scan of a patient's chest has shown higher accuracy and sensitivity than a RT-PCR test for COVID-19 detection [10]. Another study [11] further validates at least 20% higher detection sensitivity from CT scans versus RT-PCR tests. Identifying who has COVID-19 using an imaging technique is relatively non-intrusive, uses widely-available X-ray or CT scanners, and can be used at a very early stage to diagnose COVID-19 [12].

Deep neural networks have been successful in processing medical images [13] and image processing in general, for instance in object detection [14], image segmentation [15], and image classification [16]. Deep learning techniques such as, convolutional neural networks (CNN) are popular for processing medical images, *e.g.*, for classification [17] and segmentation [18]. Advances in CNN models over the past few years have led to robust implementations such as VGGNet [19], Inception [20], DenseNet [21], Xception [22], and MobileNet [23]. Recently, Attention Mechanism [24] which generates attention-aware features, based on spatial features has become popular in the fields of computer vision and image processing.

In this paper, we report on experiments with various deep learning models to detect and classify COVID-19 from chest X-ray images of patients. The models we use are

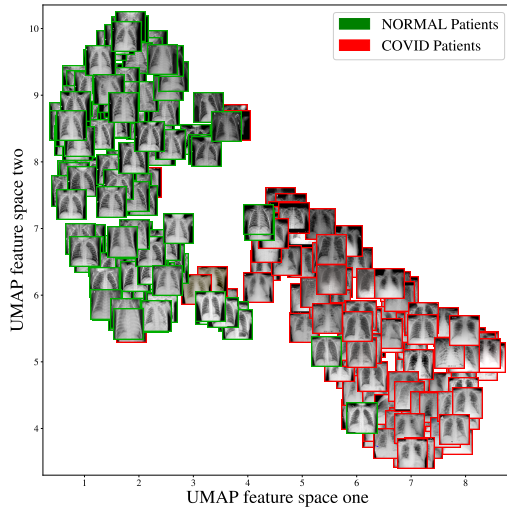


Fig. 1: UMAP features of chest X-ray images of patients in our dataset

VGG, ResNet, MobileNet, DenseNet, Xception, Attention, and Residual based CNN (Residual Attention Network). We first collected X-ray images of COVID-19 patients and people without the disease, which we call the “normal” class. It is important to note that the normal class can have other illnesses, such as pneumonia. This image dataset is from a diverse population in terms of location, age, and gender. To better understand the dataset, we analyzed it with a popular non-linear dimensionality reduction technique, Uniform Manifold Approximation and Projection (UMAP). The reduction produces a clear distinction between those with COVID-19 and those without (as shown in Figure 1). Next, we split the dataset into training, testing, and validation sets using stratified folds. We configured several deep learning models for optimal results and trained the models on the dataset. To improve the modeling, we designed and built our own Residual Attention Network with better feature extraction using custom designs of residual and attention block. Our model outperforms other models with 98% accuracy on the test set and 100% accuracy on the validation set. We extracted feature maps from our model and observed the Residual Attention Network detecting potential COVID-19 infected areas. The major challenge in this research is developing an effective model using only a very small dataset. We address this challenge by designing our Residual and Attention block to avoid overfitting. This paper makes the following contributions:

- a novel dataset of curated images for use in COVID-19 research,
- a problem-specific, highly accurate classification model using Residual Connection and Attention Mechanism,
- an explainable diagnosis using feature maps, and
- a reproducible dataset and classifier; all of our code and data are in the public domain.¹

This paper is organized as follows. Section II outlines related work and potential limitations. Section III describes

¹github.com/vishalshar/covid-19-screening-using-RAN-on-X-ray-images

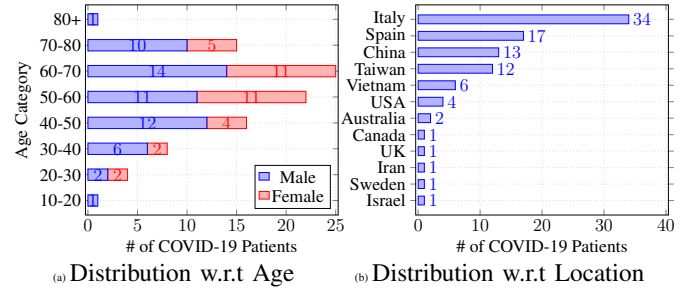


Fig. 2: Distribution of our dataset with positive COVID-19 based on location, gender and age

our dataset collection and analysis. Section IV describes our approach and design of our Residual Attention Network. Section V shows experimental configurations and reports results, and finally Section VI presents conclusions and future work.

