PARTICULATE MATTER LEVELS CLASSIFICATION USING MODIFIED AND COMBINED RESNET MODELS WITH LOW FEATURES EXTRACTION

Rayan Awni Matloob

University of Duhok, College of Engineering, Electrical and Computer Department, Zakho Street 38, Kurdistan Region (Iraq).

https://orcid.org/0000-0001-7406-8196

Mohammed Ahmed Shakir*

University of Duhok, College of Engineering, Electrical and Computer Department, Zakho Street 38, Kurdistan Region (Iraq)

Mohammed.shakir@uod.ac

Reception: 29/01/2023 **Acceptance**: 09/04/2023 **Publication**: 28/04/2023

Suggested citation:

A.M., Rayan and A.S., Mohammed. (2023). **Particulate Matter Levels Classification Using Modified and Combined ResNet Models with Low Features Extraction**. *3C TIC. Cuadernos de desarrollo aplicados a las TIC, 12(1)*, 378-398. https://doi.org/10.17993/3ctic.2023.121.378-398

ABSTRACT

Smog is a serious environmental problem. It is an atmospheric pollutant that, if inhaled frequently, can lead to lung diseases such as asthma and bronchitis. One of the most dangerous air pollutants is particulate matter with a diameter of fewer than 2.5 micrometers (PM2.5), which may be breathed into the body and cause major health issues by introducing dangerous compounds deep into the lungs and bloodstream. In this research, a new convolutional neural network is proposed, by upgrading and parallelly stacking the two pre-trained models ResNet18 and ResNet50 to form a new modified-combined convolutional model (C-DCNN). Besides, we stacked another two columns of layers to extract the low features of ResNet18 and ResNet50 separately, to create finally four stacked columns of layers. The new model classifies images into different classes based on their PM2.5 concentration levels. To assess the suggested approach, an image augmentation is applied, then divided the images randomly (80% for the training progress,20% of the used training data for validation, and 20% for testing). The experimental results demonstrate that the proposed method increased the accuracy of level estimation with an accuracy increment equal to (6.25% at LR=0.0007) compared to ResNet50.

KEYWORDS

Deep Learning, Combined Convolutional Neural Network, ResNet, Image Classification, Air Quality, Particulate Matter.

https://doi.org/10.17993/3ctic.2023.121.378-398

PAPER INDEX

https://doi.org/10.17993/3ctic.2023.121.378-398

1. INTRODUCTION

The act of polluting the air with toxins that are harmful to living beings and humans, or that degrade the environment or materials is referred to as "air pollution" [1]. Humans may suffer from sicknesses or allergies, or even pass away as a result of it. In addition, it can impact the built environment (as a result of factors like ozone depletion, climate change, or habitat degradation) as well as other living things like food crops and animals, such as acid rain [2]. With anthropogenic ozone and PM2.5 causing 2.1 million fatalities per year, outdoor air pollution from the combustion of fossil fuels alone is one of the leading causes of mortality in people [3, 4]. Particles or as also called Particulate matter are minute solid or liquid particles suspended in a gas, commonly known as (PM), atmospheric particulate matter (APM), or fine particles [4]. The 2.5 refers to the size range of particles. A particle considered to be PM2.5 has a diameter between 1 and 2.5 micrometers (which is about 1/30th of the width of a human hair) [3, 4]. Some of the biggest cities such as China and Taiwan installed PM2.5 monitoring stations, but due to the expensive and the required resources it is not always the best idea, due to the need of installing more than one station for the large cities due to coverage limitation. The use of deep learning to forecast air quality by picture classification or regression is one of the better methods in this sector. Image-based automated information extraction has been the subject of a lot of work in machine learning and computer vision. The use of the image is an efficient and easy method because of the widespread of smartphones and the ability of every person to capture an image whenever he wants and wherever he is. This study introduced an image-based PM2.5 analysis technique that uses a deep learning network to categorize the PM2.5 concentration levels of outdoor images. The method suggested in this study leverages the cutting-edge CNN algorithm for image analysis in contrast to image feature-based PM2.5 analysis methodologies. Due to the CNN's explicit end-to-end design and ability to automatically extract both low-level and high-level picture characteristics. The Shanghai dataset (1052 photos, one scene) from [5] was used to test our methodology. The rest of this study is divided into the following sections. We describe some earlier work on machine learning for image categorization in Section 2. We discuss the contributions of our study and the provisions relating to the experiments in Section 3 of this article. We provide our suggested approach to concatenated neural convolution networks in Section 4. The experimental findings in Section 5 demonstrate how well our strategy performed. In Section 6, we draw to a close this essay, and finally, the references in Section 7.

2. RELATED WORKS

In [6] they merged meteorological data and images to predict PM2.5 indices of outdoor photos, using support vector regression (SVR) and deep learning techniques. Their suggested approach employs two datasets gathered from Beijing and Shanghai city in China besides a constructed SVR model to integrate the PM2.5 predicted by the CNN with two meteorological parameters, wind speed, and humidity, to provide the expected outcomes towards the end of the PM2.5 index. For the Shanghai

https://doi.org/10.17993/3ctic.2023.121.378-398

dataset, the proposed model reduced the RMSE by 26.08% and the R-squared increased by 24.57%. For the Beijing dataset, the proposed model reduced the reduced RMSE by 5.27% to 56.03 and increased the R-squared by 8.4% to 0.6046.

The authors of [7], Using an ensemble of deep neural network-based regression, the researchers estimated the PM2.5 concentrations based on photos taken outside. Using a feedforward neural network and a dataset of 1460 pictures for performance analysis, they merged the PM2.5 predictions from three convolutional neural networks ResNet50, Inception-v3, and VGG-16, to generate the final PM2.5 forecast of the image. As a consequence of their experimental findings, which show that the meta trainer can effectively aggregate the PM2.5 predictions from base learners and provide a better prediction than any single base learner utilized, the suggested technique is suitable for monitoring PM2.5 pollution. Based on a study of a substantial number of outdoor pictures accessible for Beijing, Shanghai (China), and Phoenix, the researchers calculated PM air pollution. Six picture elements were taken from the photographs and combined with additional pertinent information, such as the sun's position, the date, the time, the location, and the weather, to forecast the PM2.5 index.

The researchers in [8], have proposed an image-based deep learning model (CNN-RC, under VGG schemes and ResNet with some layers). That combines a convolutional neural network with a regression classifier. By shots extraction feature and feature categorization into air quality categories. This model is capable of calculating the air quality at specified places. The models were tested after training on datasets, comprising different combinations of the current image, HSV (hue, saturation, value), characteristics, and the baseline image, to boost model dependability and estimation accuracy. The Linyuan air quality monitoring station in Kaohsiung City, Taiwan, collected a total of 3549 hourly air quality datasets, including images, PM2.5, and the air quality index (AQI), to quickly produce an accurate image-based estimation of multiple pollutants at once using just one deep learning model. According to their test findings, the estimation accuracy for R2 for PM2.5 using day (night) photos is 76%.

