STUDY ON FORECASTING OF WIND SPEED AND DIRECTION USING DEEP NEURAL NETWORK

SEPTEMBER 2021

ANGGRAINI PUSPITA SARI

Abstract

This research investigates forecasting of wind speed and direction based on deep learning to create an accurate forecasting model that can support the growth of wind power generation. The proposed forecasting model is composed of three dimensional convolutional neural network (3DCNN), deep convolutional long short-term memory (DCLSTM), two dimensional convolutional neural network (2DCNN) which called 3CNN-CLSTM-2CNN model. The proposed forecasting model combines the merit of convolutional neural network (CNN) and convolutional long short-term memory (CLSTM) for improving forecasting accuracy of wind speed and direction. The input of the forecasting model is time sequence images of wind speed and direction that represented an image on two dimensional coordinate (2D-coordinate). The actual observed data are taken from Automated Meteorological Data Acquisition System (AMeDAS), Japan. The proposed model forecasts one hour ahead and the forecasting accuracy of the proposed model was evaluated by the root mean square error (RMSE) between actual observed data and forecasted data. For verifying the effectiveness of the proposed model in comparison to fully connected long short-term memory (FC-LSTM), DCLSTM, 3CNN-CLSTM (3DCNN combines DCLSTM). It was confirmed that 3CNN-CLSTM-2CNN model is the strongest performance in all models. Besides that, 3CNN-CLSTM-2CNN model is compared in four cities (Tokushima, Takamatsu, Hiketa, and Choshi) to confirm an applicability to different characteristics of wind conditions. 3CNN-CLSTM-2CNN model is compared in four cities was evaluated by RMSE, mean absolute error (MAE), and mean absolute percentage error (MAPE) between forecasted data and actual observed data.

◇ ◇ ◇ CONTENTS ◇ ◇ ◇

◇ ◇ ◇ LIST OF FIGURES ◇ ◇ ◇

◇ ◇ ◇ LIST OF TABLES ◇ ◇ ◇

CHAPTER 1

Introduction

1.1 Overview

Along with increasing population and advancement economy in the world, so the amount of energy demand more growth that it consequences all countries facing a problem to suit energy demand and in 2018, global energy consumption dependence on fossil fuels still large at 79.9% [1]. Depletion of fossil fuels, many countries use alternative energy that is renewable energy to reduce dependency on fossil fuels so that can diminish poor effect on health risks, global warming or greenhouse gases effect due to methane (CH_4) and carbon dioxide (CO_2), and environmental pollution [2, 3]. Renewable energy can produce global electricity at 27.3% by the end of 2019 that is composed of hydropower, wind power, solar photovoltaic (PV), bio-power, geothermal, concentrated solar power (CSP), and ocean power is shown in Figure 1.1 [1]. The goal of renewable energy sources can contribute to supply energy at 47.7 % in 2040 [3]. Renewable energy sources potentially provide energy almost zero-emissions of greenhouse gases and air pollutants [4, 5].

Wind power, among kinds of renewable energy that has benefits in low emission of CO2, clean, green and low cost of electrical power generation. Wind power has rapidly attracted attention as alternative electrical power generation and the global fastest-growing of renewable energy. From Figure 1.1, wind power can contribute to global electricity production by 5.9%. The trend of global installed wind power capacity is 650.76 GW by the end of 2019 and has significantly increased every year (2005 - 2019) shown in Figure 1.2 [6]. Likewise in Japan, the trend installed wind power capacity is 4.38 GW by the end of 2019 and increasing significantly rather than one year before that is 3.86 GW that be increased 11.87% [7]. Figure 1.3 indicates ten top countries of wind cumulative installed capacity in

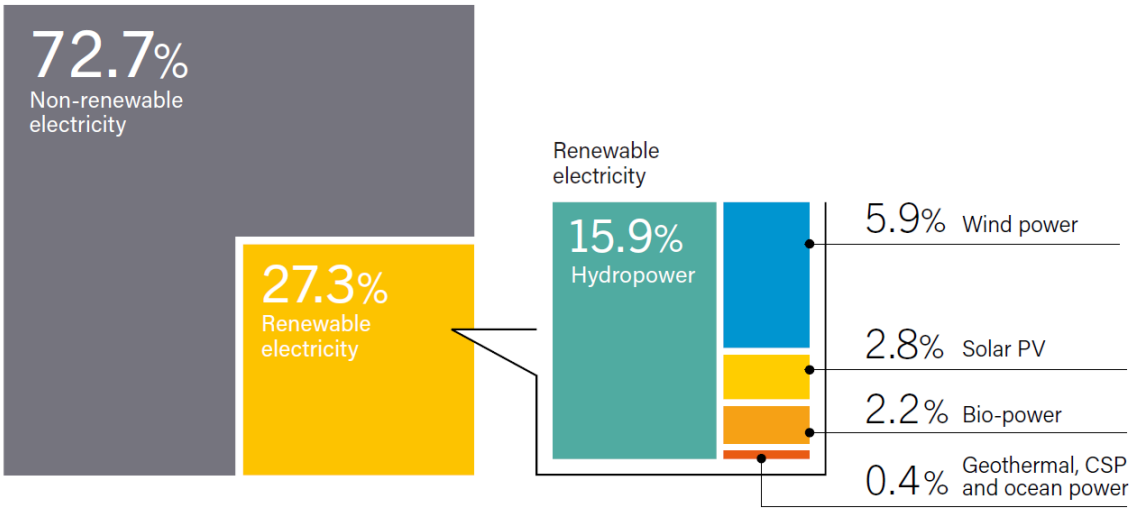


Figure 1.1: Global electricity production, end 2019

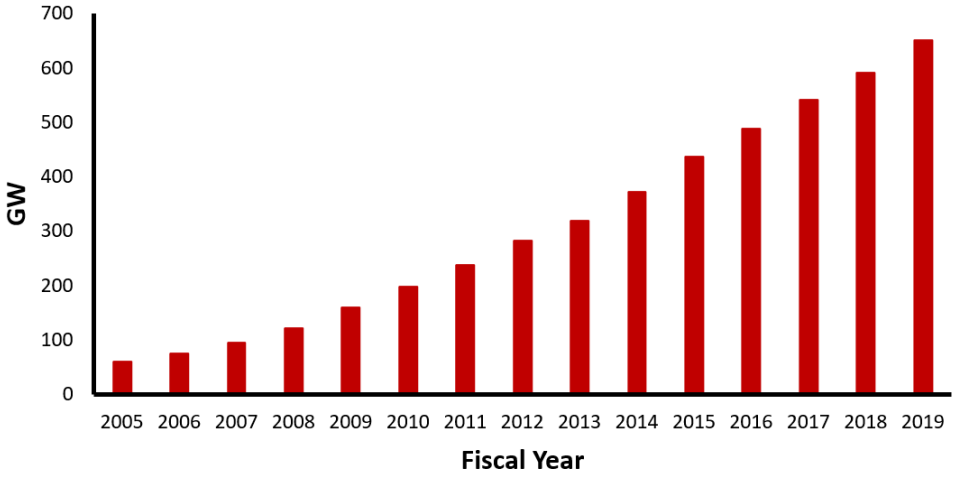


Figure 1.2: Global installed wind power capacity 2005 - 2019

2019 [8]. The greatest percentage country of wind installed was China. The second position was the United States that same percentage as the rest of the world. Japan be included in the rest of the world with a percentage of 0.7% (4.38 GW) in 2019.

1.2 Problem statement

Each renewable energy has the problem to increase electricity production included wind power generation. Wind power generation has the main problem that widely outputs fluctuation due to change of wind direction and direction so that difficult for controlling the production of wind power generation. Cause this problem, the electrical company maintains

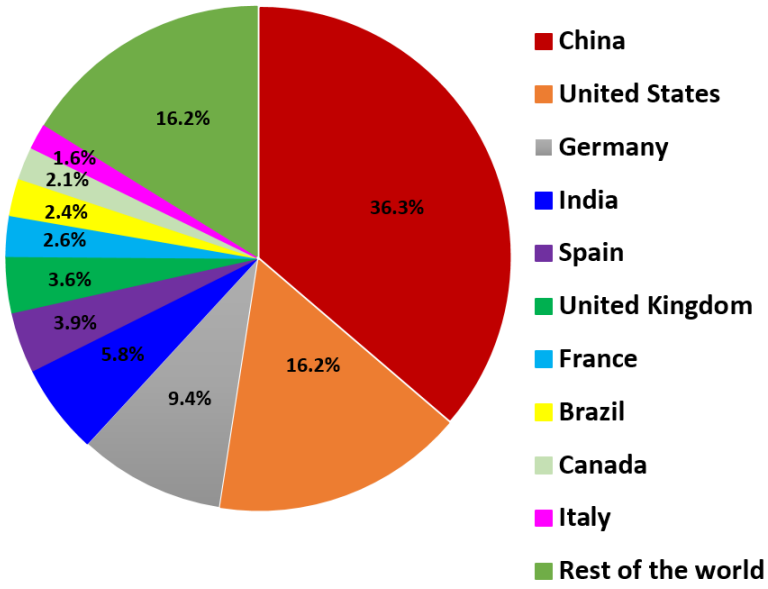


Figure 1.3: Ten top countries of wind cumulative installed capacity in 2019

electrical supply by conventional power generation. Therefore, to solve this problem so that required accurate forecasting of wind speed and direction for reducing uncertainty generated a sum of energy in the future. Moreover, this information can offer to the electrical company is useful for organizing thermal power plant output and stabilizing the electrical power system.

The forecasting wind method has three kinds: physical method, statistical method, and artificial intelligence (AI) method. Physical methods such as computational fluid dynamics (CFD), numerical weather prediction (NWP), mesoscale model (MSM) is appropriate to middle term, long term, large scale area forecast wind, can get an accurate forecast wind but need full information about a physical factor like temperature, surface roughness, terrain, weather, hub height, atmospheric data; high computational cost; an impossible account to local topography; and limited spatial and temporal resolution [9–11]. Statistical methods such as clustering analysis, Kringing interpolation, time-series analysis, Von Mises distribution, auto-regressive integral moving average (ARIMA) use the relationship between observed data and forecasted data to represent model parameter, finding a mathematical model for expressing time series, great perform for learning linear features but limited for learning non-linear features [9, 12–15]. AI method such as support vector regressor (SVR), neural network (NN), adaptive neural fuzzy inference system (ANFIS), fuzzy logic are good performs for learning complex non-linear features, forecast the future of wind sequence, and to discover the relationship between output data and input data non-statistically. [13–16].