II. RELATED WORK

There are over 24,000 research papers on COVID-19 from well known sources like *bioRxiv*, *arXiv* and *medRxiv*. More than 1,500 of these papers are peer reviewed [25]. There are two recent review papers about using AI techniques in COVID-19 detection [26], [27]. We focus on research that uses deep learning for COVID-19 detection. Wang et al. [28] used 1,065 chest CT scan images of COVID-19 patients to build a classifier using InceptionNet. They report an accuracy of 89.5%, a specificity of 0.88, and a sensitivity of 0.87. Butt et al. [29] used ResNet and reported an accuracy of 86.7%. Chen et al. [30] segment the infected areas in CT scans using UNet++ [31]. Using transfer learning and predefined models to classify COVID-19 in CT scans has also been researched, for instance using DenseNet *e.g.*, [32], [33], ResNet *e.g.*, [34], [35], and CNN *e.g.*, [34], [32], [36]. Traditional machine learning (ML) methods of feature extraction and conventional ML algorithms for classification have also been used. Mucahid et al. [37] used feature extraction techniques GLCM (grey level co-occurrence matrices), LDP (local directional pattern), GLRLM (grey-level run length matrix), and DWT (discrete wavelet transform), and using extracted features in a Support Vector Machine (SVM) for classification. They report an accuracy of 99.68% in the best configuration settings. Alqudah et al. [38] applied various ML techniques, such as SVM and Random Forest, and reported an accuracy of 95%.

To the best of our knowledge, we are the first to design and build a Residual Attention Mechanism to extract spatial-aware features and perform the classification of a COVID-19 dataset. We obtained higher test and validation accuracy, precision, recall, sensitivity, and specificity than previous work. Additionally, most of the previous research used more sophisticated CT scan images which usually take 20 to 30 minutes to perform, but we use X-ray images that are faster to extract, about 10 minutes in most cases. X-ray machines are more widely available than CT scanners, and there are portable X-ray units that can be deployed anywhere, not just in medical facilities.

III. DATASET

We collected images only from public sources, which provide the data while maintaining patients' privacy. The dataset of COVID-19 X-ray images comes from radiopaedia.org², the website of the Italian Society of Medical and Intervention Radiology³. Cohen et al. [39] scraped the images from the website using PDF processing tools. We selected 120 images of patients with COVID-19, specifically, we selected all *Posterioranterior* (PA) images. The PA view has an anterior aspect in which the ribs are much clearer than the *Anteriorposterior* or *Lateral* view X-ray. To collect images of non-COVID-19 (which we call normal) chests we randomly selected images from a repository collected by Mooney et al. [40]. In this repository, there are chest X-ray images of patients with pneumonia as well as of healthy patients. We extracted 119 PA view X-ray images of normal patients. In total, our dataset has 239 images with 120 from COVID-19 patients and 119 from normal patients.

A. Dataset Statistics

A wide distribution in a dataset is an important factor for training a deep learning model, since training on narrowly distributed data may lead to a biased model due to a failure to generalize the classification features. Figure 2 depicts our dataset distribution, with respect to location, gender, and age of patients. The figure shows that our image collection comes from 62 male and 39 female patients with a normal age distribution that is shifted towards elderly patients. These patients are from twelve countries, however, patient ethnicity was not made public.

Age: The dataset has patients from age 12 to 84, with an average age of 57.33 years. Figure 2a shows the amount of samples in the age category as well as the gender count in each category. Our dataset has a wide range of patients in terms of their ages.

Location: Figure 2b shows the location of patients. The location is an important attribute, since a model trained on data from only one country may become biased. Greater variation in the training data can help generalize a deep learning model. Our dataset has images of patients from twelve countries.

B. UMAP Exploration

We applied the Uniform Manifold Approximation and Projection (UMAP) technique, which is a non-linear dimensionality reduction technique, to the images. The feature space of UMAP is found by searching for a low-dimensional projection of data which is the closest equivalent to the real data using a fuzzy topological structure. We prepared the dataset for UMAP by performing a standard image pre-processing, as described further in Section V-A. Figure 1 shows the result of the UMAP. In the figure, the X-ray of a normal patient has a green bounding box while that of a COVID-19 patient has a red bounding box. The figure shows two clusters, one dominated by COVID-19 images, the other by normal images.

²radiopaedia.org

³<https://www.sirm.org/category/senza-categoria/covid-19/>

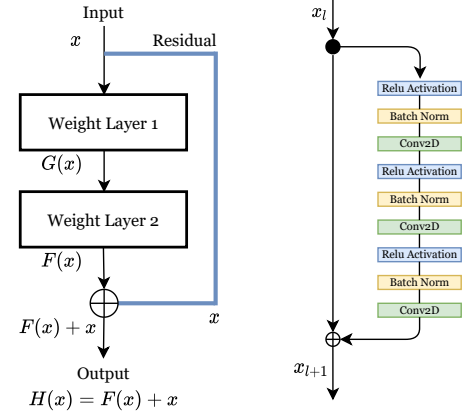


Fig. 3: Image on the left shows a Residual block and on the right shows full pre-activation used as our Residual block

IV. APPROACH

In this section, we explain our custom Residual block, Attention block, and how we used them to design and build a Residual Attention Network, which extends the original Residual Attention Network [41].

