Master's Thesis

석사 학위논문

Deep Convolutional Neural Networks for estimating PM2.5 concentration levels

Byung-Jun Kwon(권 병 준 權 柄 準)

Department of
Information and Communication Engineering

DGIST

2017

Deep Convolutional Neural Networks for estimating PM2.5 concentration levels

Advisor: Professor Suha kwak

Co-Advisor: Professor Taesup Moon

by

Byung-Jun Kwon

Department of information and Communication Engineering

DGIST

A thesis submitted to the faculty of DGIST in partial fulfillment of the requirements for the degree of Master of Science in the Department of Information and Communication Engineering. The study was conducted in accordance with Code of Research Ethics[1].

June. 28. 2017

Approved by

Professor Suha Kwak (Signature)
(Advisor)

Professor Taesup Moon (Signature)
(Co-Advisor)

[1] Declaration of Ethical Conduct in Research: I, as a graduate student of DGIST, hereby declare that I have not committed any acts that may damage the credibility of my research. These include, but are not limited to: falsification, thesis written by someone else, distortion of research findings or plagiarism. I affirm that my thesis contains honest conclusions based on my own careful research under the guidance of my thesis advisor.

Deep Convolutional Neural Networks for estimating PM2.5 concentration levels

Byung-Jun Kwon

Accepted in partial fulfillment of the requirements for the degree of Master of Science

June. 28. 2017

Head of committee _____(인)
Prof. Suha Kwak

Committee Member _____(인)
Prof. Taesup Moon

Committee Member _____(인)
Prof. Dooseok Lee

MS/IC
201522028

권 병 준. Byungjun Kwon. Deep Convolutional Neural Networks for estimating PM2.5 concentration levels. Department of Information and Communication Engineering. 2017 21p, Advisors Prof, Suha Kwak, Co-Advisors Prof. Taesup Moon

Abstract

Recent, $PM_{2.5}$ generated by various causes is spreading over a large area. As a result, the technique of estimating the concentration of $PM_{2.5}$ has become important. Many studies have been conducted to estimate the $PM_{2.5}$ concentrations. Recently research using learning-based models has been actively conducted. These learning-based models have exceeded the accuracy of existing statistical models. Existing research has a limitation that only data of the monitoring site can be used.

In this paper, we propose a method to improve the accuracy of $PM_{2.5}$ concentration estimation using the information of the surrounding area around the $PM_{2.5}$ monitoring site. To do this, we study the difference in performance between the 3-dimensional input data and the existing vector data. The algorithm used in this experiment is CNN which can efficiently use the correlation of spatial information. Experimental result show that using large spatial data gives high accuracy of $PM_{2.5}$ concentration estimation.

Keywords: Deep Learning, Neural Networks, CNN, $PM_{2.5}$

Contents

List of tables

List of figures

I. INTRODUCTION

Fine particulate matter(PM2.5) is a major concern for public health. It is created in the atmosphere by photochemical reaction with the toxic chemical. PM2.5 can be leading to various diseases such as myocardial infarction and lung inflammation. For this reason, estimating local PM2.5 concentrations is a very important issue.

Various approaches have been developed to achieve better estimating PM2.5 concentrations. Spatial interpolation, including nearest-neighbor interpolation and Kriging interpolation, was used to smooth PM2.5 concentrations. Recently, a lot of methods for using additional data have been studied. Satellite-based aerosol optical depth (AOD) measurements have been widely used to estimate PM2.5 in the various model for its large spatial coverage and repeated daily observations. In addition, land use data and meteorological data included.

Studies using statistical models as well as learning-based models are underway to use various data. For example, there is random forest [4] and neural networks [5]. This learning-based model achieved higher performance than existing statistical-based model. Both learning-based model and statistical-based model use only data point from PM2.5 monitoring site. However, at least PM2.5 is affected by surrounding area of PM2.5 monitoring site (for example, east 10km from PM2.5 monitoring site). In other words, we can better results by using spatial correlation to estimate PM2.5 concentrations. In this paper, we study performance enhancement using information surrounding area of PM2.5 monitoring site.

Unlike the traditional input data is vector form, proposed method's input data is 3-dimentional data including surrounding area information. We use convolutional neural networks algorithm to efficiently use correlation in spatial data. In the experiment, we use 1x1 dataset consist of only PM2.5 monitoring site. Also, we use 5x5, 7x7, 9x9 dataset included surrounding area.

In the experimental results, the larger range of data, the higher the performance. Dataset of existing high-performance model has higher resolution than dataset used in this study. Therefore, we expect to achieve higher performance using same high-resolution data and surrounding area.

II. BACKGROUND

2.1 Neural Networks

Mathematically, we can think of a linear layer as a function which applies a linear transformation on a vectorial input of dimension I and output a vector of dimension O. Usually the layer has a bias parameter.

$$y = Ax +$$

$$y_i = \sum_{j=1}^{I} (A_{i,j} x_j) + b_i$$

The linear layer is motivated by the basic computational unit of the brain called neuron. Approximately 86 billion neurons can be found in the human nervous system and they are connected with approximately 1014 - 1015 synapses. Each neuron receives input signals from its dendrites and produces output signal along its axon. The linear layer is a simplification of a group of neurons having their dendrites connected to the same inputs. Usually an activation function, such as sigmoid, is used to mimic the 1-0 impulse carried away from the cell body and also to add non-linearity. However, we consider here that the activation function is the identity function that output real values.

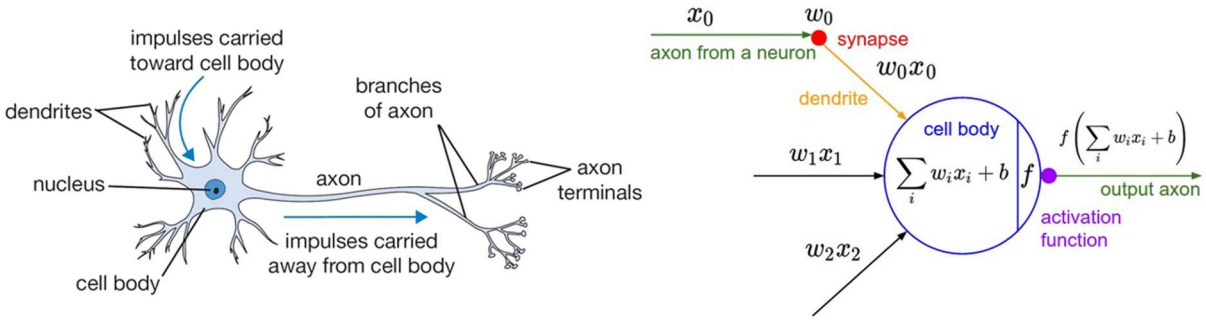

Figure 1. A biological neuron (left) and its mathematical model (right)

The capacity of the neural networks to approximate any functions, especially non-convex, is directly the result of the non-linear activation functions. Every kind of activation function takes a vector and performs a certain fixed point-wise operation on it. There are two main activation functions and one additional activation function.