Consequently, this doctoral thesis proposes a forecasting model of wind speed and direction utilize deep NN (DNN) by 3CNN-CLSTM-2CNN. In hopefulness, the proposed forecasting model can improve forecasting accuracy.

1.3 Research objectives

Each research has to vary objectives depends on object research. This research has objectives as follows:

1. For determining and proposing an accurate forecasting model to wind speed and direction

The doctoral thesis proposes a forecasting model of wind speed and direction utilize AI method that is DNN by 3CNN-CLSTM-2CNN. 3CNN-CLSTM-2CNN model combines three dimensional convolutional neural network (3DCNN), deep convolutional long short-term memory (DCLSTM), two dimensional convolutional neural network (2DCNN). The function of 3DCNN and 2DCNN for extracting spatial and temporal features. The function of CLSTM for extracting spatio-temporal features and analyzing time sequence image data. The input of the forecasting model utilizes wind speed and direction are converted as an image on two dimensional coordinate (2D - coordinate).

2. For improving forecasting accuracy

For confirming improving forecasting accuracy and the effectiveness of the forecasting model is evaluated by the root mean square error (RMSE) between actual observed data and forecasted data. The 3CNN-CLSTM-2CNN model is simulated several times every season in one year that is characteristic of Japanese climate and the efficiency of the proposed model is evaluated by compared fully connected long short-term memory (FC-LSTM), DCLSTM, and 3CNN-CLSTM (3DCNN combines CLSTM). 3CNN-CLSTM-2CNN model is compared in four cities was evaluated by RMSE, mean absolute error (MAE), and mean absolute percentage error (MAPE) between forecasted data and actual observed data.

1.4 Research scope and limitations

There are three main methods for forecasting wind such as explained in section 1.2. This research focused on forecasting wind speed and direction utilize the AI method that is the DNN model. The research limitations as follows:

1. The input of the forecasting model only utilizes wind speed and direction are converted

as an image on 2D - coordinate.
2. Data of wind speed and direction are taken from Automated Meteorological Data Acquisition System (AMeDAS), Japan at one hour interval.
3. The forecasting period only uses one hour ahead in forecasting model of wind speed and direction that forecasting point in Tokushima city, Japan.

Besides, 3CNN-CLSTM-2CNN model applied to other cities that are Takamatsu, Hiketa, and Choshi cities for confirming proposed model can handle other cities that have different characteristics from Tokushima city.
4. The forecasting program only uses python with framework Keras and Tensorflow as Backend.

1.5 Structure of doctoral thesis

This research proposed a novel forecasting model for wind speed and direction by DNN. This doctoral thesis was composed of six chapters.

Chapter 1 describes the introduction of the proposed forecasting model includes an overview, problem statement, objectives, research scope, research limitation, and structure of the doctoral thesis.

Chapter 2 explains the fundamentals of the neural network and activation function used to compose the proposed forecasting model. The Fundamentals of NN be composed of multilayer perceptron (MLP), recurrent neural network (RNN), long short-term memory (LSTM), convolutional neural network (CNN), and convolutional long short-term memory (CLSTM). The activation function is composed of linear and non-linear activation functions.

Chapter 3 explains the proposed forecasting model of wind speed and direction is composed of an explanation of wind speed and direction; dataset; learning parameters and procedures; and model configuration. Model configuration presents FC-LSTM model, DCLSTM model, 3CNN-CLSTM model, and 3CNN-CLSTM-2CNN model.

Chapter 4 explains the forecasting accuracy of wind speed and direction using the 3CNN-CLSTM-2CNN model. Furthermore, the effectiveness 3CNN-CLSTM-2CNN model is eval-

uated by comparison FC-LSTM, DCLSTM, and 3CNN-CLSTM models. The forecasting period uses one hour ahead in Tokushima city, Japan. This chapter presents mathematical model to get forecasting results, forecasting result of v_X and v_Y, and forecasting results of wind speed and direction.

Chapter 5 explains regional dedendence of 3CNN-CLSTM-2CNN model in four cities that are Tokushima, Takamatsu, Hiketa, and Choshi cities for confirming the proposed model can handle and applicability to different characteristics of wind conditions. This chapter is composed of wind speed and direction condition in four cities, forecasting results of v_X and v_Y in four cities, and forecasting results of wind speed and direction in four cities.

Chapter 6 explains conclusions the content of this study which was conducted.

CHAPTER 2

Fundamentals

This chapter explains the general theory about a neural network (NN) that are used in the forecasting model and activation function. Firstly, section 2.1 introduces multilayer perceptron (MLP). After that, recurrent neural network (RNN), long short-term memory (LSTM), convolutional neural network (CNN), and convolutional long short-term memory (CLSTM) explained in Sections 2.2 - 2.5. The end of this chapter explains the general theory of activation function. In this section presents linear and non-linear activation functions.

2.1 Multilayer perceptron

Neural network (NN) is a computing model with a simple structure, small neuron network, and interconnected each unit with weights. Multilayer perceptron (MLP) is the basic form of NN. Basically, MLP consist of the input layer, middle layers, and output layer as shown in Figure 2.1 [17]. The number of neurons depends on the given problem. The middle layer can be called the hidden layer that can any number of layer, neuron, and depth.

MLP arranges neurons in every layer and one layer of neurons is connected to neurons in the previous layer by feed-forward. Every neuron in a layer is fully connected to neurons in the next layer. Each layer of MLP connected weights and can with or without bias. From Figure 2.1, MLP uses bias that each weight and bias updated by backpropagation (BP). MLP can be matching input data to output data that one input one output and multiple inputs multiple outputs. Besides that, MLP can apply to one input multiple outputs and multiple

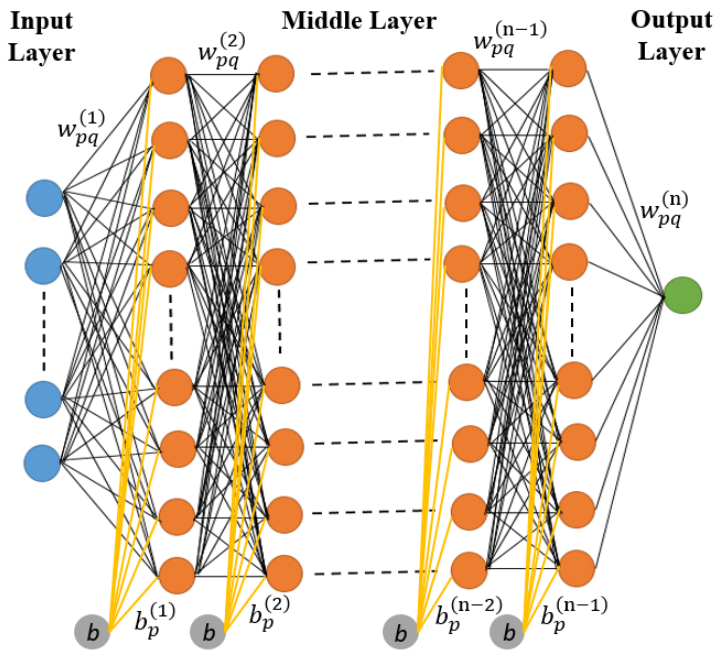


Figure 2.1: Basic structure of MLP

inputs one output. The mathematical model of MLP as follows [2, 18],

$$\boldsymbol{h}_p^1 = \sigma\left(\sum_{q=1}^{M} \boldsymbol{w}_{pq}^{(1)} \boldsymbol{X}_q^{(1)} + b_p^1\right) \tag{2.1}$$

$$\boldsymbol{h}_p^k = \sigma\left(\sum_{q=1}^{M} \boldsymbol{w}_{pq}^{(k)} \boldsymbol{h}_q^{(k-1)} + b_p^k\right) \tag{2.2}$$

$$\boldsymbol{o}_p = \sigma\left(\sum_{q=1}^{M} \boldsymbol{w}_{pq}^{(n)} \boldsymbol{h}_q^{(n-1)} + b_p^n\right) \tag{2.3}$$

where M is number of input data, $\boldsymbol{w}$ is weight, $\boldsymbol{X}$ is input, $\boldsymbol{h}$ is hidden, b is bias, $\boldsymbol{o}$ is output and σ is activation function. In MLP, the activation function uses non-linear activation function.

2.2 Recurrent neural network

Recurrent neural network (RNN) is a type of NN which can resolve time-series data. RNN increases the ability of MLP by adding a memory unit to save previous input and output that used to current computation. RNN has three layers consist of the input layer (X), hidden layer (h), and output layer (o) is shown in Figure 2.2. From Figure 2.2, the hidden layer gets

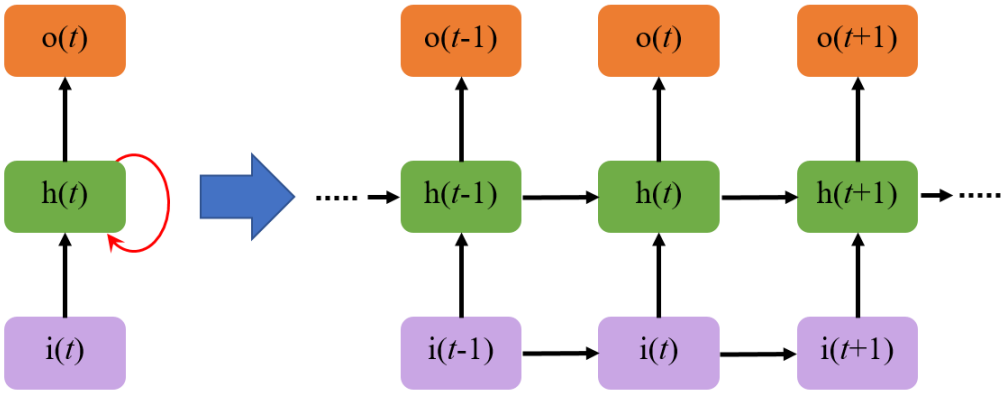


Figure 2.2: Structure of RNN

input from the input layer and the hidden layer from the previous time step. Mathematical calculation of RNN as follows [19, 20],

$$\boldsymbol{h}_t = \sigma(\boldsymbol{w}_{Xh} \cdot \boldsymbol{X}_t + \boldsymbol{w}_{hh} \cdot \boldsymbol{h}_{t-1} + b_h) \tag{2.4}$$

$$\boldsymbol{o}_t = \sigma(\boldsymbol{w}_{ho} \cdot \boldsymbol{h}_t + b_h) \tag{2.5}$$

where $\boldsymbol{w}$ means weight, σ means activation function and b means bias.