A. Residual Block

Deep convolution networks have revolutionized the field of image classification. Advancements in algorithms and hardware networks have increased the ability to add layers to a deep convolution network. With the increase in the depth of the network, it becomes harder to train a neural network because of vanishing gradients. Networks with too many layers become highly unstable as the value of gradients approaches zero in early layers. Every additional layer gradient value becomes smaller and eventually insignificant. Vanishing gradients degrade the performance of the network and adding more layers only exacerbates the problem. To solve the vanishing gradient problem, Kaiming He et al. [42] proposed *residual connections*. A residual connection merges the output of a layer with the input of a previous layer, which ensures that gradient values do not suddenly vanish. As shown in Figure 3, on the left is a residual block with a residual connection.

A deep learning model in general tries to learn a mapping function $H(x)$ from an input x to output y ,

$$H(x) = y \quad (1)$$

In a residual block, instead of learning a direct mapping, it uses the difference between the mapping of x and the original input x ,

$$F(x) = H(x) - x \quad (2)$$

re-arranging gives,

$$H(x) = F(x) + x \quad (3)$$

residual block learns the residual $F(x)$ with given x as an input and $H(x)$ as the true output. This technique helps when increasing the depth of a neural network.

Our experiments with arranging residual block for optimal gradient flow showed full pre-activation with batch normalization gives the best results, which was also suggested by Kaiming He et al. [42]. Our full pre-activation block is shown in the right of Figure 3, where we use Relu activation, batch normalization and a 2D convolution layer stacked three times. The sequence and stacking of our block are different than originally proposed [41], which was batch normalization, Relu activation and convolution layer stacked twice. From our experiments, we observed that batch normalization after Relu activation performs better. The reason for better performance happened when input features for a layer are negative, in the network they would have been truncated using non-linearity activation function, Relu, before batch normalization. Performing batch normalization prior to Relu activation will include these negative values in feature space.

B. Attention Block

The attention mechanism has become a very popular technique in natural language processing, image processing, and computer vision. The attention mechanism can generate attention-aware features and features that can be extracted based on spatial, context, or channel aware-features. It also learns the importance and correlations among features. Using visual attention in an image classification task helps determine important image regions and their correlations. The presence or absence of image regions is critical to classification. In our case, these regions are evidence of COVID-19 infection in chest X-ray images. Our attention module consists of two branches: a “trunk” branch $T(x)$ with two stacked residual blocks and an encoder-decoder “mask” branch $M(x)$.

The encoder in the mask branch consists of downsampling using max-pooling followed by residual connection and down-sampling again. The encoder acts as an input reducer. The decoder consists of upsampling using bilinear interpolation. In the original Residual Attention Network [41] there is only one upsample but, in a literature survey we found that the performance of a Residual Attention Network can be increased by increasing the number of up-sampling layers [43]. We extended the Attention block by adding two upsample layers in our model. The encoder and decoder are followed by two convolution layers and a sigmoid activation as displayed in Figure 4. The trunk branch consists of two stacked residual blocks that perform feature processing.

The final output of the module is

$$G(x) = (1 + M(x)) * T(x) \quad (4)$$

adding 1 to the equation ensures that in case of mask branch with zero output the trunk branch computation passes through, which dampens the susceptibility to noisy data.

C. Residual Attention Network

Our Residual Attention Network is built by multiple stacks of our basic unit module. The stacking of blocks is designed for optimal performance and to prevent overfitting. The Attention block is designed to explore fine-grained feature maps

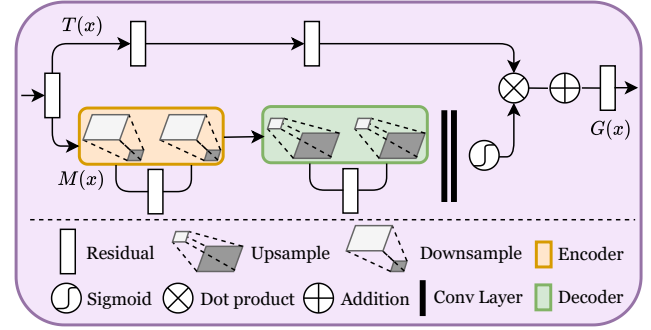


Fig. 4: Design of Attention Block

since COVID-19 infections could be a fine detail in an X-ray. There are two major attention categories: *Soft* and *Hard*, we use *Soft* attention to learn alignment for several patches.

The Residual block captures high-level features and provides input to Attention block. The Attention module generates specialized low-level features on Residual input. It divides an image into a few high-level features and from those features extracts several low-level features. We stack Residual and Attention layers alternatively three times. The Residual layer extracts high-level features from the input image which are then passed to the Attention block, which extracts low-level features. These low-level features become an input to the next Residual block. It works as both a top-down and bottom-up approach, the top-down network produces dense features and the bottom-up one produces low-resolution feature maps. Our architecture is shown in Table I. This technique has proved successful in image segmentation [44]. Our scenario is very similar to segmentation where we try to identify low-level patches of COVID-19 infections in a chest X-ray.