In the manuscript of [9], they integrated two deep convolutional neural networks (DCNNs) utilizing transfer learning to extract distinctive picture properties, running the previously trained Inception and Xceptions models concurrently. Before feeding the final fully linked layers for classification, the feature maps are merged and reduced by dropout. The system uses maximum likelihood and majority voting criteria to classify sub-images first, then the entire picture. Breast cancer is classified as having four tissue malignancy levels: invasive carcinoma, in situ carcinoma, benign, and normal. The experiments were conducted using the BACH, Breast Cancer Histology dataset, and they used 4800 photos to obtain an accuracy of roughly 95%.

2.1. COMPARISON WITH OTHER MACHINE LEARNING WORKS

Predicting the air quality using image-based machine learning has a huge role, as it reduces the dependence on the huge amount and expensive equipment. In addition to being available at hand, for ease of use by smartphone devices. The accuracy of any image-based CNN model depends on the size of the used dataset (is it enough to learn the model or not?) and how good the model is for training correctly. Besides, the number of classes for classifying the problem at hand is important with the classification model. For the same dataset, increasing the number of classes used means decreasing the number of images in each class, which may reduce the accuracy or cause an overfitting problem. The results comparison demonstrates that the proposed new model significantly achieved reasonable accuracy increment compared to the ResNet50, despite the small dataset and the number of classes that are categorized by it. It is noteworthy that the used models in [6], and [9] have higher accuracy as they are owning a large number and a clear difference of images in the dataset. Any increment in the number of images per class has a significant effect on increasing the accuracy as clarified in.

3. CONTRIBUTIONS AND MATERIAL

3.1. CONTRIBUTIONS

The following is a summary of our work's significant contributions:

- This paper uses two pre-trained CNN models for air quality (PM2.5 levels) using one scene image. Instead of creating a new model and solving the issue of the little amount of accessible dataset, this study tries to improve the performance of learning.
- The main goal of this study is to provide a comprehensive classification that considers the following five categories of pollutants: Level 1, Level 2, Level 3, Level 4, and Level 5.
- This work develops a model based on parallel convolutional neural networks, to consolidate the machine learning training process. This architecture assembles the architecture of two conventional networks: ResNet18 and ResNet50 models.

As far as we are aware, no study has been done using ResNet18 and ResNet50 with their low features to show this design.

3.2. DATASET AND AUGMENTATION

With the use of the AQI index, the daily air quality is provided. It lets you know if the air is clean or dirty, and it alerts you to any potential health risks. The AQI focuses on potential health effects that may occur hours or days after inhaling polluted air. The four primary air pollutants specified by the Clean Air Act are used to calculate the AQI: carbon monoxide, sulfur dioxide, ground-level ozone, and particle pollution. The amount of air pollution and health concern is inversely correlated with the AQI value. This paper considered the air quality using photos taken at fixed locations in Shanghai [10] with its corresponding PM2.5 [11]. The images have been divided into five classes based on the number of micrograms (the mass or weight) per cubic meter of air as (µg/m3), In this work, the level of PM2.5 images start with Level 1 class and a concentration between (0 to 18.5) and end with the Level 5 class with concentrations greater than 59.9 as shown in Table (1)

Table 1. PM2.5 classes and concentration.

Class Level	PM2.5 concentration (µg)
Level 1	< = 18.4
Level 2	18.5 - 30.4
Level 3	30.5 - 40.4
Level 4	40.5 - 59.9
Level 5	> = 60

The Shanghai PM2.5 dataset is available online through Figshare and contains about 1954 images with different air quality levels during daylight. The images are 389 by 584-pixel resolution. To use the images with ResNet, and as its first layer of size (224, 224, 3), we resized the image to fit the input layer. besides, for image augmentation, we flipped all the images horizontally to get a total image equal to 2104 images. The images number in each class are shown in Table (2), (some images are not considered to ensure data balancing between the classes). According to [12], using a pre-trained model, 150-500 images per class is sufficient to achieve reasonable classification accuracy (Reasonable, not the best).

Table 2. The number of images in each class.

Class	No. of images
Level 1	320
Level 2	411
Level 3	492
Level 4	472
Level 5	409
All classes' images	**2104**

4. METHODOLOGY

In this part, we presented a novel categorization scheme for air-quality photographs. The design is built on transfer learning by utilizing ResNet18 (Depth=18, layers =71, Size=44MB, Parameters =11.7 million, input size, 224, 224, 3) and ResNet50 (Depth=50, layers =177, Size=96MB, Parameters =25.6 million, input size, 224, 224, 3) and. The new proposed model consists of 260 layers and 35.4 million parameters, and its size is 249 MB. ResNet18 and ResNet50, two pre-trained convolutional neural networks, are upgraded and utilized without the final layers (fully connected, Softmax, and classification layers). Their role is to perform high-feature

https://doi.org/10.17993/3ctic.2023.121.378-398

extraction (dense presentations of the input images). This means, to this point, we will have two lines of stacked layers both starting with the input layers and ending with a ReLU layer, with some in between layers according to each model architecture. The ResNet18 and ResNet50 are upgraded then, by adding two important layers, (Flatten and LSTM layers). In addition to the two models, we stacked another two lines of layers called (Low_R18 and Low_R50) to extract the low image features from ResNet18 and ResNet50 respectively.

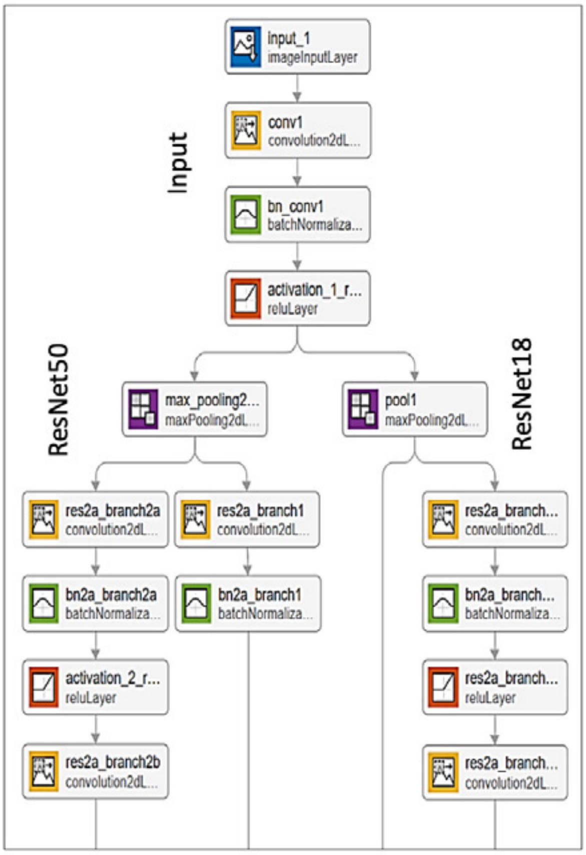
Low_R18, started by the ResNet18 layer number (18), while Low_R50, started by the ResNet50 layer number (36), with no layers in between for both. The Low_R18 and Low_R50 columns are designed by feeding them (from their starting layers) directly into a dropout layer with a probability equal to 0.5 to reduce the overfitting (A dropout layer randomly sets some input elements to zero with a given probability). Then to a “2-D max pooling” layers, with (pool size = 4,4, strid = 4,4. Each output of the “2-D max pooling” layer is fed to a 2-D average pooling layer with (pool size = 2,2, strid = 2,2). After that, we entered the output of both columns (Low_R18, and Low_R50) to a flattened layer, followed by Long short-term memory (LSTM) layer with several hidden units equal to (10). The two outputs of columns (ResNet18 and Low_R18) are added using an additional layer. Also, the two outputs of columns (ResNet50 and Low_R50) are added using another additional layer.