RNN can apply to some structures that are one input to one output, one input to many outputs, many inputs to one output, and many inputs to many outputs. RNN can be applied in action recognition, data-driven traffic, vehicle trajectory, etc [19, 21]. RNN has primary problem as regards vanishing gradient and LSTM is used to resolve it problem.

2.3 Long short-term memory

Long short-term memory (LSTM) is a advance type of RNN which resolve vanishing gradient problem effectively caused LSTM trained by BP algorithm, improve capability of RNN, run faster than RNN, make easy to converge, has gates and cell state to control information flow [22, 23]. LSTM is stable and powerful to learns long range dependency and can handles short and long-term dependencies effectively that applied by three gates: input gate for controlling admission of information, forget gate for controlling retain of information, and output gate for controlling output of information [5, 18, 24].

LSTM is famous choice to learn sequence tasks and compatible for end to end learning. LSTM can be applied to scene label [25], wind prediction [26], medical diagnostic [27], speech recognition [28], caption generation [29], etc. Figure 2.3 shows core structure of

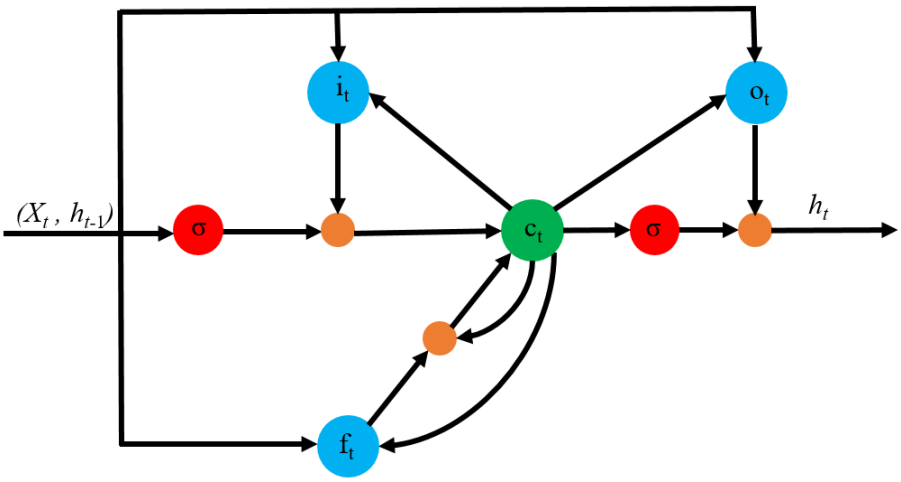


Figure 2.3: Core structure of LSTM

LSTM that is composed of cell state (c), hidden state (h), input gates (i), forget gate (g), output gate (o), input data (X), and activation function (σ) [18, 24]. Typically, the equation of LSTM as shown in Eqs. 2.6 – 2.10 [30, 31],

$$\boldsymbol{i}_t = \sigma(\boldsymbol{w}_{Xi} \cdot \boldsymbol{X}_t + \boldsymbol{w}_{hi} \cdot \boldsymbol{h}_{t-1} + \boldsymbol{w}_{ci} \circ \boldsymbol{c}_{t-1} + b_i) \tag{2.6}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \circ \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \circ \tanh(\boldsymbol{w}_{Xc} \cdot \boldsymbol{X}_t + \boldsymbol{w}_{hc} \cdot \boldsymbol{h}_{t-1} + b_c) \tag{2.7}$$

$$\boldsymbol{f}_t = \sigma(\boldsymbol{w}_{Xf} \cdot \boldsymbol{X}_t + \boldsymbol{w}_{hf} \cdot \boldsymbol{h}_{t-1} + \boldsymbol{w}_{cf} \circ \boldsymbol{c}_{t-1} + b_f) \tag{2.8}$$

$$\boldsymbol{o}_t = \sigma(\boldsymbol{w}_{Xo} \cdot \boldsymbol{X}_t + \boldsymbol{w}_{ho} \cdot \boldsymbol{h}_{t-1} + \boldsymbol{w}_{co} \circ \boldsymbol{c}_t + b_o) \tag{2.9}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \circ \tanh(\boldsymbol{c}_t) \tag{2.10}$$

where b means bias, $\boldsymbol{w}$ means weight and $\circ$ means hadamard product. Activation function uses non-linear activation function in LSTM.

2.4 Convolutional neural network

Convolutional neural network (CNN) is a type of NN based on feed forward neural network (FFNN). CNN is efficient and suitable to applied and process some dimensions: 1D (sound, waveform), 2D (images), 3D (video), 4D (multi video) [32–36]. CNN is most famous to learns image and video that includes scaling, rotation, translation. CNN can resolve many computer vision problems: object detection, semantic segmentation, image generation, etc [37, 38]. The kind of CNN architecture is composed of LeNet, AlexNet, ZFNet (Clarifai), network using blocks (VGG Net), network in network (NiN), network with parallel

concatenations (GoogleNet), residual networks (ResNet), FractalNet and densely connected networks (DenseNet) [21, 39].

The simple architecture of CNN is composed of input, convolutional layer, pooling layer, fully connected (FC) layer, and output as shown in Figure 2.4 [21]. The process of extract feature maps repeated several times every step iteration and gets results in multiple feature maps in every convolutional layer. Every layer in CNN uses the kernel for holding randomly initialize parameters. In two dimensional, the kernel has height and width represented kernel size. Each convolutional layer uses non-linear activation function. The k-th convolutional layer as follows [18] :

$$\boldsymbol{h}_{pq}^{k} = \sigma((\boldsymbol{w}^{k} * \boldsymbol{X})_{pq} + b_k) \tag{2.11}$$

where $''*''$ means convolutional operation, $\boldsymbol{w}$ means weight of the filter, $\boldsymbol{X}$ means input, σ is activation function, and b is the bias of the filter. Activation function uses non-linear activation function in CNN. The pooling layer has two kinds that narrowed and enlarged of input size and works with the kernel. The pooling layer to narrow of input size has two kinds that are max-pooling layer and average pooling layer as shown in Figure 2.5. The pooling layer to enlarge of input size is called the up-sampling layer as shown in Figure 2.6 [21,40]. FC layer is connected to every neuron with the previous layer and the advantage to take high-level feature maps as input and return output in the shape of the classification vector. Convolutional and FC layers have learnable parameters such as bias and weight but the pooling layer didn't have it.

The other hyperparameters in convolution are padding and stride. Padding has three models: full, same, and valid paddings. The Full padding is used to increase dimensional of output but infrequently used in CNN. The same padding is used to get the same size of input and output and also most commonly used in CNN. The valid padding is computing