The Sigmoid non-linearity has the following mathematical form

$$y = \sigma(x) = \frac{1}{(1 + e^{-x})}$$

It takes a real value and squashes it between 0 and 1. However, when the neuron's activation saturates at either tail of 0 or 1, the gradient at these regions is almost zero. Thus, the backpropagation algorithm fail at modifying its parameters and the parameters of the preceding neural layers.

The ReLU has the following mathematical form

$$y = max(0, x)$$

The ReLU has become very popular in the last few years, because it was found to greatly accelerate the convergence of stochastic gradient descent compared to the sigmoid/tanh functions due to its linear non-saturating form. In fact, it does not suffer from the vanishing or exploding gradient. Another advantage is that it involves cheap operations compared to the expensive exponentials. However, the ReLU removes all the negative information and thus appears not suited for all datasets and architectures.

The eLU has the following mathematical form

$$y = \begin{cases} x & if \;\; x > 0 \\ \alpha(\exp(x) - 1) & if \;\; x \leq 0 \end{cases}$$

The eLU solves the problem of removing negative information of ReLU. Positive numbers are the same as ReLU, but are used when the reference point is set to a non-zero negative number. Where α is a non-zero reference point.

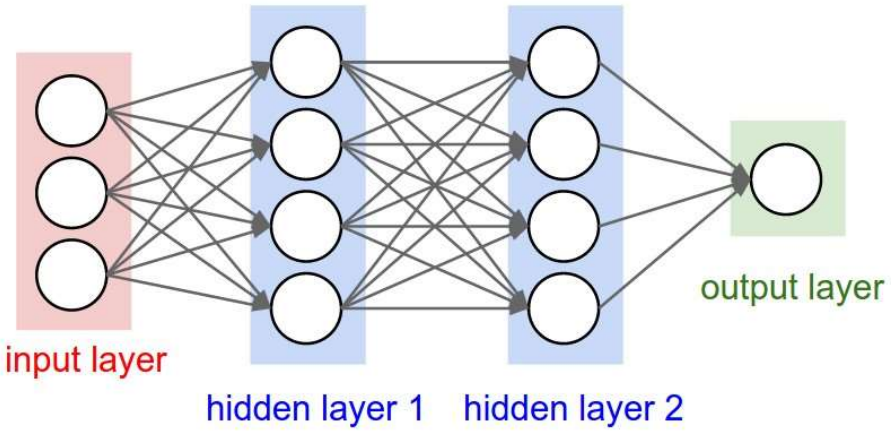

Figure 2. Neural networks architectures

2.2 Convolutional Neural Networks

A CNN consists of a number of convolutional and subsampling layers optionally followed by fully connected layers. The input to a convolutional layer is a m x m x r image where mm is the height and width of the image and r is the number of channels, e.g. an RGB image has r=3. The convolutional layer will have k filters (or kernels) of size n x n x q where n is smaller than the dimension of the image and q can either be the same as the number of channels r or smaller and may vary for each kernel. The size of the filters gives rise to the locally connected structure which are each convolved with the image to produce k feature maps of size $m - n + 1$. Each map is then subsampled typically with mean or max pooling over p x p contiguous regions where p ranges between 2 for small images (e.g. MNIST) and is usually not more than 5 for larger inputs. Either before or after the subsampling layer an additive bias and sigmoidal nonlinearity is applied to each feature map. The figure 4. illustrates a full layer in a CNN consisting of convolutional and subsampling sublayers. Units of the same color have tied weights. After the convolutional layers there may be any number of fully connected layers. The densely connected layers are identical to the layers in a standard multilayer neural network.

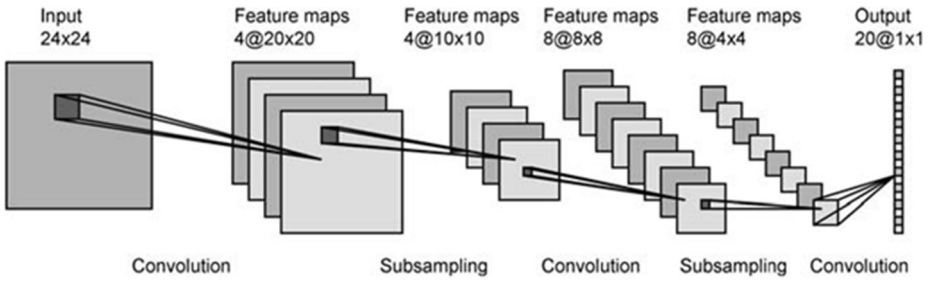

Figure 3. Convolutional neural networks architectures

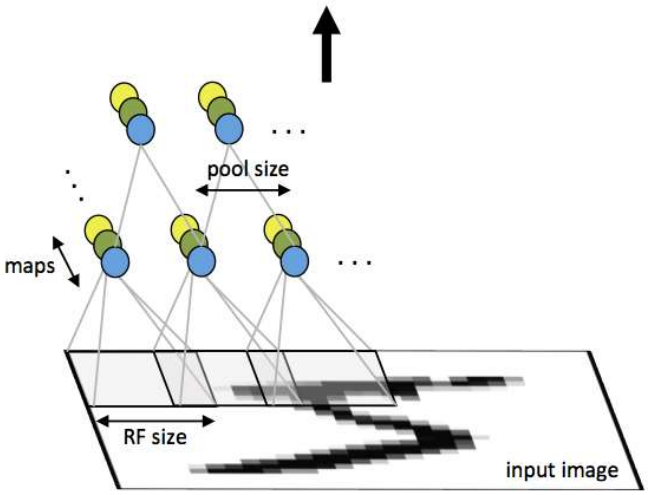

Figure 4. Process convolutional layer and pooling layer

This process is called feed forward. In the training phase, the output value through feed forward is compared with ground truth to determine the loss. For regression problems, loss function is MSE. For classification problem, we use softmax as loss function. We update parameters of the model using back propagation.

CNN has the advantage of using spatial information efficiently through the convolutional layer. In addition, the convolutional layer has fewer parameters, so memory cost can be reduced compared to performance improvement.