V. EXPERIMENTS

This section presents an experimental evaluation of our model. We start with data preprocessing and stratified data split for training, testing, and validation. We describe the experimental setup, our models, and their configurations, and we show the results.

A. Data Preprocessing

Data preprocessing plays an important role in training a deep learning model. Previous research has highlighted the impact of preprocessing on model performance [45]. Images in our dataset have different sizes, so we first start with standardizing the image size to 224×224 pixels. The collected images also have different color patterns, so we normalize the color pattern to RGB. Lastly, we normalize the maximum intensity of a pixel to 255 (lowest is 0). Our dataset has two different classes, labeled *COVID*, and *Normal*; during training, we use a label binarizer to convert them to one-hot encoding.

1) *Data Split*: The dataset is split into training (70%), testing (20%), and validation (10%) sets. We use stratified splits which ensures that each split has an equal number of samples from each class as shown in Table II. We also add a

Layers	Output Size	Kernel Size
Convolution 2D	112×112	(5×5), $p=\text{same}$
MaxPool 2D	56×56	(2×2), 2
Residual Block	56×56	$\begin{pmatrix} 1 \times 1 & 32 \\ 3 \times 3 & 32 \\ 1 \times 1 & 128 \end{pmatrix}$
Attention Block	56×56	Attention×1
Residual Block	28×28	$\begin{pmatrix} 1 \times 1 & 128 \\ 3 \times 3 & 128 \\ 1 \times 1 & 256 \end{pmatrix}$
Attention Block	28×28	Attention×1
Residual Block	14×14	$\begin{pmatrix} 1 \times 1 & 256 \\ 3 \times 3 & 256 \\ 1 \times 1 & 512 \end{pmatrix}$
Attention Block	14×14	Attention×1
Residual Block	7×7	$\begin{pmatrix} 1 \times 1 & 512 \\ 3 \times 3 & 512 \\ 1 \times 1 & 1024 \end{pmatrix}$
Residual Block	7×7	$\begin{pmatrix} 1 \times 1 & 1024 \\ 3 \times 3 & 1024 \\ 1 \times 1 & 1024 \end{pmatrix}$
Residual Block	7×7	$\begin{pmatrix} 1 \times 1 & 1024 \\ 3 \times 3 & 1024 \\ 1 \times 1 & 1024 \end{pmatrix}$
AvgPooling 2D	1×1	(7×7)
FC, Softmax		2
Depth		115

TABLE I: Residual Attention Network architecture

random rotation of 15° to images, which adds more stability to our model during training.

	COVID	Normal	Total
Training	84	83	167
Testing	25	25	50
Validation	11	11	22
	120	119	239

TABLE II: COVID dataset splits using stratified sampling

B. Experimental setup

All experiments were carried out on a computer with Intel i7 5820k, Nvidia 1080ti, 16GB of RAM running Ubuntu 18.04 OS. We use Python 3.7 and its libraries (sklearn, tensorflow, keras) to write and train the deep learning models. All of the networks were trained on Nvidia 1080ti (3584 CUDA cores and 11GB DDR5 RAM) with CUDA and cuDNN configured for performance enhancement.

1) *Baseline*: We use the following models as baseline for our Residual Attention Network:

VGGNet: VGGNets were introduced in 2014 with the intention to perform state of the art image classification with the least depth of CNN possible. We use both VGG16 and VGG19 in our experiments.

ResNet: ResNet introduced the concept of residual connection to solve the vanishing gradient problem. Deciding a kernel size for ResNet is hard and that's why we use InceptionResNet a variant of ResNet, which uses multiple size kernels within the same layer.

Xception: Xception introduced the concept of depthwise separable convolution to reduce the number of the parameter without loss of performance.

MobileNet: In addition to depthwise convolution, MobileNet introduced pointwise convolution to reduce the number of parameters in order of 100 to 1000. We specifically use MobileNet and its variant MobileNetV2.

DenseNet: DenseNet extends the idea of residual from ResNet, but instead of learning residual (the difference between previous and current layer) it proposes to merge the output of the previous and current layer. We use DenseNet121 and DenseNet201 in our experiments.

NASNet: NASNet is Neural Architecture Search Network. They are a family of models designed to learn model architecture automatically on the dataset of interest. In our case *imagenet* was used as dataset to design NASNet.

Vanilla Residual Attention Network (RAN): To compare our work with existing state-of-the-art Residual Attention Network, we also implemented Attention-56 which was proposed in [41].