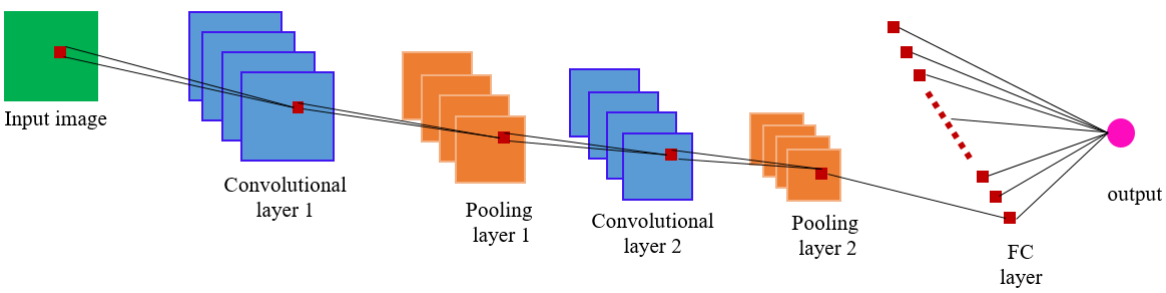


Figure 2.4: Simple structure of CNN

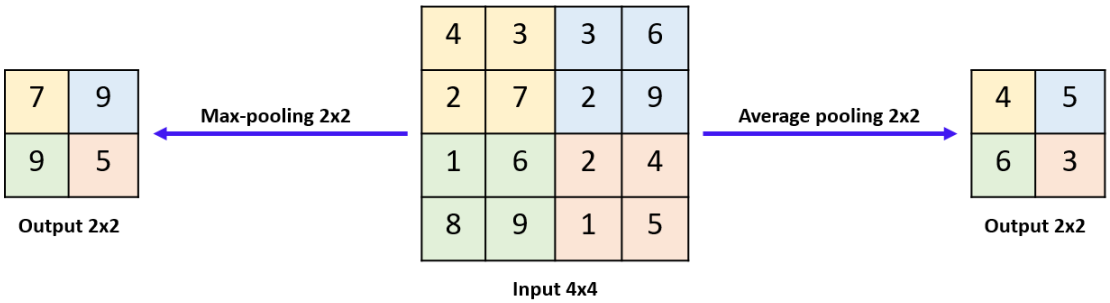


Figure 2.5: Max-pooling and average pooling operations

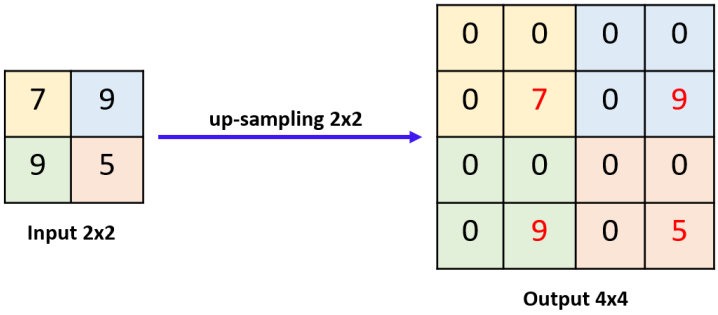
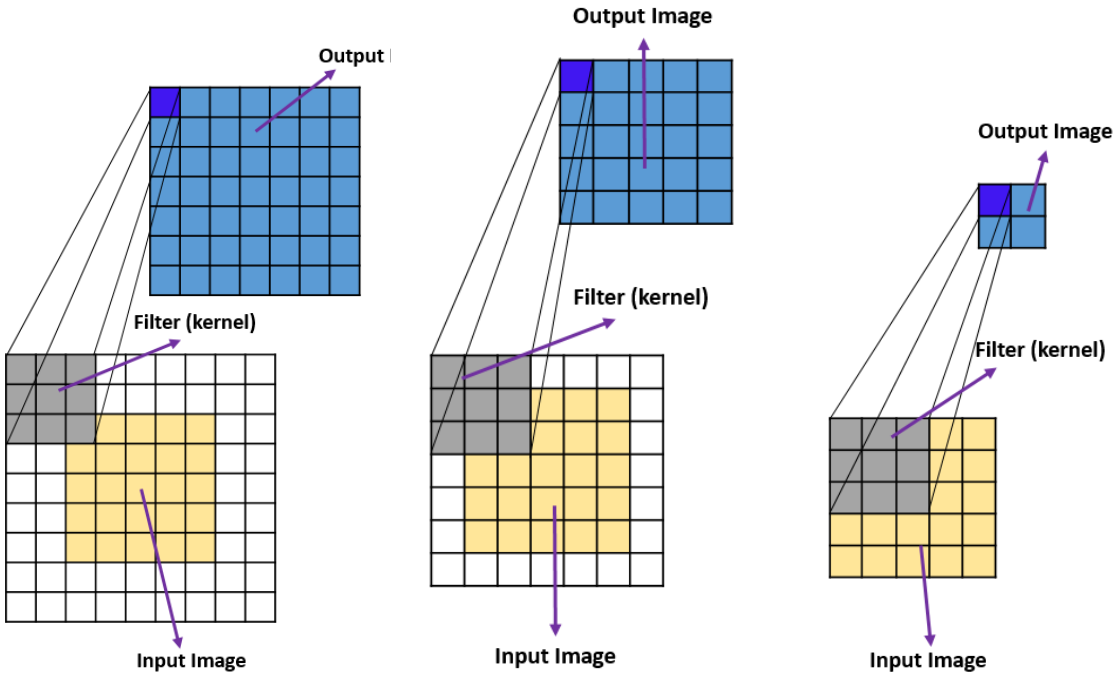


Figure 2.6: Up-sampling operations

convolution without padding. Stride is used to modifying the movement of an image or video. For example, the stride of 1 that meaning the filter will move one unit or pixel at a time. Figure 2.7 shows illustrates three models of padding with 5x5 size of the input, 3x3 size of the kernel, and stride of 1. In the same padding, the output size of convolution can determine by [19],

$$o = \frac{X + 2p_s - k_f}{s} + 1 \tag{2.12}$$

where o means output size, X means input size, p_s means padding, k_f means size of filter (kernel), and s means stride.



(a) Full padding (b) Same padding (c) Valid padding

Figure 2.7: Padding operations

2.5 Convolutional long short-term memory

Convolutional long short-term memory (CLSTM) combines advantages CNN and LSTM for solving image sequences which improve the LSTM model using convolutional structure in input-to-state transition and state-to-state transition as shown in Figure 2.8. Similarly LSTM unit, CLSTM unit has cell state (c), hidden state (h), input gates (i), forget gate (g), output gate (o), input data (X), activation function (σ) and a set of weights (w) for weighted connections between input-to-state, state-to-state, and state-to-output. Activation function uses non-linear activation function in CLSTM. The CLSTM can learn and resolve sequence to sequence process, produce the same size of input and output images, and learn spatio-temporal features.

The CLSTM can receive feature maps arranged five dimensions: number of samples data (N), time-series (t_s), height (H), width (W), and the number of channels (C). The product operation $''\cdot''$ in equation of LSTM is replaced by convolutional operation $''*''$ in equation of CLSTM. The equation of CLSTM as shown in Eqs. 2.13 – 2.17 [23, 31].

$$\boldsymbol{i}_t = \sigma(\boldsymbol{w}_{Xi} * \boldsymbol{X}_t + \boldsymbol{w}_{hi} * \boldsymbol{h}_{t-1} + \boldsymbol{w}_{ci} \circ \boldsymbol{c}_{t-1} + b_i) \tag{2.13}$$

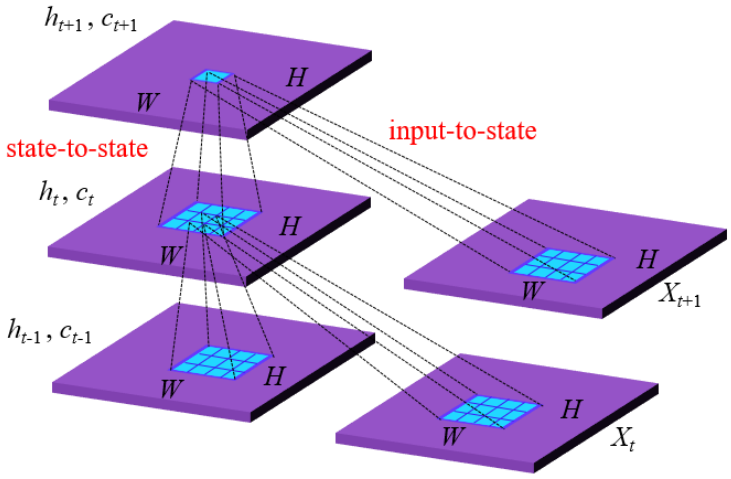


Figure 2.8: Core structure of CLSTM

$$\boldsymbol{f}_t = \sigma(\boldsymbol{w}_{Xf} * \boldsymbol{X}_t + \boldsymbol{w}_{hf} * \boldsymbol{h}_{t-1} + \boldsymbol{w}_{cf} \circ \boldsymbol{c}_{t-1} + b_f) \tag{2.14}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \circ \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \circ \tanh(\boldsymbol{w}_{Xc} * \boldsymbol{X}_t + \boldsymbol{w}_{hc} * \boldsymbol{h}_{t-1} + b_c) \tag{2.15}$$

$$\boldsymbol{o}_t = \sigma(\boldsymbol{w}_{Xo} * \boldsymbol{X}_t + \boldsymbol{w}_{ho} * \boldsymbol{h}_{t-1} + \boldsymbol{w}_{co} \circ \boldsymbol{c}_t + b_o) \tag{2.16}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \circ \tanh(\boldsymbol{c}_t) \tag{2.17}$$

CLSTM can be applied to wind prediction [5], rainfall intensity [23], solar irradiation [41], text recognition [42], gesture recognition [43], etc.

2.6 Activation function

Basically, the activation function has two kinds: linear and non-linear activation functions. Linear activation function is used for preserving a constant and non-linear activation function is used for producing more variation that uses NN.

2.6.1 Linear activation function

The function of a linear activation function is similar to a straight line that activation comparable to the input and the mathematical model as follows,

$$f(X_r) = kX_r \tag{2.18}$$

where X_r means input of activation function, k means fixed constant, and $f(X_r)$ means appropriate output of the input. If all layers use linear activation function regardless the number

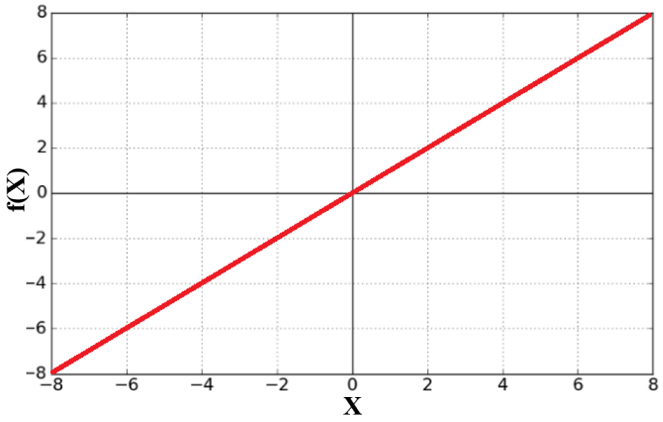


Figure 2.9: Linear activation function

of layers, so all network is equal to a single layer with a linear activation function [44]. The curve of linear activation function as shown in Figure 2.9.

2.6.2 Non-linear activation function

Non-linear functions are more used and make it easy for adapting various data and to distinguish among output. Common non-linear functions that often used consist of sigmoid, hyperbolic tangent function (tanh), rectified linear unit (ReLU), and Leaky rectified linear unit (Leaky ReLU).

The sigmoid function is used in NN. The equation of sigmoid function as follows,

$$f(X_r) = \frac{1}{1 + e^{-X_r}} \tag{2.19}$$

where $f(X_r)$ is appropriate output of the input, X_r is input of activation function, and k is fixed constant. Characteristic of sigmoid function can consistent with neurons synapsis in neurology, derivative can easy to get but now, rarely used due to it has characteristic of soft saturability in a sigmoid curve. This meaning, a slope of the graph incline to zero if the input is very small or very big. When the slope of function approaches zero, the gradient continued to the underlying network to be very small so makes network parameter difficult to be effectively trained [44, 45]. Figure 2.10 shows curve of sigmoid function.

Tanh function is used in NN and updated from the sigmoid function. The equation of tanh function as follows,

$$f(X_r) = \frac{1 - e^{-2X_r}}{1 + e^{-2X_r}} \tag{2.20}$$

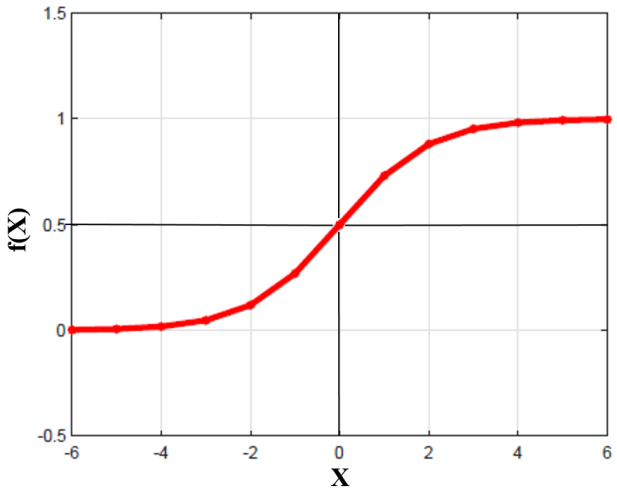


Figure 2.10: Sigmoid function

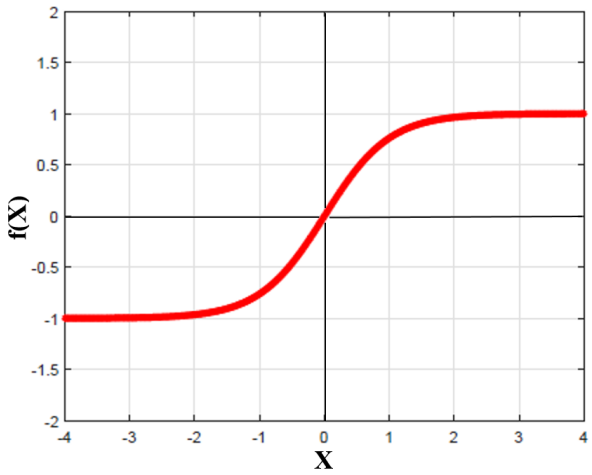


Figure 2.11: Tanh function

The main merit of tanh function produces a symmetric function zero centered output. Performance of tanh function better training than sigmoid function in multilayer neural networks and convergence rate is higher than sigmoid function but has a gradient diffusion problem. The limitation of tanh function so has further research and get ReLU function [44–46]. Figure 2.11 shows curve of tanh function.