III. RELATED WORKS

The problem of estimating the PM2.5 concentrations has been the subject of much research since the past. It is being studied in various countries including the United States [4][5] and China [6], Netherland [1]. The most traditional method is to use a statistical model [6][7]. Recently, as the development of machine learning field, many research approach the estimation of PM2.5 concentrations using Random Forest [4] or neural network model [5]. This new approach brings performance improvements over existing methods. In [4], their results archive an overall cross-validation R2 value of 0.80. Dataset consists of MODIS AOD (10km x 10km resolution), Land use variable (~30m resolution), meteorological fields (~32km and ~13km resolution). Resolution of entire dataset is 1km x 1km and they are averaged over the overlapping grid. In [5], their total R2 value is 0.84 and is a very high score. Time domain is from 2000 to 2012 and highest R2 is 0.88 in 2004. Dataset consists of MAIAC AOD (1km x 1km resolution), chemical transport model, meteorological fields (~32km), Land use term(~30m), Etc.

IV. MATERIAL

In this study, we use various kind of data. First data is $PM_{2.5}$ measurements as the target. And meteorological fields data, Land use variables, regional and temporal variables. These data have an environmental impact on the $PM_{2.5}$ concentrations.

We define our study area as the conterminous United States consisting of 48 adjoining U.S states and Washington D.C. The time domain is one year of 2011.

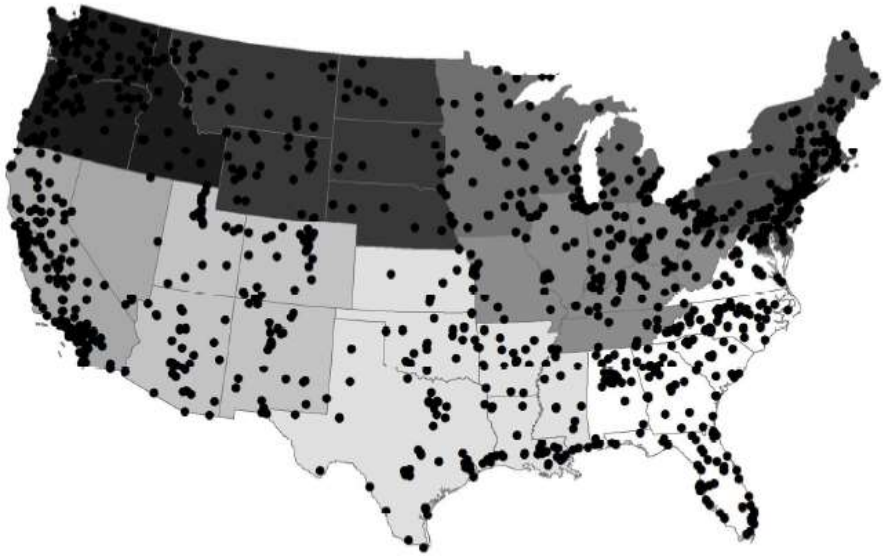

Figure 5. Study area and $PM_{2.5}$ monitoring station

4.1 $PM_{2.5}$ measurements

The 24-hour averaged $PM_{2.5}$ concentrations for 2011 collected from 1248 U.S Environmental Protection Agency(EPA) federal reference method samplers were downloaded from the EPA's Air Quality System Technology Transfer Network.

Not all $PM_{2.5}$ monitoring site collect normal data every day. Comparing Figure 5 and Figure 6, you can see several data points, not all. The location and number of $PM_{2.5}$ data points are changed daily.

4.2 AOD (Aerosol Optical Depth) data

AOD means the depth of the aerosol composed of particles in the atmosphere. AOD is one of the most significant features of PM2.5 concentrations. Since AOD is proportional to the concentration of aerosol, it generally tends to be proportional to the PM2.5.

We use satellite-derived AOD from MODIS as our primary and GEOS-Chem (Goddard Earth Observing System) AOD simulations. Because AOD uses optical spectrum, it is affected by cloud or snow and there are many missing data points. Figure 7 shows valid AOD data points (Magenta color) and missing AOD data points (Grey color).

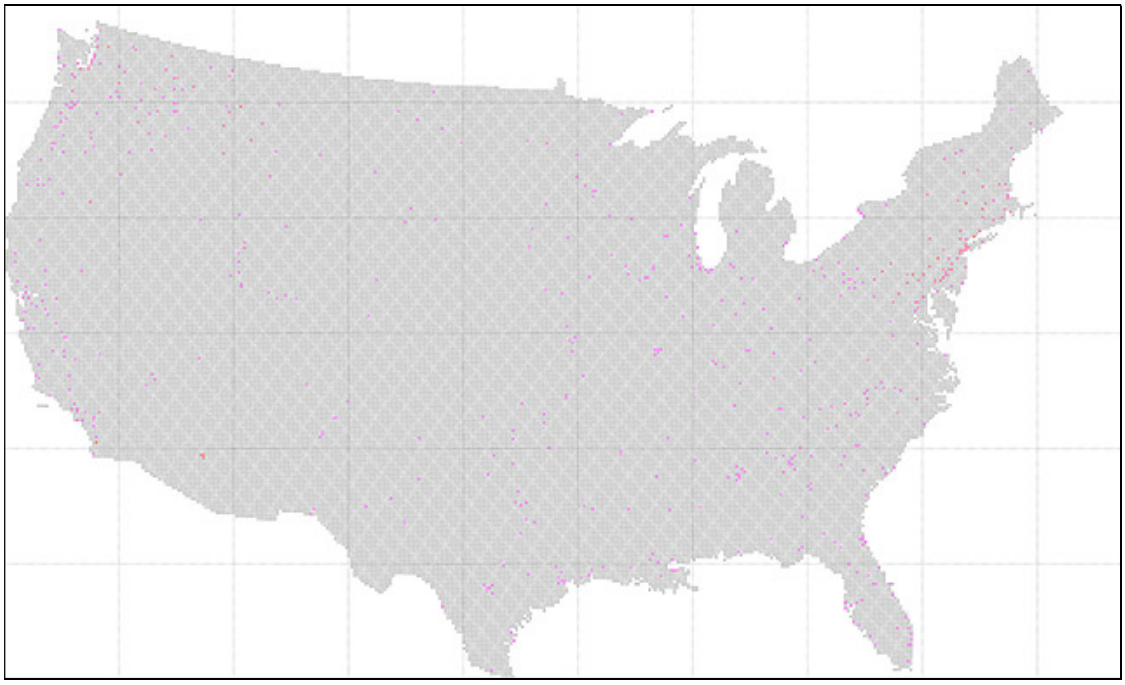

Figure 6. $PM_{2.5}$ data point on Jan 1 (Valid – Magenta / Missing – Grey)