2) *Evaluation Metric*: To evaluate the performance of our models, we use the most commonly used performance metrics for deep learning, namely, sensitivity, specificity, precision, recall, and accuracy. Their value range is $[0, 1]$: higher is better. The metrics are given below, where TP is true positive, FP is false positive, TN is true negative, and FN is false negative.

$$\text{Sensitivity (Sens)} = \frac{TP}{TP + FN}$$

$$\text{Specificity (Spec)} = \frac{TP}{FP + TN}$$

$$\text{Precision (Prec)} = \frac{TP}{TP + FP}$$

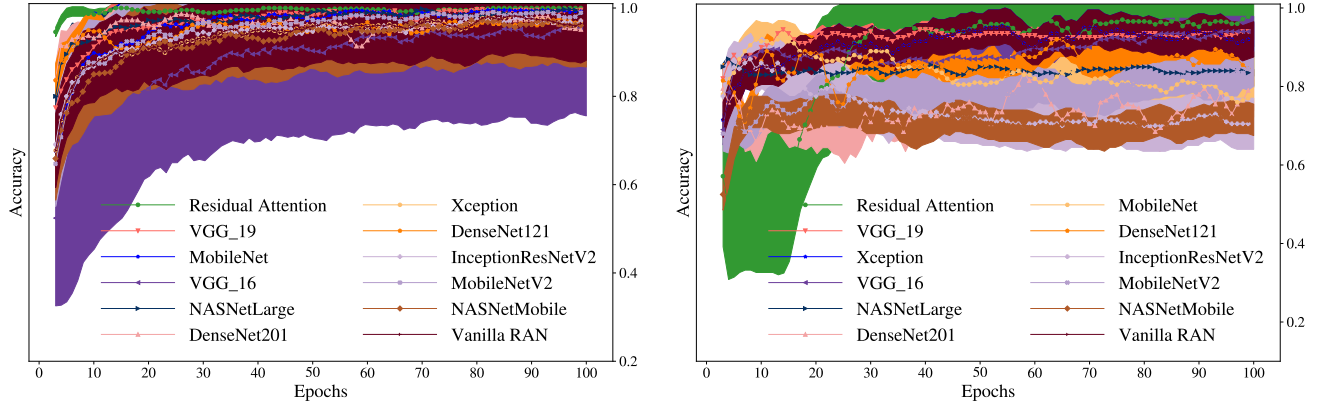
$$\text{Recall (Rec)} = \frac{TP}{TP + FN}$$

$$\text{Accuracy (Acc)} = \frac{TP + TN}{TP + FP + TN + FN}$$

3) *Models configurations*: Our selection of model size ranges from 15 layers to 270 layers deep. We trained each model with an initial learning rate of $1e-4$ and a mini-batch size of 8 with 100 epochs. We used Adam as our optimizer. Adam optimizer uses two popular optimization techniques, Root Mean Square Propagation (RMSprop) and Stochastic Gradient Descent (SGD), with momentum. To provide stability during the training of our models, we used a learning rate decay with the decay rate shown in Equation 5. During the training, we used Binary Cross entropy as the loss function for all of the models. For all benchmark models, we use them with pre-trained weights using *imagenet*, which contains 14 million images with over 1000 classes. Training on a large dataset requires massive computational power, for example, NASNetLarge was trained using 500 GPUS for four days on the *imagenet* dataset. We used the weights of models after the training and retrained them on our dataset. This methodology is commonly known as *transfer learning*.

$$\text{Decay Rate} = \frac{\text{Initial Learning Rate}}{\text{Epochs}} \quad (5)$$

4) *Output head modification*: The predefined models have been designed and trained for the very large *imagenet* dataset.



Accuracy over epochs of models on training set

Accuracy over epochs of models on testing set

Fig. 5: Training and testing accuracy for all models. The shaded region represents 95% confidence interval.

Network	Testing Set						Validation Set					
	Layers	Sens	Spec	Prec	Rec	Acc	Sens	Spec	Prec	Rec	Acc	AUC
NASNetMobile	198	0.72	0.68	0.69	0.70	0.7000	0.82	0.55	0.64	0.75	0.6818	0.86
MobileNetV2	54	0.72	1.00	1.00	0.78	0.8600	0.55	0.91	0.86	0.67	0.7273	0.72
InceptionResNetV2	246	0.40	1.00	1.00	0.63	0.7000	0.55	1.00	1.00	0.69	0.7727	0.72
DenseNet201	202	0.52	1.00	1.00	0.68	0.7600	0.82	0.91	0.90	0.83	0.8182	0.82
DenseNet121	122	0.92	0.72	0.77	0.90	0.8200	0.82	0.82	0.82	0.82	0.8636	0.90
Xception	42	0.84	1.00	1.00	0.86	0.9200	0.91	0.82	0.83	0.90	0.8636	0.88
NASNetLarge	270	0.72	0.92	0.90	0.77	0.8200	0.82	0.91	0.90	0.83	0.8636	0.90
MobileNet	29	0.56	1.00	1.00	0.69	0.7800	0.82	1.00	1.00	0.85	0.9091	0.94
Vanilla Residual Att Net	145	0.88	0.92	0.92	0.88	0.9000	0.91	0.91	0.91	0.91	0.9091	0.94
VGG16	15	1.00	0.88	0.89	1.00	0.9400	1.00	0.82	0.85	1.00	0.9091	0.96
VGG19	18	0.96	0.92	0.92	0.96	0.9400	0.92	1.00	1.00	0.92	0.9545	0.98
Residual Att Net (Our)	115	1.00	0.96	0.96	1.00	0.9800	1.00	1.00	1.00	1.00	1.0000	1.00