ReLU function is a popular activation function in NN. The equation of ReLU function as follows,

$$f(X_r) = \max(0, X_r) \begin{cases} X_p, & X_r > 0 \\ 0, & X_r < 0 \end{cases} \tag{2.21}$$

ReLU function is faster computation and learning than sigmoid and tanh functions. The

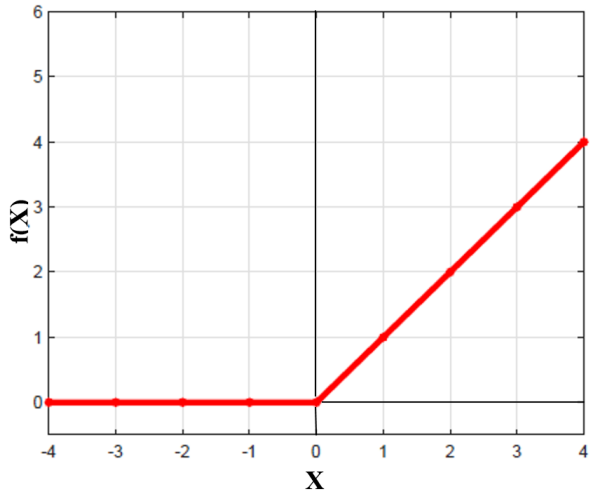


Figure 2.12: ReLU function

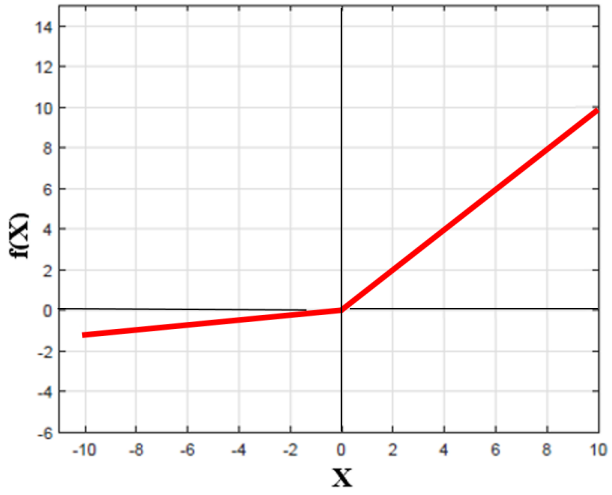


Figure 2.13: Leaky ReLU function

limitation of ReLU function is sometimes brittle during training that effect activation of weight update is off so gives zero activation or is called dying ReLU. For resolving the limitation ReLU function that uses Leaky ReLU [44–46]. The curve of ReLU function as shown in Figure 2.12.

Leaky ReLU function has a small negative slope to keep and sustain weight update during training process. The equation of Leaky ReLU function as follows:

$$f(X_r) = \begin{cases} a \cdot X_r, & X_r < 0 \\ X_r, & X_r \geq 0 \end{cases} \tag{2.22}$$

From Eq. 2.22, a parameter is used to resolve dying ReLU so gradient didn't get zero during

training time [44–46]. The curve of Leaky ReLU function as shown in Figure 2.13.

CHAPTER 3

Forecasting Model of Wind Speed and Direction

This chapter explains the proposed forecasting model that firstly explains the definition of wind speed and direction in section 3.1. Afterward, section 3.2 presents the dataset. The dataset consists of training data, validation data, and test data. In section 3.2, the process to get the input image is explained. learning parameters and procedures explains in section 3.3. The last section in this chapter explains forecasting models. The forecasting models be composed of FC-LSTM, DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models.

3.1 Wind speed and direction

The wind speed and direction (wind information) are taken from AMeDAS, Japan at one hour interval in Tokushima city, Japan. besides that, this research applies the forecasting model to other cities in Japan which are Takamatsu, Hiketa, Choshi. It meant to verify the forecasting model can suitable or not for other cities that will investigate more deeply in section 5. Wind direction expressed in 16 direction as shown in Figure 3.1. The wind information can be represented on 2D-coordinate system by graphic expression as shown in Figure 3.2. The wind information expressed $v_X(t)$ and $v_Y(t)$ on 2D-coordinate system, that X and Y-axis accord east (E) - west (W) and north (N) - south (S) directions. From Figure 3.2(b), mathematical model of wind vector components $v_X(t)$ and $v_Y(t)$ are calculated by

$$v_X(t) = v(t) \cdot \cos\varphi(t) \tag{3.1}$$

$$v_Y(t) = v(t) \cdot \sin\varphi(t) \tag{3.2}$$

where $v(t)$ is wind speed [m/s] and $\varphi(t)$ is wind direction [°]. The proposed model forecasts wind information one hour ahead by considering the capability to pumped-storage hydro-electricity and adapt thermal power plant.

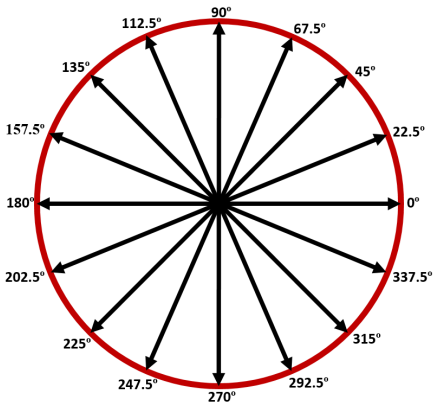
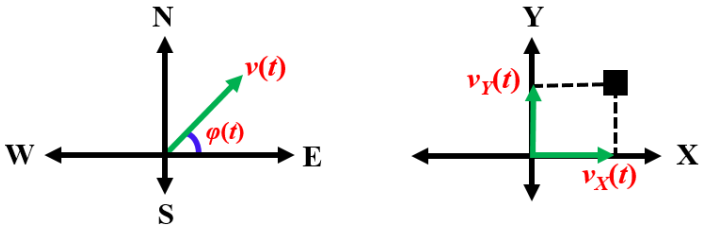


Figure 3.1: Explanation of wind direction



(a) (b)

Figure 3.2: Wind speed and direction on 2D-coordinate system (a) Vector diagram (b) Explanation of component $v(t)$

3.2 Dataset

Wind information is converted to $v_X(t)$ and $v_Y(t)$ by Eqs. 3.1 and 3.2 and also plotted to image as shown in Figure 3.2(b). The image is used to input and output data of the forecasting model and it sizes are set to 128 × 128 pixels. From the training, validation, and test data, the maximum of wind speed is used to plotting scale customized the size of the input image and plotted to $p_X(t)$ and $p_Y(t)$ using python imaging library (PIL) by

$$p_X(t) = 64 + \frac{64}{v_{max}} v_X(t) \tag{3.3}$$

$$p_Y(t) = 64 - \frac{64}{v_{max}} v_Y(t) \tag{3.4}$$

where 64 means half of image size and v_{max} means maximum of wind speed. The value of v_{max} for Tokushima, Takamatsu, and Hiketa cities are 20 m/s. The value of v_{max} for Choshi city is 30 m/s.

In order for expressing a change of wind information is used six points at time p(t-5) to p(t) which is connected and plotted by line that describes one hour data as shown in Figure

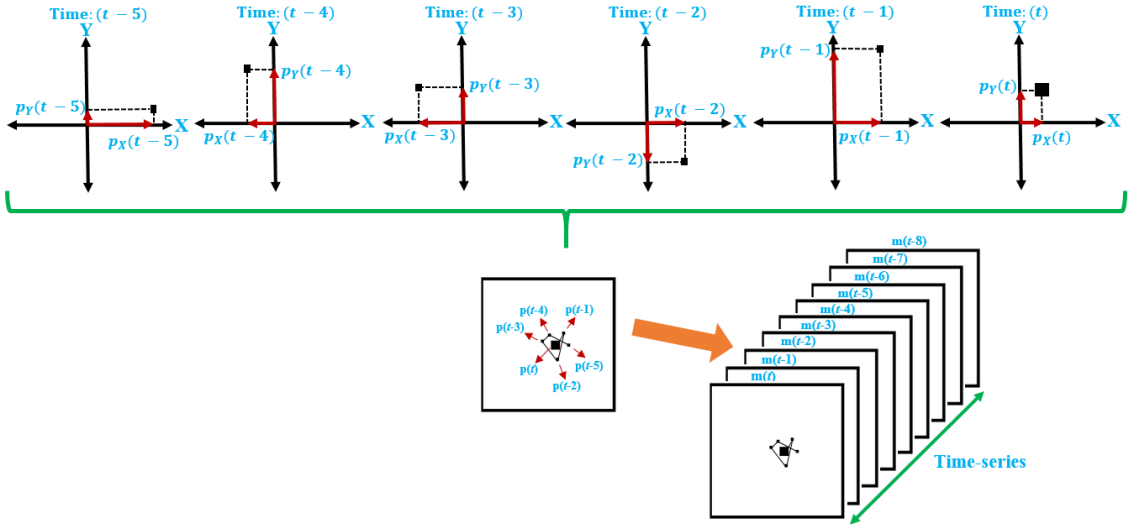


Figure 3.3: Process to get input image

3.3. The point of current data in the image is plotted biggest point for making it different and expressing time-series. In addition, the time series image uses nine images from m(t-8) to m(t) that expressing the change transition of the wind vector. This paper is used wind information for four years: training data uses two years, validation data uses one year and test data uses one year as shown in Table 3.1.