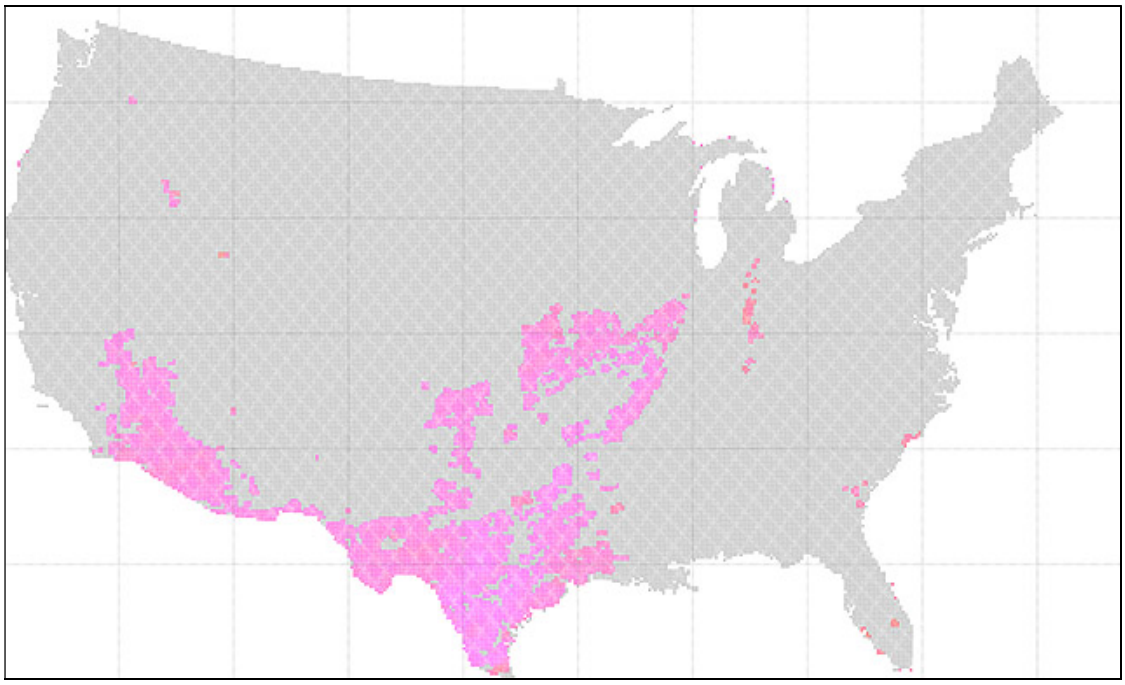

Figure 7. AOD data point on Jan 1 (Valid – Magenta / Missing – Grey)

4.3 Meteorological fields

We obtained meteorological fields from two separate datasets. The first is the North American Regional Reanalysis (NARR) with a spatial resolution of ~32km and a temporal resolution of three hours. The second is the North American Land Data Assimilation System Phase 2 (NLDAS -2) with a spatial resolution of ~13km and a temporal resolution of one hour. The meteorological fields used in this analysis include air temperature, dew point temperature, visibility, pressure, potential evaporation, downward longwave radiation flux, downward shortwave radiation flux, relative humidity, u-wind (the east-west component of the wind vector) and v-wind (the north-south component of the wind vector). All meteorological measurements for the period from 10 am to 4 pm local standard time were averaged to generate daytime meteorological fields.

4.4 Land use variables

We downloaded elevation data at a spatial resolution of ~30 m from the National Elevation Dataset (NED). We extracted road network data, including limited access highway, highway, and local roads, from ESRI StreetMap USA (Environmental System Research Institute, Inc., Redland, CA). We obtained forest cover and impervious surface, both at the spatial resolution of ~30m, from the 2011 Landsat-derived land cover map and impervious surface map downloaded from the National Land Cover Database (NLCD). We obtained primary $PM_{2.5}$ and PM10 emissions data (155 tons/year) from the 2011 EPA National Emissions Inventory (NEI) facility emissions report. Finally, we obtained 2010 population density data at the census tract level from the U.S. Census Bureau.

4.5 Regional and temporal dummy variables

The $PM_{2.5}$-AOD relationship has been shown to exhibit some regional variation due to meteorological conditions [1], and there are also daily variations in this relationship [2]. Hence, we include a dummy climate region variable to account for regional variations and monthly and daily dummy variables to account for daily variations.

4.6 Data integration

All data were re-projected to the USA Contiguous Albers Equal Area Conic coordinate system. We averaged forest cover, impervious surface, and elevation values and summed road length and point emission values over a 10km x 10km. We assigned each monitoring site a population density value of the census tract containing its location.

Target data is daily 24-hour averaged ground-level PM2.5 measurements. Dataset consists of 75 features. The feature includes (1~3) the x-y coordinate and PM2.5 monitoring site ID; (3~4) temporal dummy variables; (5~19) land use variables; (20~22) emission variables; (23~33) NLDAS variables; (34~72) NARR variables; (73) Geos-Chem AOD; (74) MODIS AOD; (75) PM2.5 convolutional feature (convolutional feature is explained in Section 6.1).

We leave data to blank when data is abnormal such as missing data point or overflow.

V. METHOD

5.1 Data generation

5.1.1 Convolutional feature

The basic dataset structure is described in Section 5.

To account for spatial and temporal autocorrelations, we adopt the distance-inversed weighted average function proposed by [3] to create the convolutional feature for nearby PM2.5 measurements and all land used terms and use these layers as ordinary input feature variables in our model. For each monitoring site and MODIS grid cell, distance-inversed weighted averages of nearby PM2.5 measurements and land use terms were calculated for that site and grid cell. The kernel function can be expressed in general terms as

$$z_j = \frac{\sum_{i=1}^{n} w_{ij} z_i}{\sum_{i=1}^{n} w_{ij}}$$

Where z_j is the value of convolutional feature at monitoring site or grid cell j, z_i is the $PM_{2.5}$ and land use terms at monitoring site i, and $w_{ij} \propto \frac{1}{d_{ij}^2}$ ($d_{ij}$is the distance between grid cell j and monitoring site i). When creating convolutional layers for monitoring site j, the $PM_{2.5}$ and land use terms at monitoring site j were not included in the calculation. There are 10 convolutional features calculated by this formula.