The outputs from the two additional layers are concatenated using the concatenation layer and passed again to another dropout layer with a probability of 0.5, and finally passed to the last dense layer followed by a Layer normalization layer.

The strategy's conceptual underpinnings include:

- Initial pre-processing for image resizing.
- Image augmentation with horizontal flip to the right.
- ResNet18 and ReNet50 upgrading.
- Low and high feature extraction based on transfer learning, by the newly added columns of layers (Low_18 and Low_50).

Figures (1 and 2), which depict the upper and bottom portions of the newly proposed model, respectively, exhibit the block diagram of the suggested technique. The suggested models and the added layers will be covered in detail in the sections below. The proposed model will add the high features of the modified ResNet18 with the low features of the ResNet18 after stacking some useful layers (Low_R18). And the same for ResNet50, it will add the high features of the modified ResNet50 with the low features of the ResNet50 after stacking some useful layers (Low_R50).



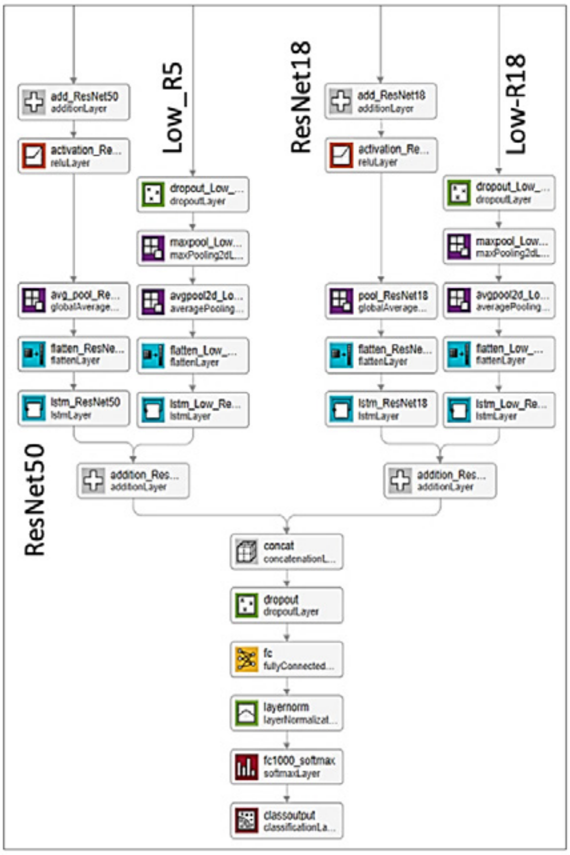


Figure 1. The upper layers of the new model **Figure 2.** The lower layers of the new model

4.1. TRANSFER LEARNING FOR TRAINING DEEP LEARNING MODELS

A model that has been trained for one job is utilized as a starting point for a model that completes a related task in the transfer learning method of deep learning [13]. Transfer learning may often update and retrain a network more quickly and easily than starting from scratch. Transfer learning is a well-liked approach because: It allows you to reuse well-liked models that have already been trained on huge datasets, allowing you to train models with less labeled data. Training and testing data for traditional machine learning often share the same input feature space and data distribution. The performance of a predictive learner may suffer when the distribution of data between training and test sets differs [14]. In some circumstances, it might be challenging and expensive to find training data that fits the test data's feature space and projected data distribution properties. A high-performance learner for a target domain is thus required, and it must be taught from a comparable source domain. The drive behind transfer learning is this. On the other hand, combined deep convolutional neural networks (C-DCNN) eliminate the requirement to extract complex features before training the classifier and allow for the learning of high-level and advanced features from the training dataset [9]. Convolutional layers for feature extraction give the classification system effective field knowledge. CNNs enable the reduction of the field knowledge required to create a classification system, followed by layer pooling. As a result, the approach's performance is less influenced by the dataset that was utilized,

https://doi.org/10.17993/3ctic.2023.121.378-398

and comparable network topologies can produce positive outcomes for a variety of issues. When used to estimate air quality, the C-DCNN has improved pollutant concentration predictions [15]. The performance of the created learning model heavily depends on the accessibility of the learning dataset. Due to their small datasets, certain classification problems cannot conduct the necessary deep learning. Additionally, data gathering is costly and may need a complex capture procedure and professional annotation. Transfer learning seeks to solve the dataset availability issue by using a pre-trained model on a sizable, labeled dataset of a generic context. To meet the new setting, only a minor training change is necessary [9]. The massive Imagenet dataset, which includes more than a million photos belonging to a thousand different object categories, is used to train ResNet50 and ResNet18. The accuracy of the two models on the Imagenet dataset is good. The following is a description of the two CNNs' structures:

4.2. RESNET18 ARCHITECTURE

ResNet18 has 18 layers with a 7x7 kernel as 1st layer. It has four layers of ConvNets that are identical. Each layer is made up of two residual blocks. Each block is made up of two weight layers with a skip connection connected to the output of the second weight layer with a ReLU. If the result is equal to the input of the ConvNet layer, then the identity connection is used. But, if the input is not similar to the output, then a convolutional pooling is done on the skip connection. ResNet18 also used two pooling layers throughout the network one at the network's inception and the other at its conclusion. The input size taken by it is (224, 224, 3), where 224 is the width and height, and 3 represents the RBG channel. The output is a fully connected layer that gives input to the sequential layer [16, 17].