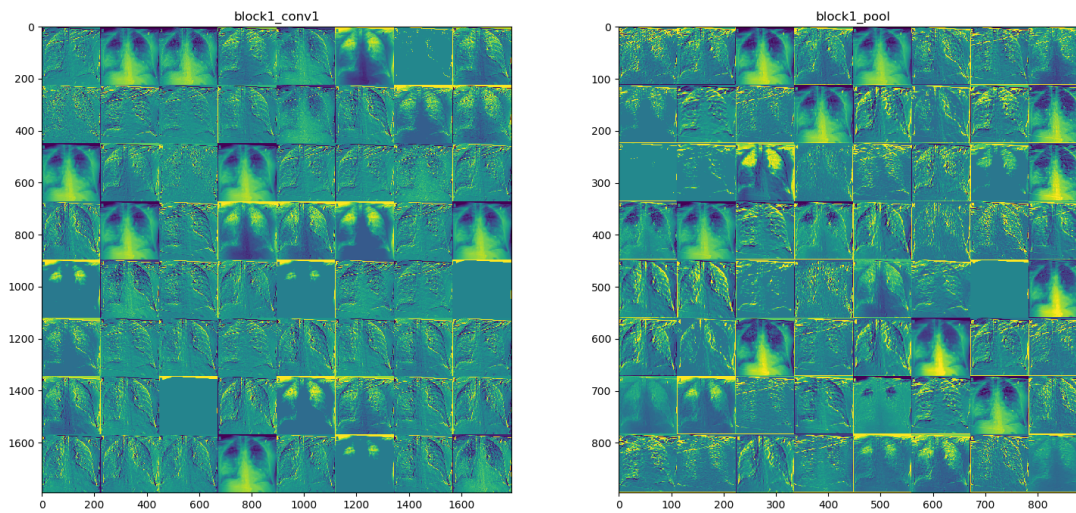
TABLE III: Sensitivity, Specificity and Accuracy from all models on Testing and Validation set (sorted based on validation accuracy)

This dataset consists of 1000 classes and millions of images whereas our dataset has two classes and 239 images. To prevent overfitting of the predefined models on our small dataset, we modified the output layers of all of the models. We removed the output layer and added a custom output head. For example, VGG16 has three fully connected layers as the output head, the first two layers have 4096 neurons while the third has 1000 (number of classes) neurons. We modified this output from three layers to two layers with 64 neurons in the first layer and 2 (number of classes) neurons in the second layer. We did this on all of the models to standardize the output head.

C. Results

Table III displays the accuracy of the models on the testing and validation sets of our COVID-19 dataset. We observe that our custom Residual Attention Network outperforms all other deep learning models, while NASNetMobile turns out to be the worst performer. Our experiments show that VGG19 performs better than VGG16. Even though both models are

similar, VGG19 has more layers and it is commonly thought that deeper pre-trained models perform better than shallow models. But that is not the case for our dataset, for instance, the network with a shallower depth, DenseNet121, performed better than a very deep network, DenseNet201 (both networks are at least six times deeper than VGG19). From this, we observe that it boils down to feature maps of layers from a model, which is dependent on the convolution layers. We also observe that very deep networks do not perform better than shallower ones, since the top two performing networks are shallow networks (1/4 of the size of the largest network). Figure 5 shows the model training and testing accuracy for every epoch. We observe large/deeper models (DenseNet201, NASNetLarge, InceptionResNetV2) tend to overfit and shallower models (VGG16, VGG19, Residual Attention Network) are able to generalize our dataset better. We also observe our residual attention model during testing took 15 epochs to start improving accuracy. This can be explained by our layers weight initializer using Xavier where it initializes the weights of the layer with zero mean and unit variance.



(a) Feature Map at Convolution Layer 1

(b) Feature Map at pooling Layer

Fig. 6: Feature maps of Residual Attention Network at first 2 Layers

Overall, our experiments show that the Residual Attention Network performs best: with 98% accuracy on testing and 100% on validation to screen COVID-19 in X-ray images.

COVID-19 prediction explanation: Figure 6 visualizes the feature maps of the first two layers of our Residual Attention Network model trained on our COVID-19 dataset. There are 64 images in each picture with an 8×8 grid for each image. The feature maps were generated by passing an image to our trained model and collecting data of activated neurons. We select only the first two layers and 64 neurons of each layer, since the number of neurons and parameters grows exponentially at later layers. We observe from the feature maps of both layers that our model can extract relevant details from an image, e.g., lungs, spine, veins, potential COVID-19 affected areas.