5.1.2 Spatial dataset

We made several datasets containing surrounding data points to take advantage of the spatial correlation of PM2.5. In common sense, the PM2.5 concentrations is affected by the

surrounding area a little. We created datasets using the figure 8 method. In figure 8, it shows each data point as a grid cell. All data points have all feature except for MODIS AOD and target value PM2.5 and are grayed out. PM2.5 value. If the PM2.5 value is included, it is displayed as blue. MODIS AOD is displayed as yellow. If both are included, it is displayed as green. We use all possible variables by setting the center of each data to green point. If n is 1, we only use one data point, and it is defined as a 1x1 dataset. To compare the performance improvement due to increased spatial area, we created 4 datasets (1x1, 5x5, 7x7, 9x9). In other words, 1x1 dataset is 2-dimentional matrix of 73,000 (number of samples) * 75 (number of features). 5x5 dataset is 4-dimentional matrix of 73,000 (number of samples) * 75 (number of features) * 5 (width of spatial area) * 5 (height if spatial area).

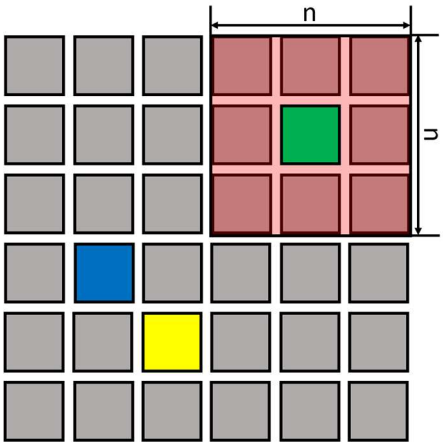

Figure 8. Grid cell (Blue-$PM_{2.5}$, Yellow-MODIS AOD, Green-$PM_{2.5}$ and MODIS AOD, Grey-Neither)

5.2 Validation and data preprocessing

We applied a 10-fold cross validation (CV) technique to establish and validate our prediction result. Before splitting into 10 subsets, entire training data is randomly shuffled. There are no overlapping samples in the divided 10 subsets. In each round of cross validation, we use one subset for model validation and other 9 subsets for model training. The process was repeated 10 times until subset was validated.

Before train phase in each round of cross validation, train set and validation set are normalized. At this time, we use normalization by the standard deviation. It calculates for each feature and the general term is as follows.

$$\hat{x}_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$$

Where x_{ij} is $i-th$ sample, $j-th$ feature, μ_j and σ_j are the mean and the standard deviation of the $j-th$ feature.

5.3 Model structure and experiments

We use two main model: Neural networks and convolutional neural networks. CNN is good for use in 5x5, 7x7, 9x9 datasets due to spatial correlation can be actively used. On the other hand, when the dataset is 1x1, we use simple neural networks because 1x1 dataset cannot use surrounding data point.

Neural networks for the 1x1 dataset is a simple structure. The model consisted of 3-layer and each layer has 1024 nodes and there is dropout (p = 0.5) between each layer. The number of layers and the number of nodes can vary depending on the input data, but in this experiment, this setting yielded the best result. The activation function is eLU (Exponential Linear Unit). We set the learning rate to decrease the number of steps. the general term is as follow and base learning rate is set to 0.002, the decay rate is set to 0.95.

$$\boldsymbol{lr} = \frac{\boldsymbol{lr}}{\boldsymbol{1 \ + \ decay \ rate \ * \ step}}$$

Convolutional neural networks can set various hyper-parameters than neural networks. A number of convolutional layers and a number of filters are depending on the size of data. The basic model of convolutional neural networks consists of two convolutional layers and two fully connected layers, 3x3 kernel / 256 filter - 3x3 kernel / 128 filter – 1024 fully connected – 1024 fully connected (Figure 10). In the convolutional layer, using padding, the output is same size as the input. After all the convolutional layers, batch-normalization layers are located. Learning rate policy is same as neural networks. The base learning rate is 0.001. Activation functions of the convolutional layer is ReLU, and eLU is used in fully connected.