4.3. RESNET50 ARCHITECTURE

ResNet-50 is a pre-trained model that won the 2015 ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) competition. It was trained on a portion of the ImageNet database. The model can classify photos into 1000 item categories and is trained on more than a million photographs. It contains 177 layers in total, which corresponds to a 50-layer residual network (224, 224, 3) [18]. The ResNet architecture (figure 3) is considered to be among the most popular Convolutional Neural Network architectures around. Residual Networks (abbreviated ResNet) were first described by Xiangyu Zhang, Kaiming He, Jian Sun, and Shaoqing Ren in their 2015 computer vision research paper titled "Deep Residual Learning for Image Recognition" [18]. ResNet was later introduced by Microsoft Research in 2015 and set numerous records.

https://doi.org/10.17993/3ctic.2023.121.378-398

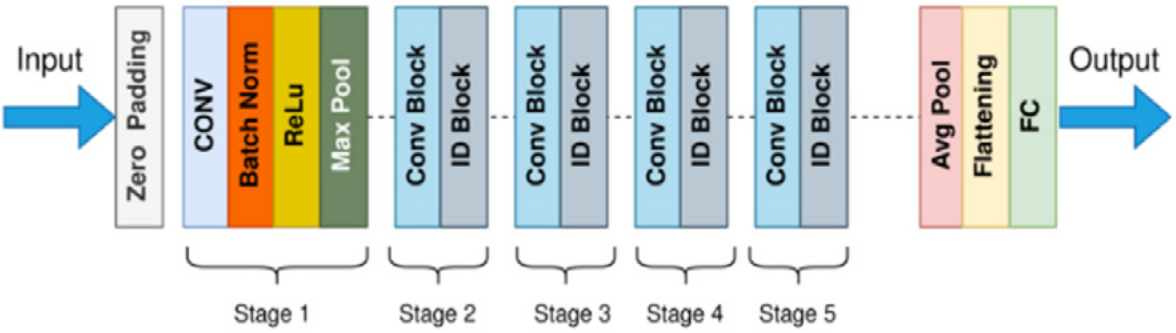


Figure3. ResNet architecture.

A significant drawback of convolutional neural networks is the "Vanishing Gradient Problem." Weights scarcely change as a result of the considerable fall in radiant value that occurs during backpropagation. ResNet is employed to get around this. It employs "SKIP CONNECTION," which, as illustrated in Figure (4) [18], adds the original input to the convolutional block's output. The authors of Xiao Tian [19] provide many common convolutional neural networks, including VGG16, VGG19, Inception, Xception, and ResNet50, which are utilized to determine the constellation's modulation pattern. Through trials, it has been shown that among other models, the ResNet50 network works best and has the greatest accuracy. Numerous more studies, including [20] and [21], demonstrate ResNet50's superior accuracy performance.

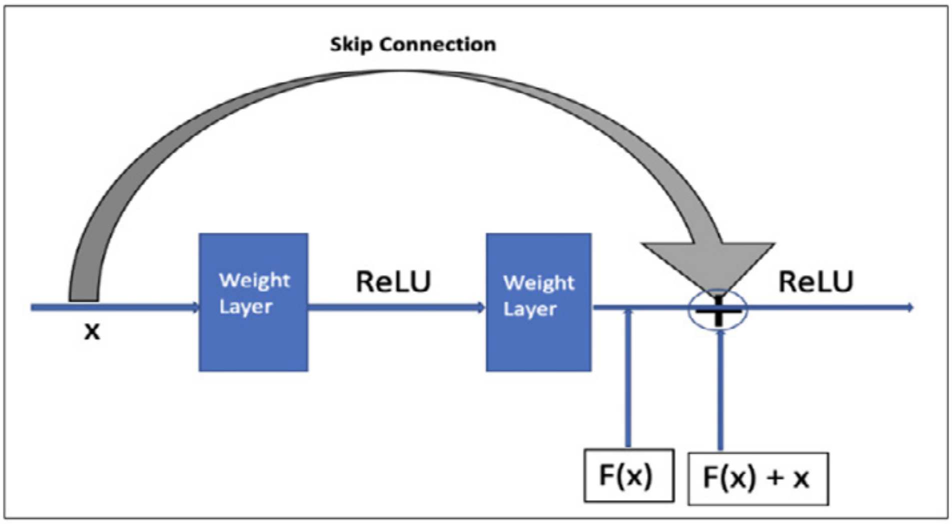


Figure 4. ResNet skip connection.

4.4. COMBINING MODELS AND ADDED LAYERS

Next in sections (4.4.1 to 4.4.5), we will describe the added layers to ResNet18 and ResNet50, and the layers added to extract the ResNet18 low features and ResNet50 low features (Low_R18 and Low_R50 respectively), to build the whole proposed model.

https://doi.org/10.17993/3ctic.2023.121.378-398

4.4.1. DROPOUT LAYER

Problems with training time and learning overfitting are frequent in deep learning, particularly in C-DCNN. Additionally, it costs a lot of computation to combine the outputs of several trained models [22]. To address these issues, the feature vector should be dropped out before being fed into the max pooling layers. Dropout [23] is a newly suggested regularizer to combat the overfitting problem. It is a regularization technique that stochastically adjusts the hidden unit activations for each training example to zero during training. Other stochastic model averaging techniques like stochastic pooling [24], drop-connect [25], and maxout networks [26] were influenced by dropout. The following figure (5) illustrates how the dropout layer affected the Low R50 and Low R18 input features.

Adding a dropout layer before the first max pooling layer row [27] demonstrate that sampling activation based on a multinomial distribution with an adjustable parameter p is equal to utilizing max-pooling dropout at training time (the retaining probability). This method, which is only carried out during training time, has a major impact on cutting down on training time and preventing overfitting. The following layer for columns (Low R50 and Low R18) is a max pooling layer based on that.

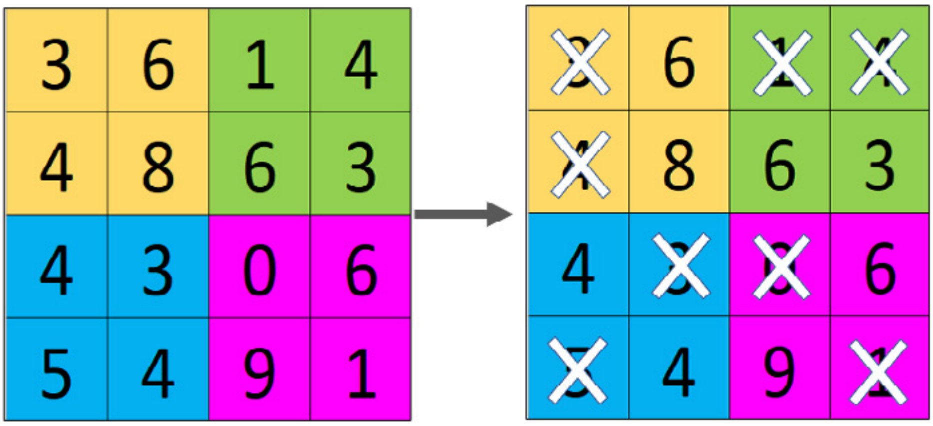


Figure 5. Dropout layer with probability = 0.5

4.4.2. 2D MAX POOLING AND AVERAGE LAYERS

The output feature vectors of each dropout are fed into a “2-D max pooling” row layer and symbolized as “maxPooling2dLayer”, with (pool size = 4,4, strid = 4,4). This layer is used to perform downsampling separately on columns (Low_R50 and Low_R18), by breaking the layer’s input into square or rectangle areas according to the pool size, then finding each area's maximum value. The used pool size refers to the width and height of the rectangular regions and the stride represents the stride dimensions which is the steps that the rectangular region will move as shown in Figure (6).