Table 3.1: Dataset of period

Dataset	Period
Training	Dec. 2015 - Nov. 2017
Validation	Dec. 2017 - Nov. 2018
Test	Dec. 2018 - Nov. 2019

3.3 Learning parameters and procedures

The forecasting model is implemented in python with framework Keras and Tensorflow as backend. Learning parameters of the proposed model as shown in Tabel 3.2. The process of training and validation were iterated 20 epochs. For training of the model using root mean square propagation (RMSProp) optimizer with learning rate (lr) and decay factor (ρ) parameters. Leaky rectified linear unit (Leaky ReLU) applied as an activation function to speed up

Table 3.2: Learning parameters of proposed models

Description	Data		
Optimizer	RMSProp	lr	0.001
		ρ	0.9
Activation function	Leaky ReLU		
Epoch	20		
Batch size	4		

training time and resolve dying ReLU which has a non-zero slope part. The performance of the proposed forecasting model was evaluated by RMSE that defined as follows,

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(Y_t - \hat{Y}_t)^2} \tag{3.5}$$

where Y_t means actual observed data, $\hat{Y}_t$ means forecasted data, and N means the number of learning data.

3.4 Forecasting model

This paper proposed a forecasting model is combined the 3DCNN, CLSTM, and 2DCNN models that can be called the 3CNN-CLSTM-2CNN model. To verify the effectiveness of the proposed model was compared FC-LSTM, DCLSTM, and 3CNN-CLSTM models.

3.4.1 FC-LSTM

Fully connected LSTM (FC-LSTM) is a type of LSTM model which uses several LSTM layers was adjusted with the FC layer for easy learning temporal features [41]. FC-LSTM is powerful to learn temporal relationship that effect loss spatial information. All gates and state of FC-LSTM uses one dimensional (1D) vector [18, 23, 41]. FC-LSTM model consists of an encoder and forecaster network with LSTM units that composed of five LSTM layers respectively and the last process is used FC layer to produce forecasting results. Configuration of FC-LSTM model shown in Figure 3.4.

Figure 3.5 shows the operation of the encoder and forecaster with the LSTM unit that

Figure 3.4: Configuration of FC-LSTM model

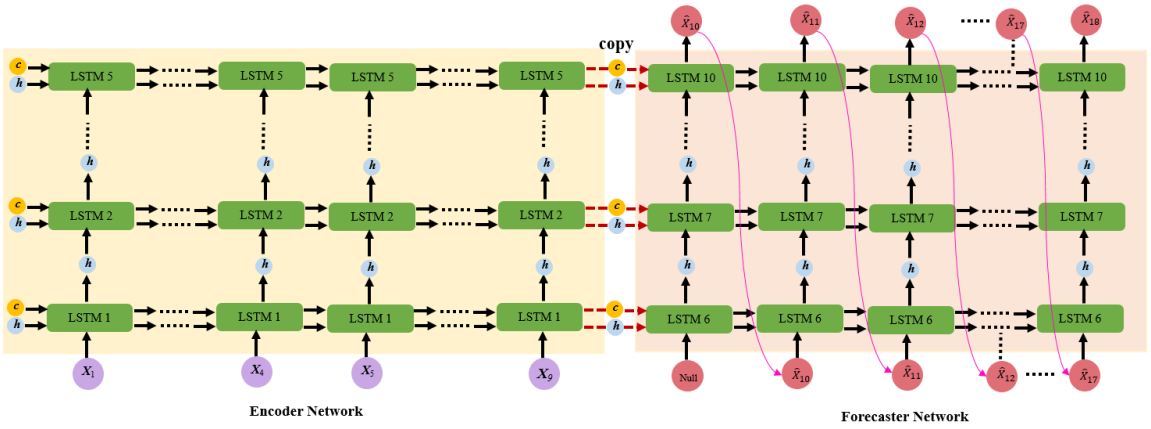


Figure 3.5: Operation of the encoder and forecaster network in FC-LSTM model

indicated the process from the input layer to the LSTM layer. Firstly, the input sequence was read in the encoder network. Subsequent to the end time-step of input was finishing read, cell state (c) and the hidden state (h) of the LSTM layer in the encoder network were taken to the cell state and the hidden state of the LSTM layer as input in the forecaster network to forecast output sequence. LSTM layers use 32 cells. The time-series of LSTM model is set to nine is the same number of time-series in DCLSTM, 3CNN-CLSTM, and 3CNN-CLSTM-2CNN models.

3.4.2 DCLSTM

CLSTM model combines CNN and LSTM model to improve LSTM model and to resolve image sequence. CLSTM can be learning spatio-temporal features and learning process sequence to sequence images. This section is used deep convolutional long short-term memory (DCLSTM) model that developed the CLSTM model. Input and output images of DCLSTM model are five dimensions: the number of sample (N), time-series (ts), height (H), width (W), and channel (C). DCLSTM model has five CLSTM layers as shown in Figure 3.6.

DCLSTM model is composed of an encoder and forecaster network. The encoder network consists of a Conv layer and five CLSTM layers. The forecaster network consists of

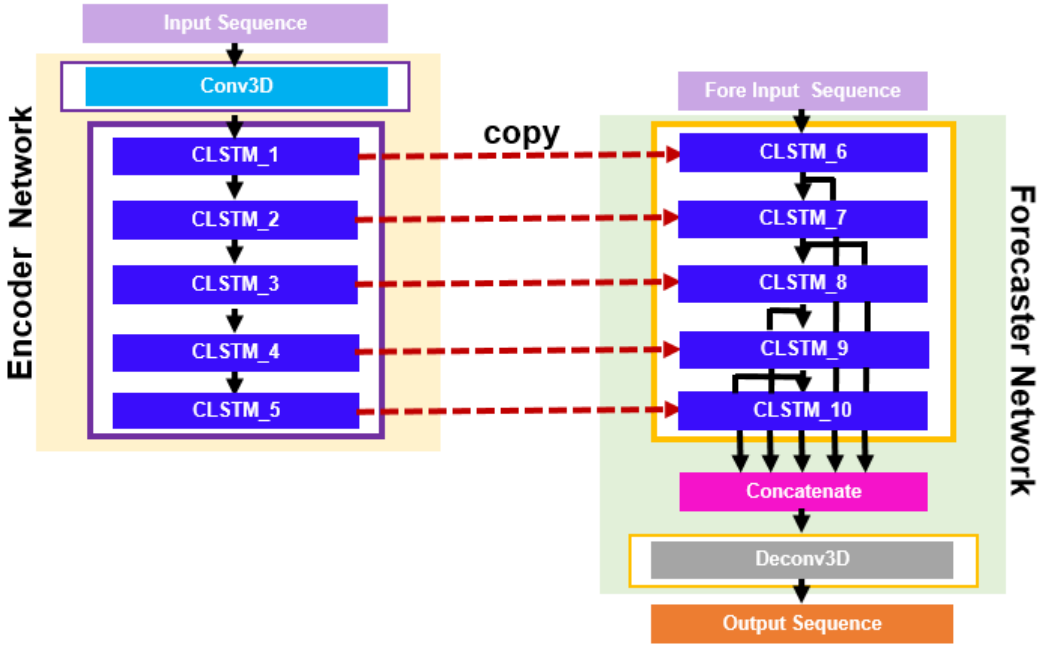


Figure 3.6: Configuration of DCLSTM model

five CLSTM layers and a Deconv layer. The stride in Conv layer and Deconv layer are set to two. The kernel size of Conv layer, Deconv layer, CLSTM layer use five. The channel size of the Conv layer is 16. The channel size of the CLSTM layers are 32. Fore inputs use zero arrays in the forecaster network. The last state of the CLSTM layer in the encoder network is copied to the state of the CLSTM layer in the forecaster network. The size of input and output images in DCLSTM are the same. In the forecaster network, all states of CLSTM concatenate become the input of the Deconv layer that uses channel size one to get the forecasting result.

3.4.3 3CNN-CLSTM

3CNN-CLSTM model combines 3DCNN and DCLSTM models. 3CNN-CLSTM model can improve DCLSTM that combines the advantage of CNN and CLSTM: decrease training time, easier for training, improve forecasting accuracy [5]. Cause 3CNN-CLSTM model can easier for training: the height and weight of feature maps are downsampled use max-pooling layer till can easy for training in CLSTM layer and to get the same size of output and input images uses up-sampling layer. Input and output images of 3CNN-CLSTM model are five dimensions: N, ts, H, W, and C.

Configuration of 3CNN-CLSTM model as shown in Figure 3.7 and composed of en-

coder network and forecaster network. The encoder network is composed of three Conv layers, three max-pooling layers, and five CLSTM layers which H and W of feature maps can be narrowed using the max-pooling layer. The forecaster network is composed of three Deconv layers, three up-sampling layers, and five CLSTM layers which H and W of feature maps can be enlarged using the up-sampling layer. The channel size of the Conv layers are 16, 16, 32 respectively. The channel size of the Deconv layers are 32, 16, 1 respectively. The channel size of the CLSTM layers are 32. The kernel size of CLSTM, Conv, Deconv layers are set to five.

In the forecaster network, fore input uses zero arrays. In the encoder network, the last state of the CLSTM layer was copied to the CLSTM layer in the forecaster network. In the forecaster network, all states CLSTM layers concatenate to input of Deconv layer and the last Deconv layer uses one for getting forecasting results.

3.4.4 3CNN-CLSTM-2CNN

3CNN-CLSTM-2CNN model combines 3DCNN, DCLSTM, and 2DCNN models. 3CNN-CLSTM-2CNN model can improve DCLSTM and 3CNN-CLSTM models that combine the advantage of 3DCNN, DCLSTM, and 2DCNN: decrease training time, easier for training, improve forecasting accuracy. Cause 3CNN-CLSTM-2CNN model can easier for training than DCLSTM and 3CNN-CLSTM: the process of enlarged size of output image uses 2D that four dimensions (N, H, W, and C). Input image of 3CNN-CLSTM-2CNN model is five dimensions: N, ts, H, W, and C. Configuration of 3CNN-CLSTM-2CNN model as shown in Figure 3.8. Difference 3CNN-CLSTM and 3CNN-CLSTM-2CNN models is Deconv and up-sampling layers in 3CNN-CLSTM model uses 3D while 3CNN-CLSTM-2CNN model uses 2D.