All modeling was done using tensorflow version 1.0.1

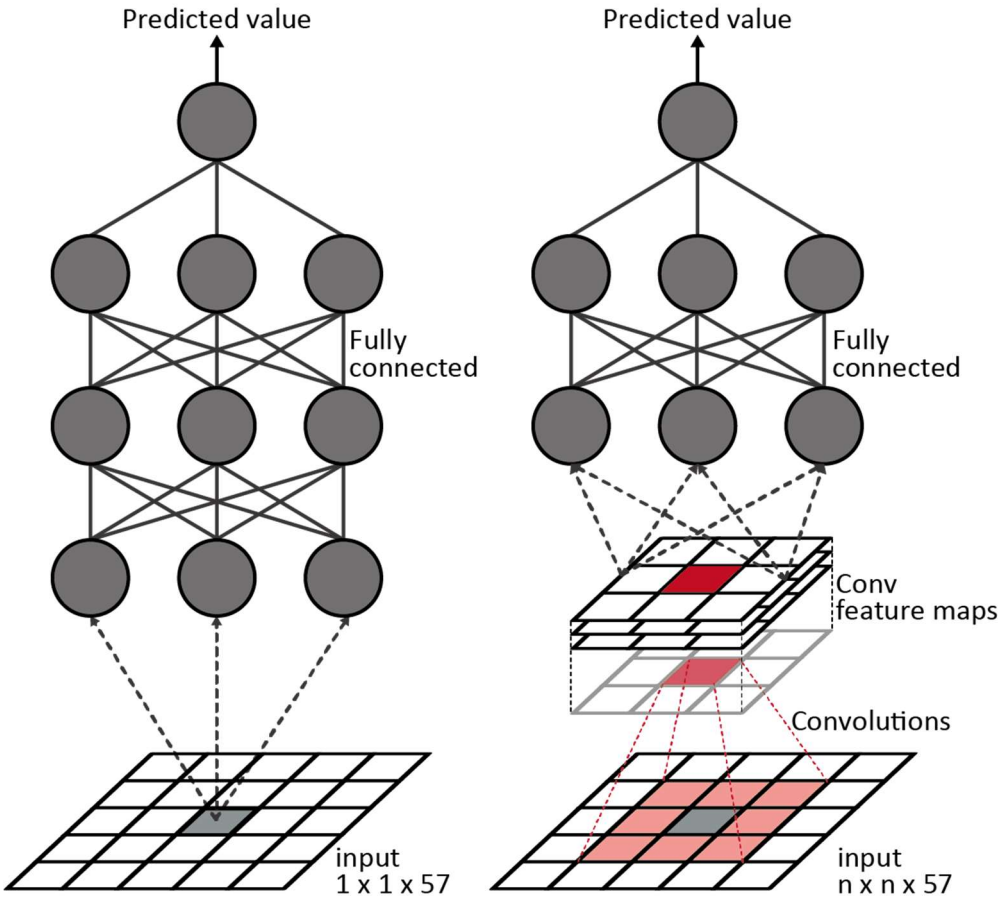

Figure 9. (left) Traditional method; (right) Proposed method

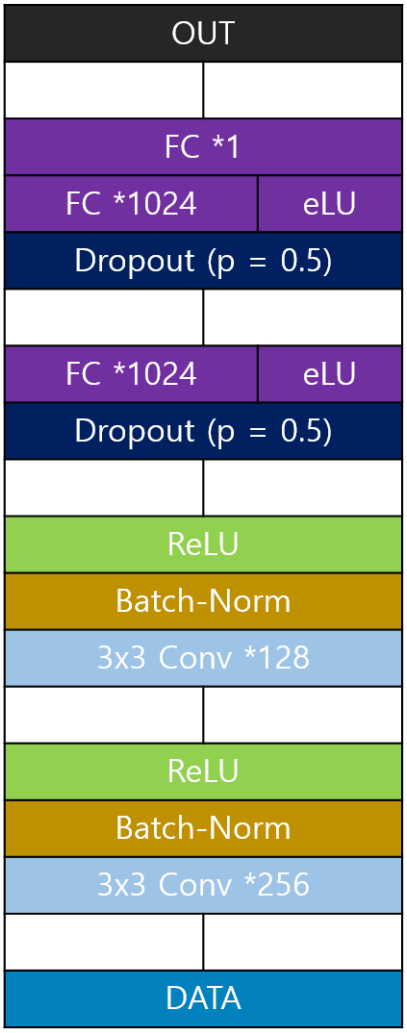

Figure 10. CNN model structure

VI. RESULT

We use $R^2 = 1 - \frac{\sum(y-\hat{y})^2}{\sum(y-\bar{y})^2}$ as a performance metric. Where y is the actual value, $\hat{y}$ is the predicted value of y, $\bar{y}$ is the mean of the y values. If R^2 close to 1, the predicted value becomes equal to the actual value. If R^2 close to 0, the predicted value becomes equal to the average of the actual values.

The R^2 score is calculated in the evaluation phase at the end of each epoch. We averaged R2 obtained from 10-fold cross validation.

The Table 1. shows the overall result. The resolution of the datasets (No.1 to No.4 in Table 1.) used in this experiment is 10 km x 10 km. 1x1 dataset R^2 recorded 0.7332, and 5x5 dataset R^2 increased to 0.7562. R2 increased as the surrounding area expanded and recorded 0.7801 at 9x9 dataset. We can see that using spatial correlation of CNN increase the performance in $PM_{2.5}$ concentration estimation problem. No.5 is the dataset used in [4] and has 1x1 spatial size and 1km resolution. This dataset was used in the Random Forest model and recorded r2 = 0.8. We measured performance with neural networks model for comparison. This Neural networks model was roughly tuned for this dataset. The result is R^2 = 0.7909, which can be higher than this result when using higher resolution data.

No.	Model	Dataset	Resolution	R^2
1	NN	1x1	10km	0.7332
2	CNN	5x5	10km	0.7562
3	CNN	7x7	10km	0.7792
4	CNN	9x9	10km	0.7801
5	NN	1x1	1km	0.7909

Table 1. Result on datasets

We compare the performance according to the kind of Activation function used. Table 3. shows the performance of the Activation function according to Sigmoid, Relu, Relu6, eLU, tanh, Softsign, and Softplus. We only changed the Activation function in Fully connected.

Activation Function	R^2
eLU	**0.7801**
Softsign	0.7704
ReLU6	0.7692
Sigmoid	0.7644
tanh	0.7466
Softplus	0.7291
ReLU	0.6690

Table 2. Result by activation function in fully connected layer

Depending on the active function used, there is a significant difference in performance. It is surprising that ReLU which is widely used recently has the lowest performance. Figure 11. shows Activation functions used in the experiment.

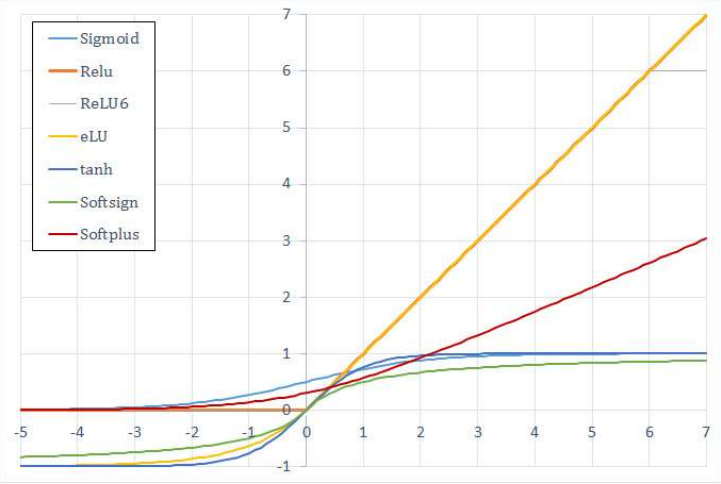

Figure 11. Activation functions

Figure 12. shows the information of the coordinates. Figure 12. (a) shows the difference between the annual mean prediction and the mean observation of specific coordinates. If the difference is larger, coordinate have redder color. Figure 12. (b) shows the ratio between missing points and valid points in the spatial dataset based on specific coordinates. If the coordinate color is red, there are many missing points, and if green, there are many valid points. The figure is based on a 9x9 spatial dataset. Because the dataset contains only US mainland information, there are many missing points near the sea, lake, and area of borderline. There are not many missing points in the spatial dataset based on the entire dataset. Table 3 shows how many valid points are among the data points of the spatial dataset. Wider the range, the fewer valid points, but all the datasets used in the experiment have more than 92.8% valid points. Since the positions of the red points in Figure 12. (top) and (bottom) are not similar, it seems that the missing points do not cause low prediction accuracy of the model. it seems to the model can handle missing points well.