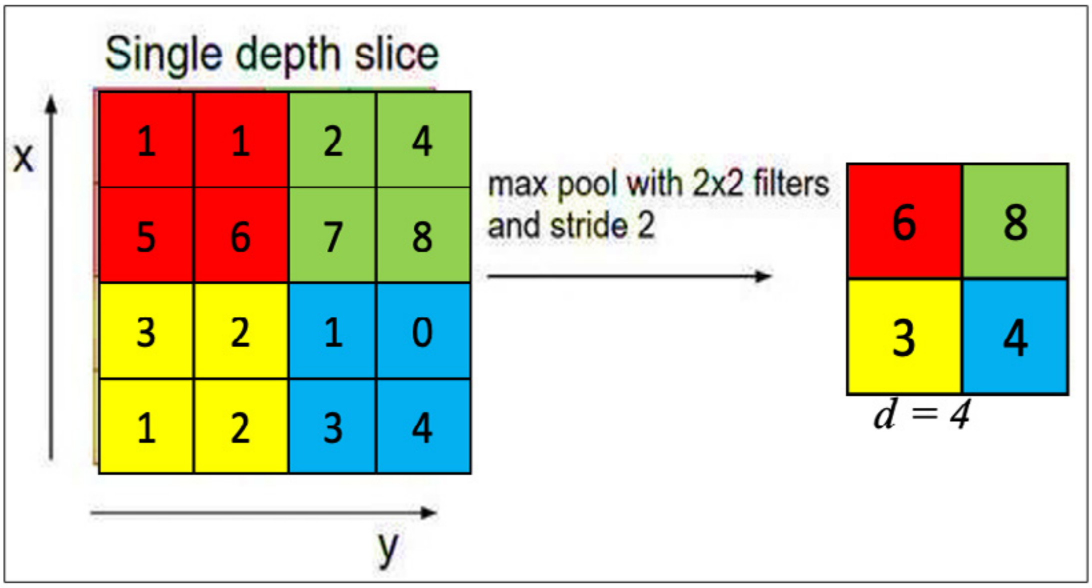


Figure 6. Max pooling layer.

The pooling zones overlap if the stride dimensions are smaller than the corresponding pooling dimensions [28]. To overcome this problem, we have to choose a size equal to or greater than the pooling size (in our model we used a stride size equal to the pooling size), to prevent the overfitting issue by reducing the number of parameters. Each (x × y) area represents a feature map, for example, the green area to the left of the figure is changed to a single number (the green area with the number '4' in the right of the figure tallied as the max of [x y] values. The resultant feature is an array with d values, where d is the number of filters. Introduced in[28], maxPooling2dLayer provides a more accurate classification. Additionally, the Max pooling layer offers the key functionality (i.e., edges), Additionally, it is more adept at handling the extraction of the extreme characteristics. This implies that feature mapping truly uses all values [29].

The same is for the 2D average pooling layer shown in Figure (7) which downsamples the pooled area by splitting the features into the square area and calculating the average of that area.

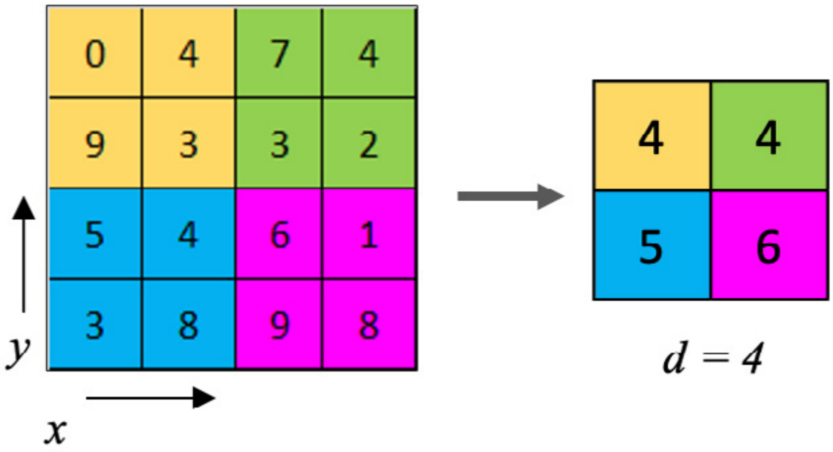


Figure 7. 2D average pooling layer.

https://doi.org/10.17993/3ctic.2023.121.378-398

4.4.3. FLATTEN LAYER

The square or rectangle input features are converted into a single dimension by a flattened layer since rectangular or cubic forms cannot be used as direct inputs (convert images to feature vectors). Figure (8) shows a flattened layer applied on a pooled feature map.

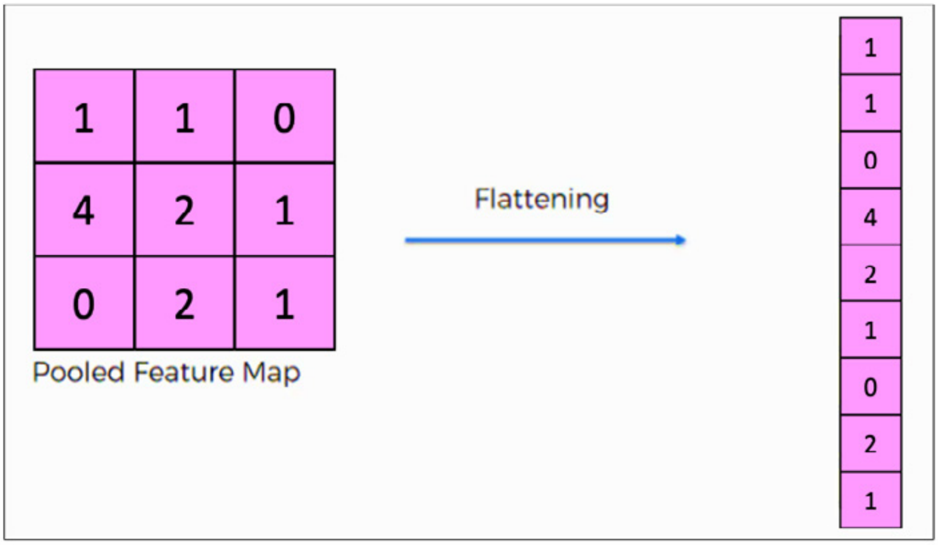


Figure 8. Applying a flattened layer.