3CNN-CLSTM-2CNN model consists of encoder network and forecaster network. The encoder network is composed of three Conv layers, three max-pooling layers, and five CLSTM layers which H and W of feature maps downsampled using the max-pooling layer and Conv layer. The forecaster network is composed of three Deconv layers, three up-sampling layers, and five CLSTM layers which H and W of feature maps upsampled using the up-sampling layer and Deconv layer. The channel size of the Conv layers are 16, 16, 32 respectively. The channel size of the Deconv layers are 32, 16, 1 respectively. The channel size of the CLSTM layers are 32. The kernel size of CLSTM, Conv, Deconv layers are set to five.

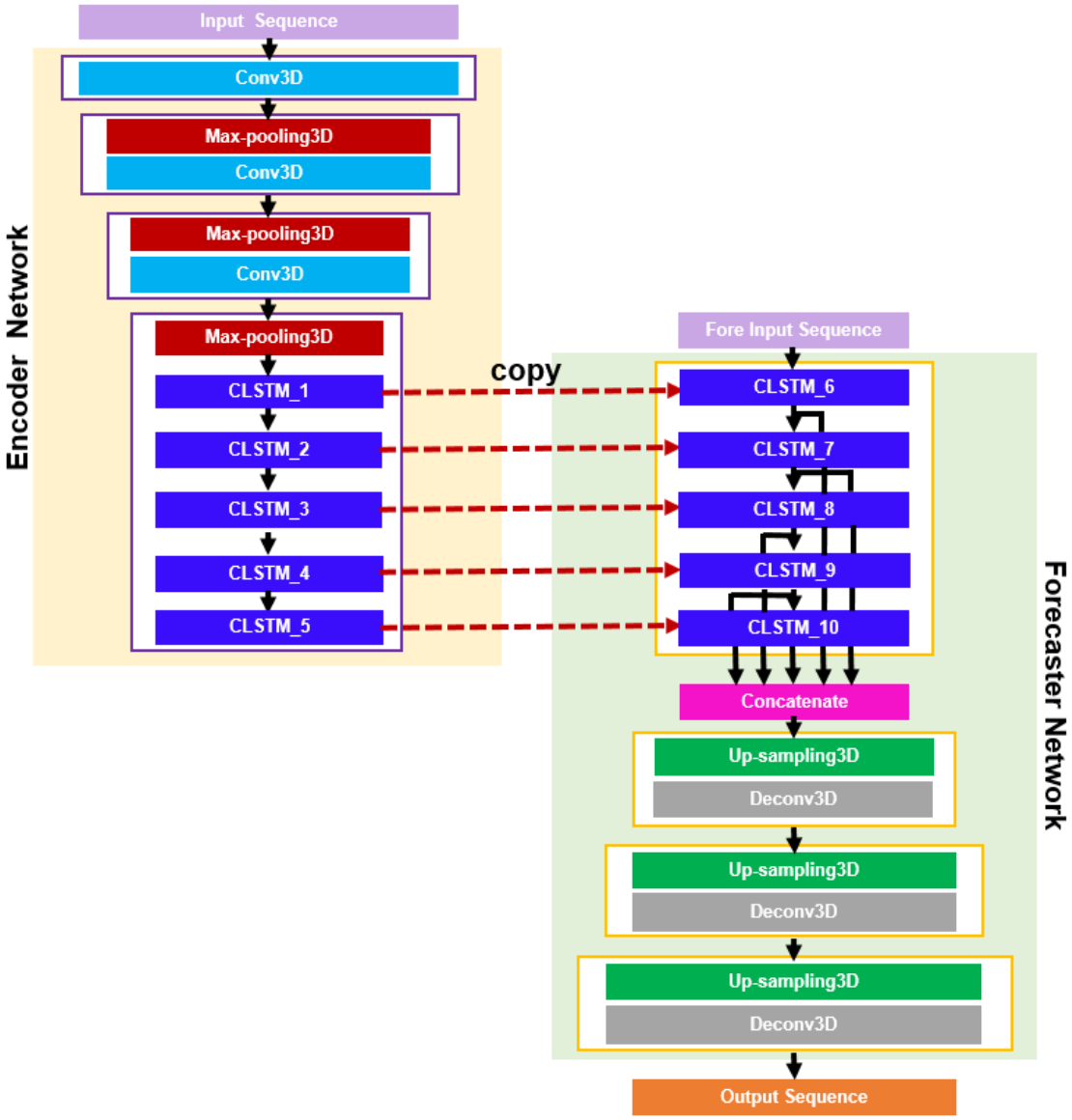


Figure 3.7: Configuration of 3CNN-CLSTM model

Fore input uses zero arrays in the forecaster network. The last state of the CLSTM layer in encoder network layer was copied to the CLSTM layer in the forecaster network. all states CLSTM layers in the forecaster network concatenate as the input of Deconv layer and the last Deconv layer uses one for getting forecasting results.

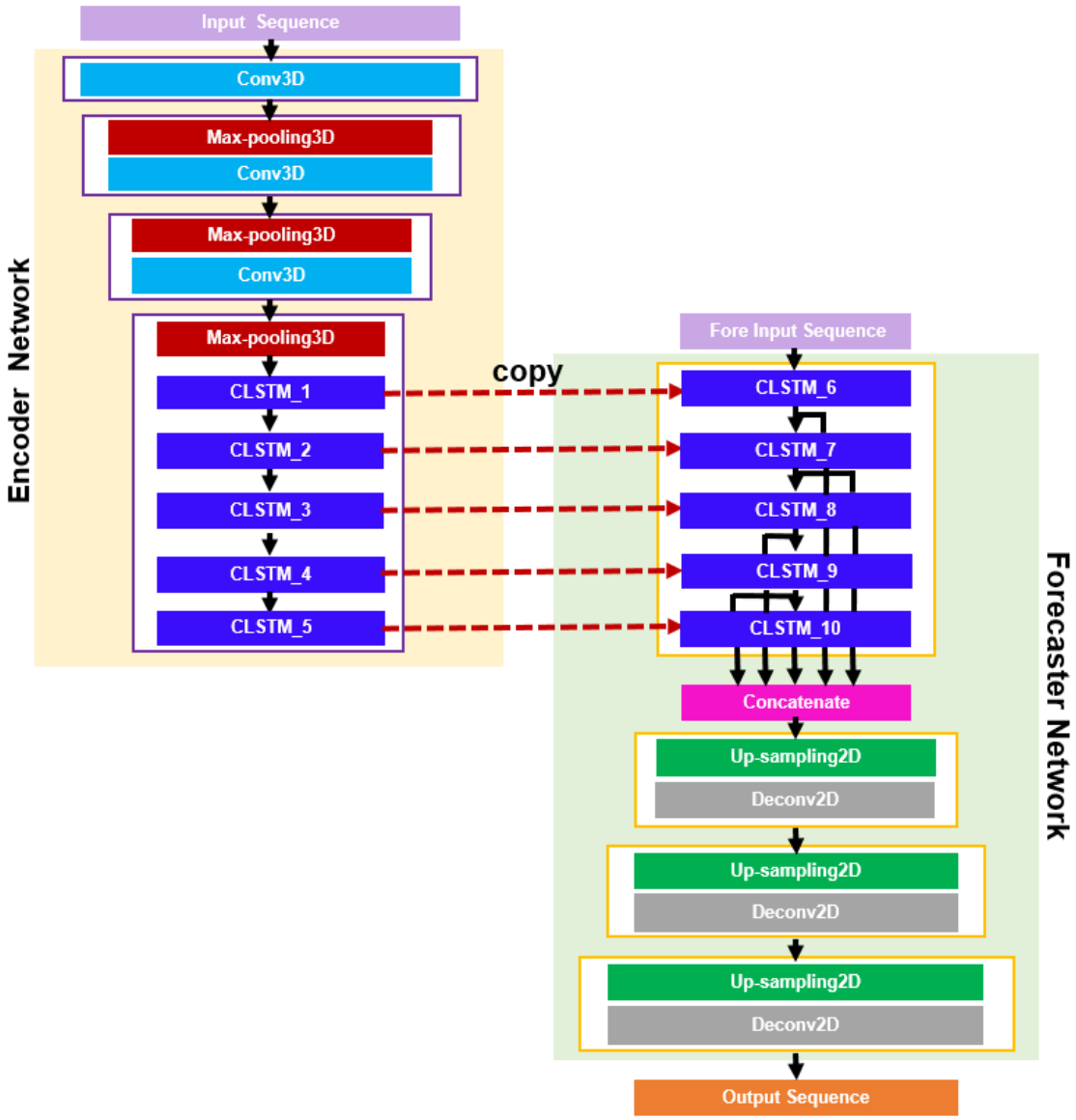


Figure 3.8: Configuration of 3CNN-CLSTM-2CNN model

CHAPTER 4

Forecasting Accuracy of Wind Speed and Direction

This chapter explains the forecasting results of wind speed and direction and also discusses the efficiency of the proposed forecasting model in comparison to other models. Section 4.1 explains the mathematical model to get the forecasting results. Forecasting results of $v_X(t)$ and $v_Y(t)$ are explained in section 4.2. Then, section 4.3 explains the forecasting results of wind speed and direction. In this chapter, the forecasting point in Tokushima city, Japan.