Dataset	The number of valid points
1x1	Full
5x5	23.73 / 25 ≈ 94.9%
7x7	45.97 / 49 ≈ 93.8%
9x9	75.23 / 81 ≈ 92.8%

Table 3. The number of valid point

Figure 13. shows the relationship between observations and prediction. Each point represents the measurement and prediction values on the x, y-axis. The dotted line represents a linear line which is an ideal position of the points. The solid line represents the trend line of these points. Figure 13 shows the model validation for each season. Since the trend line is tilted

clockwise from the linear line, the model tends to overestimating. Winter is the worst among the four seasons. The reason is that it is difficult to measure AOD which one of most influential features for estimate PM2.5 because of snow or cloudy weather.

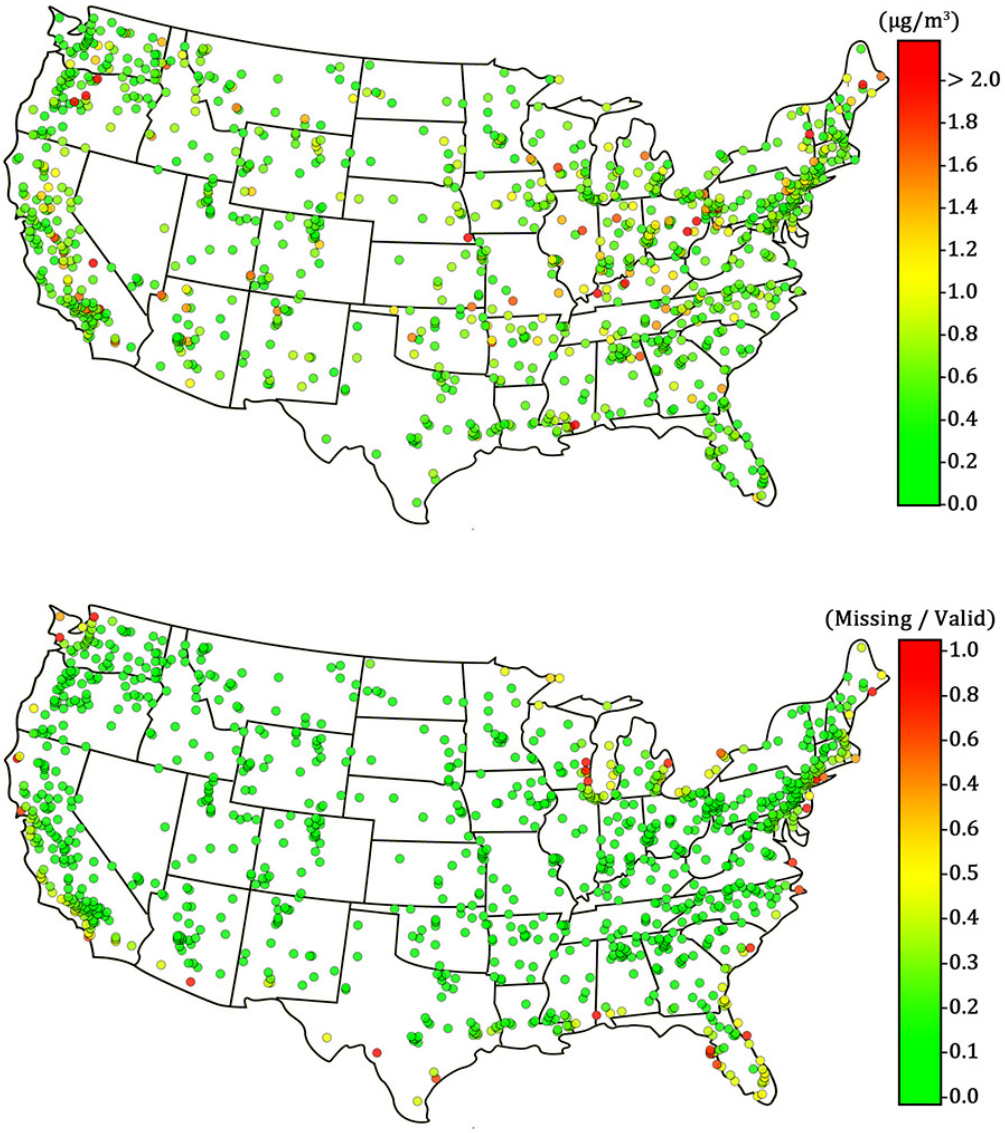

Figure 12. (top) Difference between annual mean prediction and measurement; (bottom) The ratio of valid point to missing point