The reason for flattening the intermediate outputs (feature maps) is: After the pooling layer, we will get a feature map with different heights and widths for each column as shown in the activation of Table (3), which is already obtained from images after passing through a lot of layers. Flatten layers allow us to add features of different lengths.

Table 3. Activations properties of each column.

Column	Activations	After flattening
ResNet50	1(S) x 1(S) x 2048(C)	4048(C)
ResNet18	1(S) x 1(S) x 512(C)	512(C)
Low_R18	7(S) x 7(S) x 256(C)	12544(C)
Low_R50	7(S) x 7(S) x 64(C)	3136(C)

For our problem we have only five classes, so, we have to flatten the output of each pooling layer and pass through a neural network that has the number of output layers corresponding to the number of classes (5). In the imaging context, these are referred to as linear layers. In other words, the convolution layer acts as a feature extractor which helps a fully connected layer to do the task of classifying the images. A 1D array is used by these fully connected layers to carry out the categorization. To facilitate smooth 1D array operations, we flatten the data as we analyzed the images.

4.4.4. LSTM LAYERS (LONG SHORT-TERM MEMORY)

The long-term relationships between time steps in time series and sequence data are learned by an LSTM layer. The additive interactions carried out by this layer during training may aid in enhancing gradient flow over lengthy periods. Recurrent neural networks of the (LSTM) kind can learn order dependency in sequence prediction tasks, and complicated problem domains demand this behavior. It contains feedback connections, which means that aside from single data points like photos, it can interpret the complete sequence of data. The theory of LSTM is shown in the following figure (9).

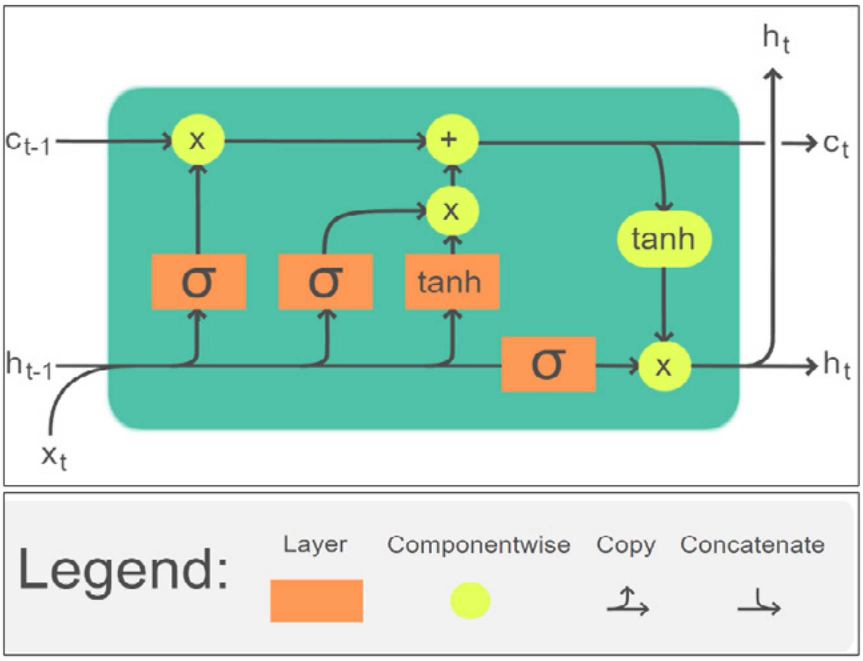


Figure 9. LSTM layer architecture

An important role in an LSTM model is played by a memory cell known as a "cell state" that maintains its state across time. The horizontal line that passes across the top of the aforementioned figure indicates the cell state. It may be seen as a conveyor belt across which data simply and unaltered passes. Information removal from or addition to the cell state is controlled by the LSTM layer gates. These gates could let data through and out of the cell. It has a sigmoid neural network layer and a pointwise multiplication operation that support the technique shown in Figure (10). The sigmoid layer generates numbers ranging from 0 to 1, where 0 denotes that nothing should be let through and 1 denotes that everything should be allowed through. Simple addition or multiplication operations that pass-through cell states are used by LSTM to make little changes to the data. The LSTM selectively forgets and recalls information in this way. The quantity of data remembered in between time steps (the hidden state) is configured to match the number of hidden units (10). Regardless of the duration of the series, the concealed state can contain data from every previous time step.

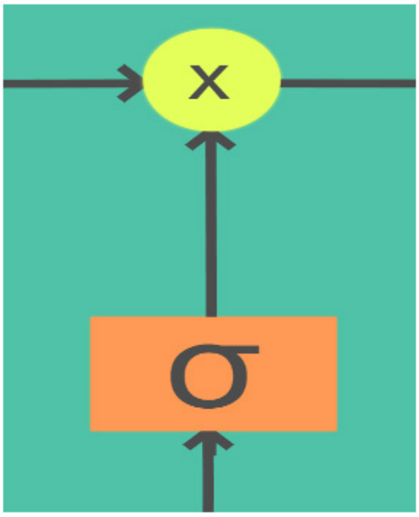


Figure 10. Multiplication operation and a sigmoid neural netlayer

4.4.5. DENSELY CONNECTED LAYERS

Finally, the output features of the concatenation layer are fed again to a dropout layer, then to a densely connected layer (fully connected layer) followed by a “layer normalization layer and”, then to a Softmax logistic regression layer. The reason behind using the layer normalization layer is to speed up the training of recurrent and multilayer perceptron neural networks and reduce the sensitivity to network initialization. After normalizing progress, the input to the layer is scaled with a factor γ called “learnable scale” and then it is shifted by the learnable offset β (which are left as their default values).

5. EXPERIMENTATION SETTINGS

The five classes are presented with several images as in Table (2). The images are in (.jpg) format and have a size of 389 by 584 pixels. Figure 11 highlights the variability of the five classes. Before the learning and classification steps, the input images are used directly after resizing them to (224, 224, 3), with no other previous filtering or pre-processing. The combination of the two models with the low features allows for achieving an improvement in air quality prediction by about 6.31% (with a learning rate LR=0.0007). The dataset was expanded through image data augmentation and obtained using Python programming language and achieved by flipping the images horizontally to one side only (right). We specified No special parameters in Matlab for image data augmentation. For columns (Low_R18 and Low_R50), a dropout of 0.5 is used to prevent the overfitting of the training process. Followed by two rows of 2D_max pooling and 2D average pooling layers. From this point, all the columns, then are passed to flatten layers to convert the features map into vector form followed by LSTM layers. The experiments were set with the following parameters (option of training): Solver for training network = sgdm (use the stochastic gradient descent with momentum optimizer), This work, tried different learning rates (0.007, 0.0007, and 0.00007), with tested batch sizes equal to 10 and 60 epochs. The number of iterations per epoch was 149 of a total of 8940 iterations. The loss function

https://doi.org/10.17993/3ctic.2023.121.378-398

computes the cross-entropy loss between the labels and predictions for the training set and the validation set.