VI. CONCLUSION

This paper proposes a novel method to detect COVID-19 using chest X-ray images. The method uses a Residual Attention Network and data augmentation. We collected and curated a dataset of 239 images: 120 images of patients infected with COVID-19, and 119 images of non-COVID-19 patients, which we label “normal.” The dataset is diverse in terms of patient age, gender, and location. We applied a non-linear dimensionality reduction technique, UMAP, and observed a clear distinction between X-rays of COVID-19 and normal patients. We designed and implemented a Residual Attention Network to classify COVID-19 patients and compared our model with many popular deep learning models: VGG, DenseNet, NASNet, Xception, and Inception. Our experiments show that our custom Residual Attention Network performs best among all of the models with 98% testing and 100% validation accuracy. We generated feature maps of the Residual Attention Network and they show that

the low-level features extracted from a given image include areas of potential COVID-19 infection.

This research shows that chest X-ray images can potentially be used to detect COVID-19, or can be combined with other testing methods to corroborate a diagnostic outcome. The immediate future work is to add more images to our dataset. The second avenue of future work is to work with domain experts to study the utility of deploying the technique in practice. Due to the recent nature of the COVID-19 pandemic, the deployment has lagged behind the research. Another aspect of future work is to study whether chest X-rays can be used to detect COVID-19 early in the illness or in asymptomatic cases. Our X-ray dataset does not have information about patient symptoms or time with the disease, only that the X-rays are of COVID-19 positive patients. Since early detection is important to effective quarantining, it would be interesting to test our model on X-rays from patients who have just been infected or who are asymptomatic.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under Award No. 1759965, *Collaborative Research: ABI Development: Symbiota2: Enabling greater collaboration and flexibility for mobilizing biodiversity data*. Opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the National Science Foundation.

REFERENCES

- [1] T. Singhal, “A review of coronavirus disease-2019 (covid-19),” *The Indian Journal of Pediatrics*, vol. 87, 03 2020.
- [2] *Summary of probable SARS cases with onset of illness from 1 November 2002 to 31 July 2003*, 2004. [Online]. Available: https://www.who.int/csr/sars/country/table2004_04_21/en/