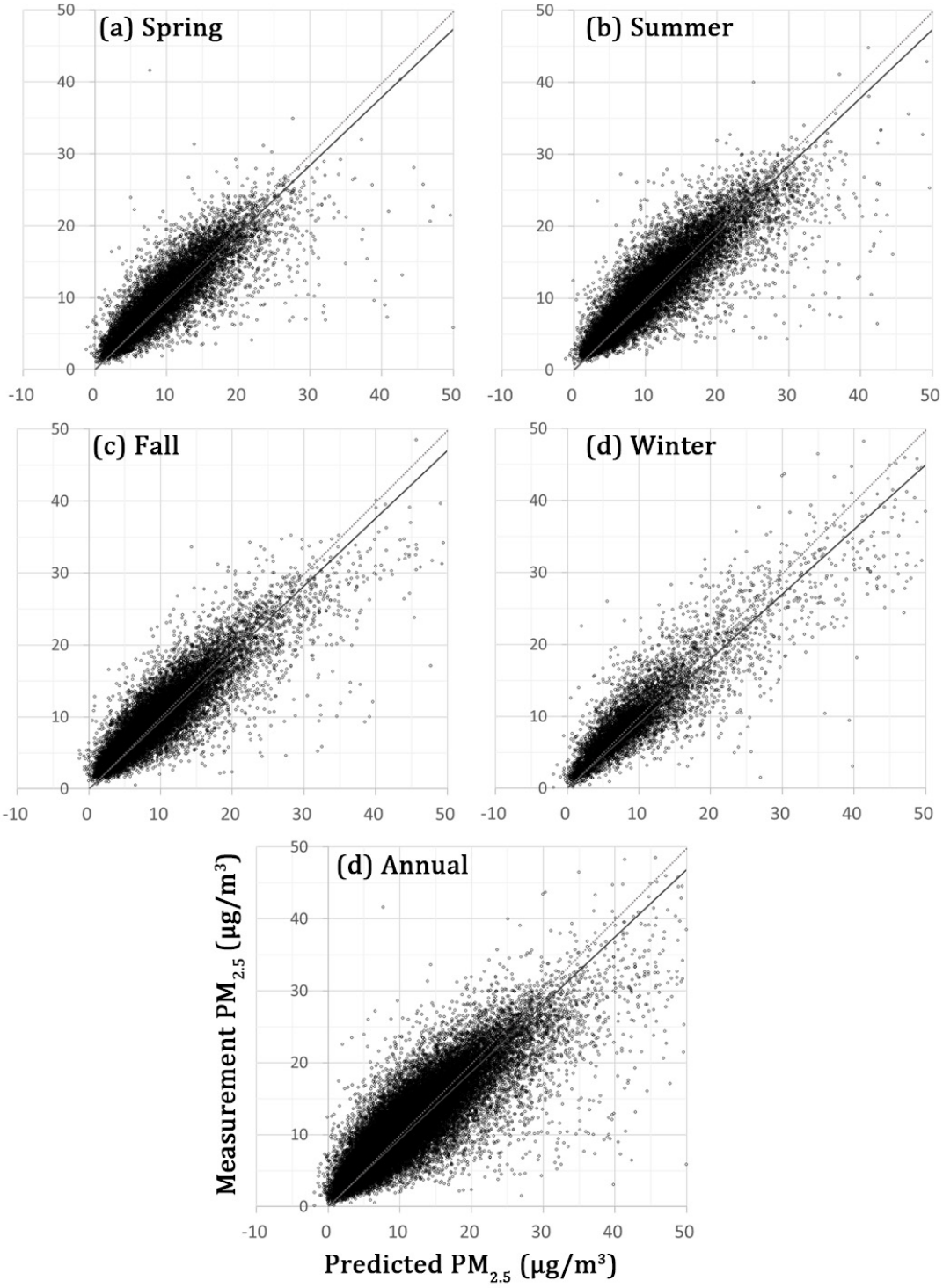

Figure 13. Model validation

VII. DISCUSSION

Our approach helps to improve performance by using additional spatial areas, and shows that CNN is suitable for this method. The statistical model of existing studies considers only one point. Also [4] uses random forest which is a learning base, but cannot effetely utilize the spatial area. However, this method requires more data size than one-point method. 9x9 dataset has a data size 81 times larger than 1x1 dataset. A 9x9 dataset has a data capacity 81 times larger than 1x1.

As explained in the Result, there is a performance difference according to the activation function. We need to find the reason and select the most appropriate activation function. For each activation function, we need to track how a node is activated in fully connected. Also, as shown in Figure 13, model validation tends to overestimate. If we correct this, we can expect better performance.

There are room for improvement through the extension of the CNN model. The structure of CNN used in this study is rough, the depth is shallow, and the parameter is small. Also, we expect high-performance improvement by using 1km resolution MAIAC AOD. Deep learning varies in performance depending on data quality. For example, [5] recorded R = 0.84 using MAIAC (1 km resolution AOD) despite using a simple NN model.

REFERENCE

[1] Schaap, M.; Apituley, A.; Timmermans, R. M. A.; Koelemeijer, R. B. A.; de Leeuw, G., "Exploring the relation between aerosol optical depth and $PM_{2.5}$ at Cabauw, the Netherlands." Atmos. Chem. Phys. 2009, 9, (3), 909-925.

[2] Levy, R., Collection 006 (C6) "Modis atmosphere l2 aerosol product. In 6 ed.; LAADS" Web: https://ladsweb.nascom.nasa.gov/data/search.html, 2014.

[3] Di, Q.; Kloog, I.; Koutrakis, P.; Lyapustin, A.; Wang, Y.; Schwartz, J., "Assessing $PM_{2.5}$ Exposures with High Spatiotemporal Resolution across the Continental United States." Environ. Sci. Technol. 2016, 50, (9), 4712-4721.

[4] X. Hu, J. Belle, X. Meng, A. Wildani, L. Waller, M. Strickland, Y. Liu "Estimating $PM_{2.5}$ Concentrations in the conterminous United States Using the Random Forest Approach" Under review for Environ. Sci. Technol.

[5] Di Q, Kloog I, Koutrakis P, Lyapustin A, Wang Y, Schwartz J. "Assessing $PM_{2.5}$ Exposures with High Spatiotemporal Resolution across the Continental United States." Environ Sci Technol. 2016 May 3;50(9):4712-21.

[6] Zongwei Ma, Xuefei Hu, Lei Huang, Jun Bi, and Yang Liu "Estimating Ground-Level $PM_{2.5}$ in China Using Satellite Remote Sensing" Environ. Sci. Technol., 2014, 48 (13), pp 7436–7444

[7] X. Hu1, L. A. Waller2, A. Lyapustin3, Y. Wang3,4, and Y. Liu1 "10-year spatial and temporal trends of $PM_{2.5}$ concentrations in the southeastern US estimated using high-resolution satellite data" Atmos. Chem. Phys., 14, 6301-6314, 2014

요 약 문

$PM_{2.5}$ 농도 추정을 위한 딥 컨볼루션 뉴럴 네트워크

최근 사람의 건강에 치명적인 초미세먼지($PM_{2.5}$)가 다양한 원인으로 인해 넓은 지역에서 나타나 큰 이슈가 되고 있다. 이로 인해 $PM_{2.5}$ 의 농도를 추정하는 기술이 중요해지고 있다.

이러한 이유로 $PM_{2.5}$ 농도를 추정하기 위한 많은 연구가 진행되고 있으며 최근에는 학습 기반의 모델을 이용한 연구가 활발히 이루어지고 있다. 이러한 학습 기반의 모델들은 기존의 통계적 모델의 정확도를 넘어서고 있다. 기존의 연구에 사용된 모델은 $PM_{2.5}$ 모니터링 지점이 있는 영역의 정보만을 이용하고 있다.

본 논문에서는 $PM_{2.5}$ 모니터링 지점의 데이터만 사용하는 기존의 방법과는 달리 $PM_{2.5}$ 모니터링 지점을 중심으로 주변 지역의 정보를 포함한 데이터를 이용함으로써 $PM_{2.5}$ 농도 추정의 정확성을 향상 시킬 수 있는 방법을 제시한다. 이를 위해서 공간 정보를 가진 3 차원 데이터로 가공하여 기존의 연구에서 사용된 vector 형태의 입력 데이터와의 비교를 통해 공간의 넓이에 따른 성능을 차이를 알아본다. 본 실험에 사용된 알고리즘은 공간 정보의 상관관계를 효율적으로 이용할 수 있는 Convolutional Neural Networks 를 사용한다. 실험 결과 CNN 을 이용하여 넓은 공간의 데이터를 이용할수록 높은 정확도를 내는 것을 확인하였다.

핵심어: Deep Learning, Neural networks, CNN, $PM_{2.5}$