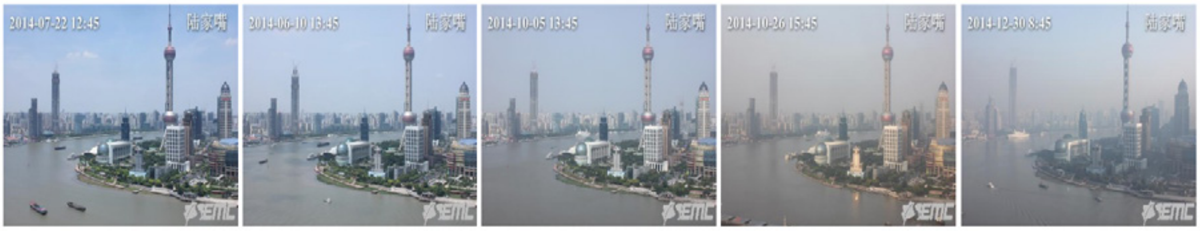


Figure 11. A variability of the five classes starting with Level 1, Level 2, Level 3, Level 4 and Level 5.

The proposed model is trained first on 80% of the dataset while 20% of the training data are used for the validation step and 20% for testing. That led to 65% of the data for training, 15% for validation, and 20% for testing. The statistics about the training, test, and validation for the sub-images datasets are detailed in Table (4).

Table 4. Datasets statistics for training, validation, and testing.

Set	Percentage	No. of images
Training	65 %	1368
Validation	15 %	316
Test	20 %	420

The environment for the experiment used along with MATLAB R2022a is as follows:

- System Manufacturer: LENOVO LEGION5
- OS Name: Microsoft Windows 11 Home
- System Type: x64-based PC
- Processor: Intel(R) Core (TM) i7-10750H CPU @ 2.60GHz, 2592 Mhz, 6 Core(s), 12 Logical Processor(s)
- Installed Physical Memory (RAM): 16.0 GB

5.1. IMAGES CLASSIFICATION RESULTS

As the number of images is limited and too low, two pre-trained models are used. dataset augmentation, learning transfer, allows for to prevent or reduce the overfitting issue by dataset increasing and enhance the training process. The classes are created by splitting the utilized image dataset (each of size 389 by 584) into five classes each of size (224 by 224). The estimated level of the five sub-images classes is obtained using the presented deep model. In this paper, first, a zero-weight ResNet50 is used, to test a model that hasn't been trained before to show if using a small image dataset could achieve reasonable accuracy or not. It is found that the zero-weight ResNet50 could not exceed 28% of accuracy, whatever the learning rate is. Then the pre-trained ResNet50 along with the three learning rates (LR= 0.007, 0.0007, and 0.00007) is applied, without any changes on its layers. It is found that the system achieved an average accuracy (tested three times with the same model options) equal to (68.61%, 70.86%, and 67.44% respectively with the LR). Also, the average accuracy using the test part of the image gave a close accuracy to the validation accuracy which is a clue that the system has not overfitted, as shown in Table (5).

Table 5. ResNet50 accuracies to LR.

ResNet50			
Learning rate	**Avg. Validation acc.**	**Avg. test acc.**	**Avg. total time**
0.007	68.49	67.64	51.34
0.0007	70.92	70.88	54.59
0.00007	66.91	67.52	58.54

net50 achieved the best accuracy (70.86%) with LR = 0.0007 with an average time equal to 52 minutes and 85 seconds. Then the new model with the same previously mentioned training options is applied, three times on each of the three learning rates to calculate the average validation accuracy. The experimental results show that the new model can achieve an average validation accuracy equal to (72.92%, 77.17%, and 73.54% respectively with the LR= 0.007, 0.0007, and 0.00007) as shown in Table (6).

Table 6. The new model accuracies for LR.

The New Proposed Model			
Learning rate	**Avg. Validation acc.**	**Avg. test acc.**	**Avg. total time**
0.007	72.92	71.88	82.84
0.0007	77.17	75.31	81.21
0.00007	73.54	73.27	88.41

The new model achieved the highest accuracy with LR=0.0007, it is also clear that the model has a test accuracy very close to the validation accuracies. Compared to a fresh model like the zero-weight ResNet50, the system achieved a very big jump in accuracy close to 50%. Compared to the pre-trained ResNet50, the new model achieved a reasonable increment in accuracy with about (6.31%) for the learning rate (0.0007), while it achieved about (4.31% and 6.1%) for learning rates (0.007 and 0.00007). From tables (5 and 6). it is noticed that the new model takes longer time for learning, which is because the new model has a higher number of layers and extracts the features four times.

https://doi.org/10.17993/3ctic.2023.121.378-398

6. CONCLUSION AND FUTURE WORK

In this work, a deep architecture for the classification of air quality based on PM2.5 concentration levels images is proposed. The new proposed model allows for the discrimination of the low and high features. Compared to existing research, the performance of our approach outperforms state-of-the-art methods. From the results, because the new models have a higher number of layers and extract the features four times, the new models take a longer time per epoch for learning. The new proposed model achieved an accuracy increment of about (6.31% at LR=0.0007), while it achieved (4.31% and 6.1% with LR=0.007 and 0.00007 respectively) as clarified in Tables (5 and 6). In the future, our approach can be extended by adding other models to the proposed one in this paper, or by utilizing other pre-trained models. Also, a multi-scene image dataset can be used next.