- [3] *Middle East respiratory syndrome coronavirus (MERS-CoV)*, 2013. [Online]. Available: <https://www.who.int/emergencies/mers-cov/en/>
- [4] S. Steven, Y. T. Lin, C. Xu, E. Romero-Severson, N. Hengartner, and R. Ke, "High contagiousness and rapid spread of severe acute respiratory syndrome coronavirus 2," in *Center for Disease Control and Prevention (CDC)*, 2020. [Online]. Available: <https://doi.org/10.3201/eid2607.200282>
- [5] M. Shereen, S. Khan, A. Kazmi, N. Bashir, and R. Siddique, "Covid-19 infection: Origin, transmission, and characteristics of human coronaviruses," *Journal of Advanced Research*, vol. 24, 03 2020.
- [6] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao, J. Guo, M. Cai, J. Yang, Y. Li, X. Meng, and B. Xu, "A deep learning algorithm using ct images to screen for corona virus disease (covid-19)," *medRxiv*, 2020. [Online]. Available: <https://www.medrxiv.org/content/early/2020/04/24/2020.02.14.20023028>
- [7] K. G. Malone, *Testing backlog linked to shortage of chemicals needed for covid-19 test*, March 25, 2020. [Online]. Available: <https://bit.ly/2SEXzVY>
- [8] Y.-H. Wu, S.-H. Gao, J. Mei, J. Xu, D.-P. Fan, C.-W. Zhao, and M.-M. Cheng, "Jcs: An explainable covid-19 diagnosis system by joint classification and segmentation," 2020.
- [9] J. Zhang, Y. Xie, Y. Li, C. Shen, and Y. Xia, "Covid-19 screening on chest x-ray images using deep learning based anomaly detection," *arXiv preprint arXiv:2003.12338*, 2020.
- [10] 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*, p. 200642, 2020, pMID: 32101510. [Online]. Available: <https://doi.org/10.1148/radiol.20200642>
- [11] Z. Huang, S. Zhao, Z. Li, W. Chen, L. Zhao, L. Deng, and B. Song, "The battle against coronavirus disease 2019 (covid-19): Emergency management and infection control in a radiology department," *Journal of the American College of Radiology*, 2020. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1546144020302854>
- [12] X. Xu, X. Jiang, C. Ma, P. Du, X. Li, S. Lv, L. Yu, Y. Chen, J. Su, G. Lang, Y. Li, H. Zhao, K. Xu, L. Ruan, and W. Wu, "Deep learning system to screen coronavirus disease 2019 pneumonia," 2020.
- [13] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in neural information processing systems*, 2014, pp. 3320–3328.
- [14] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Advances in neural information processing systems*, 2013, pp. 2553–2561.
- [15] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
- [16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
- [17] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, "Medical image classification with convolutional neural network," pp. 844–848, 2014.
- [18] B. Kayalibay, G. Jensen, and P. van der Smagt, "Cnn-based segmentation of medical imaging data," *arXiv preprint arXiv:1701.03056*, 2017.
- [19] S. Liu and W. Deng, "Very deep convolutional neural network based image classification using small training sample size," in *2015 3rd IAPR Asian conference on pattern recognition (ACPR)*. IEEE, 2015, pp. 730–734.
- [20] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CoRR*, vol. abs/1409.4842, 2014. [Online]. Available: <http://arxiv.org/abs/1409.4842>
- [21] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016. [Online]. Available: <http://arxiv.org/abs/1608.06993>
- [22] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1251–1258.
- [23] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
- [24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.
- [25] M. T. Review, "Over 24,000 coronavirus research papers are now available in one place," 2020. [Online]. Available: <https://www.technologyreview.com/2020/03/16/905290/coronavirus-24000-research-papers-available-open-data/>
- [26] F. Shi, J. Wang, J. Shi, Z. Wu, Q. Wang, Z. Tang, K. He, Y. Shi, and D. Shen, "Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for covid-19," *IEEE Reviews in Biomedical Engineering*, 2020.
- [27] L. Wynants, B. Van Calster, M. M. Bonten, G. S. Collins, T. P. Debray, M. De Vos, M. C. Haller, G. Heinze, K. G. Moons, R. D. Riley *et al.*, "Prediction models for diagnosis and prognosis of covid-19 infection: systematic review and critical appraisal," *bmj*, vol. 369, 2020.
- [28] 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)," *MedRxiv*, 2020.
- [29] C. Butt, J. Gill, D. Chun, and B. A. Babu, "Deep learning system to screen coronavirus disease 2019 pneumonia," *Applied Intelligence*, p. 1, 2020.
- [30] J. Chen, L. Wu, J. Zhang, L. Zhang, D. Gong, Y. Zhao, S. Hu, Y. Wang, X. Hu, B. Zheng *et al.*, "Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high-resolution computed tomography: a prospective study," *medRxiv*, 2020.
- [31] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," *CoRR*, vol. abs/1807.10165, 2018. [Online]. Available: <http://arxiv.org/abs/1807.10165>
- [32] L. Li, L. Qin, Z. Xu, Y. Yin, X. Wang, B. Kong, J. Bai, Y. Lu, Z. Fang, Q. Song *et al.*, "Artificial intelligence distinguishes covid-19 from community acquired pneumonia on chest ct," *Radiology*, p. 200905, 2020.
- [33] X. Li and D. Zhu, "Covid-xpert: An ai powered population screening of covid-19 cases using chest radiography images," *arXiv preprint arXiv:2004.03042*, 2020.
- [34] O. Gozes, M. Frid-Adar, N. Sagie, H. Zhang, W. Ji, and H. Greenspan, "Coronavirus detection and analysis on chest ct with deep learning," *arXiv preprint arXiv:2004.02640*, 2020.
- [35] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks," *arXiv preprint arXiv:2003.10849*, 2020.
- [36] K. E. Asnaoui, Y. Chawki, and A. Idrı, "Automated methods for detection and classification pneumonia based on x-ray images using deep learning," *arXiv preprint arXiv:2003.14363*, 2020.
- [37] M. Barstugan, U. Ozkaya, and S. Ozturk, "Coronavirus (covid-19) classification using ct images by machine learning methods," *arXiv preprint arXiv:2003.09424*, 2020.
- [38] A. M. Alqudah, S. Qazan, H. Alquran, I. A. Qasmieh, and A. Alqudah, "Covid-2019 detection using xray images and artificial intelligence hybrid systems," 2020.
- [39] J. P. Cohen, P. Morrison, and L. Dao, "Covid-19 image data collection," *arXiv 2003.11597*, 2020. [Online]. Available: <https://github.com/ieee8023/covid-chestxray-dataset>
- [40] P. Mooney, *Chest X-Ray Images (Pneumonia)*, June 2019. [Online]. Available: <https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia>
- [41] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3156–3164.
- [42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: <http://arxiv.org/abs/1512.03385>
- [43] L. Liang, J. Cao, X. Li, and J. You, "Improvement of residual attention network for image classification," in *International Conference on Intelligent Science and Big Data Engineering*. Springer, 2019, pp. 529–539.
- [44] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *CoRR*, vol. abs/1605.06211, 2016. [Online]. Available: <http://arxiv.org/abs/1605.06211>
- [45] S. Crone, S. Lessmann, and R. Stahlbock, "The impact of preprocessing on data mining: An evaluation of classifier sensitivity in direct marketing," *European Journal of Operational Research*, pp. 781–800, 09 2006.