REFERENCES

(1) Silva, L. F. O., Oliveira, M. L. S., Neckel, A., Maculan, L. S., Milanes, C. B., Bodah, B. W., & Dotto, G. L. (2022). **Effects of atmospheric pollutants on human health and deterioration of medieval historical architecture (North Africa, Tunisia)**. *Elsevier BV*. https://doi.org/10.1016/j.uclim.2021.101046

(2) Brook, R. D., Brook, J. R., & Rajagopalan, S. (2003). **Air pollution: The "heart" of the problem**. *Springer Science and Business Media LLC*. https://doi.org/10.1007/s11906-003-0008-y

(3) Landrigan, P. J. (2017). **Air pollution and health**. *Elsevier BV*. https://doi.org/10.1016/s2468-2667(16)30023-8

(4) Zheng, S., Wu, X., Lichtfouse, E., & Wang, J. (2022). **High-resolution mapping of premature mortality induced by atmospheric particulate matter in China**. *Springer Science and Business Media LLC*. https://doi.org/10.1007/s10311-022-01445-6

(5) Liu, C., Tsow, F., Zou, Y., & Tao, N. (2016). **Particle Pollution Estimation Based on Image Analysis (H. Liu, Ed.)**. *Public Library of Science (PLoS)*. https://doi.org/10.1371/journal.pone.0145955

(6) Won, T., Eo, Y. D., Sung, H., Chong, K. S., Youn, J., & Lee, G. W. (2021). **Particulate Matter Estimation from Public Weather Data and Closed-Circuit Television Images**. *Springer Science and Business Media LLC*. https://doi.org/10.1007/s12205-021-0865-4

(7) Rijal, N., Gutta, R. T., Cao, T., Lin, J., Bo, Q., & Zhang, J. (2018). **Ensemble of Deep Neural Networks for Estimating Particulate Matter from Images**. *Presented at the 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC)*. https://doi.org/10.1109/icivc.2018.8492790

(8) Kow, P.-Y., Hsia, I.-W., Chang, L.-C., & Chang, F.-J. (2022). **Real-time image-based air quality estimation by deep learning neural networks**. *Elsevier BV*. https://doi.org/10.1016/j.jenvman.2022.114560

(9) Elmannai, H., Hamdi, M., & AlGarni, A. (2021). **Deep Learning Models Combining for Breast Cancer Histopathology Image Classification**. *Springer Science and Business Media LLC*. https://doi.org/10.2991/ijcis.d.210301.002

(10) Liu, Chenbin; Tsow, Francis; Zou, Yi; Tao, Nongjian (2016): **Particle pollution estimation based on image analysis**. *figshare. Figure*. https://doi.org/10.6084/m9.figshare.1603556.v2

(11) Airnow.gov. (n.d.). **U.S. Embassies and Consulates - China - Shanghai. Retrieved April 20, 2023, from** https://www.airnow.gov/international/us-embassies-and-consulates/#China$Shanghai

(12) Shahinfar, S., Meek, P., & Falzon, G. (2020). **“How many images do I need?” Understanding how sample size per class affects deep learning model performance metrics for balanced designs in autonomous wildlife monitoring**. *Elsevier BV*. https://doi.org/10.1016/j.ecoinf.2020.101085

(13) Weiss, K., Khoshgoftaar, T. M., & Wang, D. (2016). **A survey of transfer learning**. *Springer Science and Business Media LLC*. https://doi.org/10.1186/s40537-016-0043-6

(14) Shimodaira, H. (2000). **Improving predictive inference under covariate shift by weighting the log-likelihood function**. *Elsevier BV*. https://doi.org/10.1016/s0378-3758(00)00115-4

(15) Gilik, A., Ogrenci, A. S., & Ozmen, A. (2021). **Air quality prediction using CNN+LSTM-based hybrid deep learning architecture**. *Springer Science and Business Media LLC*. https://doi.org/10.1007/s11356-021-16227-w

(16) Ramirez, O. J. V., Cruz de la Cruz, J. E., & Machaca, W. A. M. (2021). **Agroindustrial Plant for the Classification of Hass Avocados in Real-Time with ResNet-18 Architecture**. *Presented at the 2021 5th International Conference on Robotics and Automation Sciences (ICRAS)*. https://doi.org/10.1109/icras52289.2021.9476659

(17) Li, Y., Huang, J., & Luo, J. (2015). **Using user generated online photos to estimate and monitor air pollution in major cities**. *Presented at the ICIMCS ’15: International Conference on Internet Multimedia Computing and Service*. https://doi.org/10.1145/2808492.2808564

(18) He, K., Zhang, X., Ren, S., & Sun, J. (2016). **Deep Residual Learning for Image Recognition**. *Presented at the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. https://doi.org/10.1109/cvpr.2016.90

(19) Tian, X., & Chen, C. (2019). **Modulation Pattern Recognition Based on Resnet50 Neural Network**. *Presented at the 2019 IEEE 2nd International Conference on Information Communication and Signal Processing (ICICSP)*. https://doi.org/10.1109/icicsp48821.2019.8958555

(20) Yang, X., Yang, D., & Huang, C. (2021). **An interactive prediction system of breast cancer based on ResNet50, chatbot and PyQt**. *Presented at the 2021 2nd International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT)*. https://doi.org/10.1109/ainit54228.2021.00068

(21) Kumar, D., Sharma, P., Anupama, A., & Sharma, P. (2022). **A Performance Study on Deep Learning Covid-19 Prediction through Chest X-Ray Image with ResNet50 Model**. *Presented at the 2022 Second International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*. https://doi.org/10.1109/icaect54875.2022.9807920

(22) Garbin, C., Zhu, X., & Marques, O. (2020). **Dropout vs. batch normalization: an empirical study of their impact to deep learning**. *Springer Science and Business Media LLC*. https://doi.org/10.1007/s11042-019-08453-9

(23) Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). **Improving neural networks by preventing co-adaptation of feature detectors (Version 1)**. *arXiv*. https://doi.org/10.48550/ARXIV.1207.0580

(24) Zeiler, M. D., & Fergus, R. (2013). **Stochastic Pooling for Regularization of Deep Convolutional Neural Networks (Version 1**. *arXiv*. https://doi.org/10.48550/ARXIV.1301.3557

(25) Santos, C. F. G. dos, Roder, M., Passos, L. A., & Papa, J. P. (2022). **MaxDropoutV2: An Improved Method to Drop Out Neurons in Convolutional Neural Networks**. *Springer International Publishing*. https://doi.org/10.1007/978-3-031-04881-4_22

(26) do Santos, C. F. G., Colombo, D., Roder, M., & Papa, J. P. (2021). **MaxDropout: Deep Neural Network Regularization Based on Maximum Output Values.** *Presented at the 2020 25th International Conference on Pattern Recognition (ICPR)*. https://doi.org/10.1109/icpr48806.2021.9412733

(27) Wu, H., & Gu, X. (2015). **Max-Pooling Dropout for Regularization of Convolutional Neural Networks.** *Springer International Publishing.* https://doi.org/10.1007/978-3-319-26532-2_6

(28) Fouad, M. M., Mostafa, E. M., & Elshafey, M. A. (2020). **Detection and localization enhancement for satellite images with small forgeries using modified GAN-based CNN structure.** *Universitas Ahmad Dahlan, Kampus* 3. https://doi.org/10.26555/ijain.v6i3.548

(29) Ma, Z., Chang, D., Xie, J., Ding, Y., Wen, S., Li, X., … Guo, J. (2019). **Fine-Grained Vehicle Classification With Channel Max Pooling Modified CNNs**. *Institute of Electrical and Electronics Engineers (IEEE)*. https://doi.org/10.1109/tvt.2019.2899972