4.1 Mathematical model to get forecasting results

Forecasting results of wind speed and direction are taken from a forecasting image that calculates a center of gravity (CoG) the largest pixel cluster with a value of zero pixels. The forecasting position $\hat{p}_X(t)$ and $\hat{p}_Y(t)$ are converted to $\hat{v}_X(t)$ and $\hat{v}_Y(t)$ by substracting center position of an image (64, 64) as defined by,

$$\hat{v}_X(t) = \frac{v_{max}}{64}(\hat{p}_X(t) - 64) \tag{4.1}$$

$$\hat{v}_Y(t) = \frac{v_{max}}{64}(64 - \hat{p}_Y(t)) \tag{4.2}$$

where v_{max} is the maximum of wind speed, $\hat{v}_X(t)$ and $\hat{v}_Y(t)$ are wind vector of $\hat{v}(t)$ at X and Y components. In this chapter, the forecasting model is used in Tokushima city and the value of v_{max} is 20 m/s. The forecasting results of wind speed and direction are calculated from $\hat{v}_X(t)$ and $\hat{v}_Y(t)$ is given by,

$$\hat{v}(t) = \sqrt{\hat{v}_X^2(t) + \hat{v}_Y^2(t)} \tag{4.3}$$

Table 4.1: RMSE of v_X for each month

Description	RMSE [m/s]			
	FC-LSTM	DCLSTM	3CNN-CLSTM	3CNN-CLSTM-2CNN
Dec.	1.5719	1.2276	1.0840	**1.0559**
Jan.	1.4376	1.1949	1.0849	**1.0094**
Feb.	1.4329	1.3186	1.2217	**1.1490**
Mar.	1.5855	1.3864	1.3274	**1.2487**
Apr.	1.5400	1.3319	1.1749	**1.0887**
May	1.4498	1.3184	1.1365	**1.0776**
Jun.	1.4647	1.3623	1.2022	**1.1014**
Jul.	1.3852	1.3732	1.1503	**1.1066**
Aug.	1.7351	1.3056	1.1450	**1.0750**
Sep.	1.4489	1.3168	1.1555	**1.0577**
Oct.	1.4131	1.2287	1.0888	**1.0201**
Nov.	1.3662	1.1535	1.0637	**0.9924**
Total	1.4900	1.2950	1.1547	**1.0836**

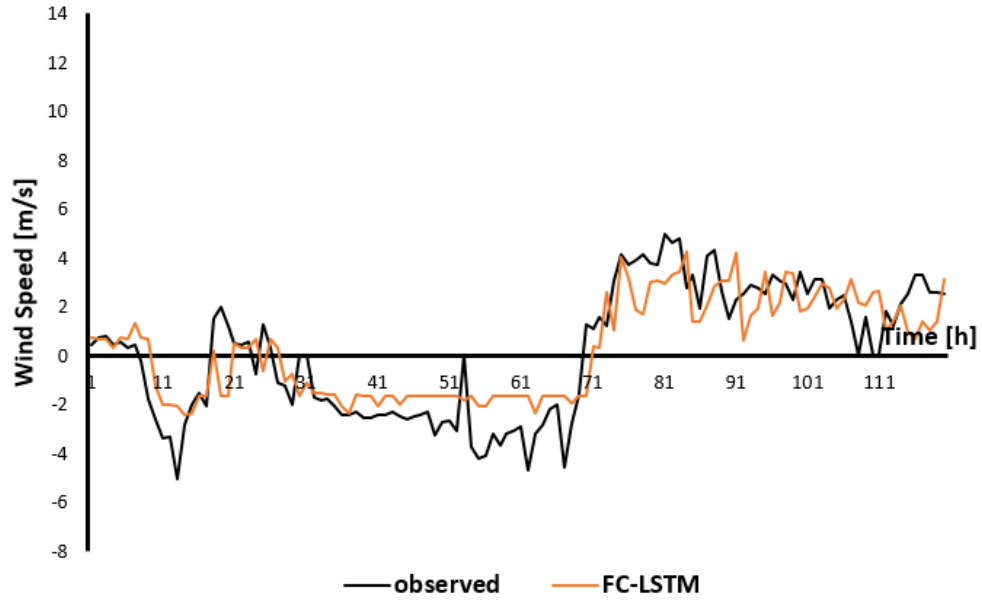
$$\hat{\varphi}(t) = \tan^{-1}\left(\frac{\hat{v}_Y(t)}{\hat{v}_X(t)}\right) \tag{4.4}$$

where $\hat{v}(t)$ means forecasting wind speed [m/s], and $\hat{\varphi}(t)$ means forecasting wind direction [°].

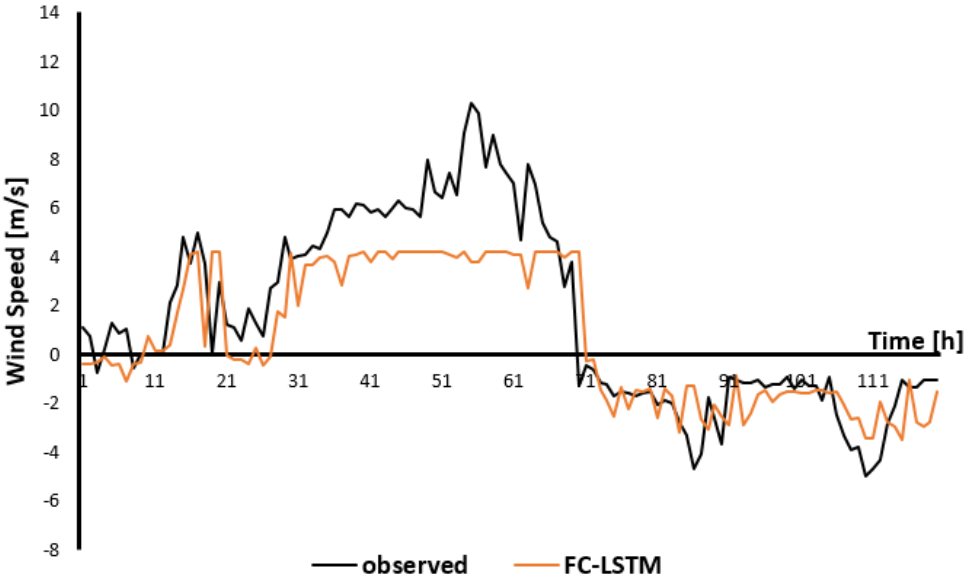
4.2 Forecasting results of v_X and v_Y

The target forecasting results are short-term time that one hour ahead and evaluated by RMSE between actual observed data and forecasted data. Before this research analyzes wind speed and direction, it analyzes forecasting results of v_X and v_Y to all models. The forecasting result of $v_X(t)$ and $v_Y(t)$ are obtained Eqs. 4.1 and 4.2. Figures 4.1 - 4.4 show the forecasted and observed v_X and v_Y for five days on Oct., 1-5, 2019. The delay in FC-LSTM model can reduced effectively uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models. The forecasting result of v_X and v_Y by 3CNN-CLSTM-2CNN is approaching observed data rather than other models that it confirms the best forecasting model.

Forecasting error (RMSE) v_X and v_Y of each forecasting model from Dec. 2018 – Nov. 2019 is shown in Table 4.1 and 4.2.

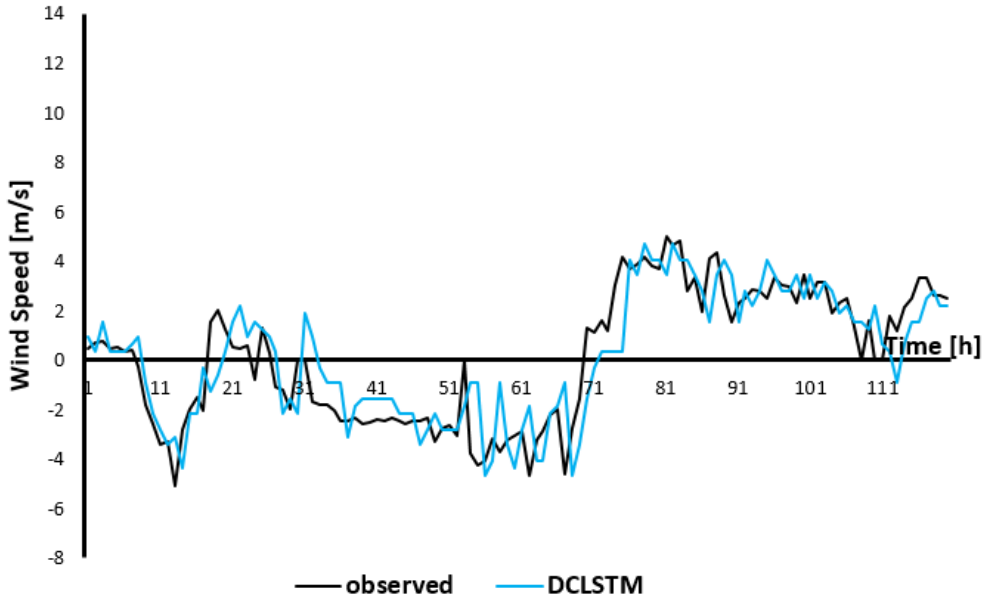


(a) v_X

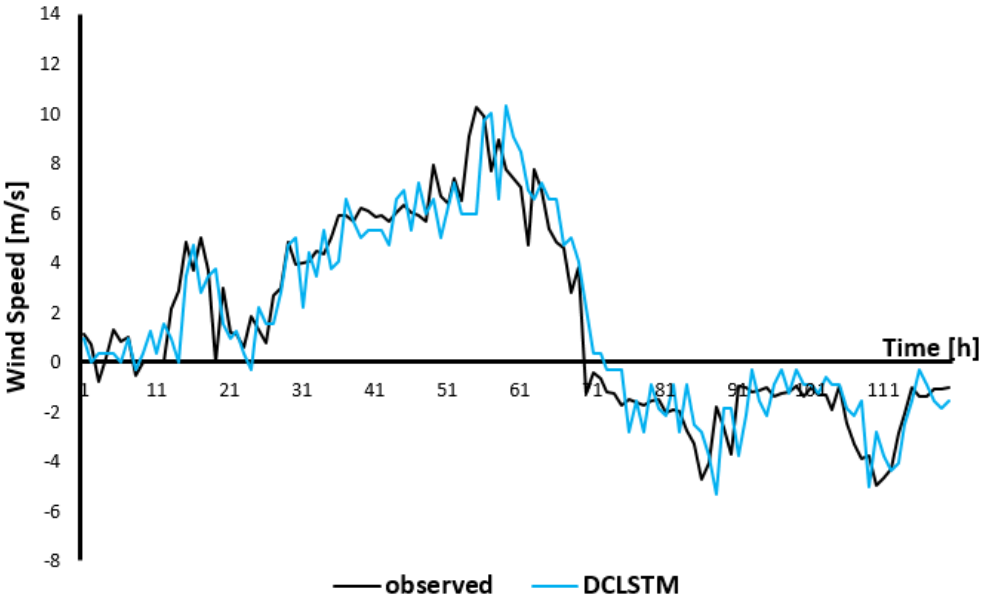


(b) v_Y

Figure 4.1: Forecasting result of $v_X(t)$ and $v_Y(t)$ of FC-LSTM model

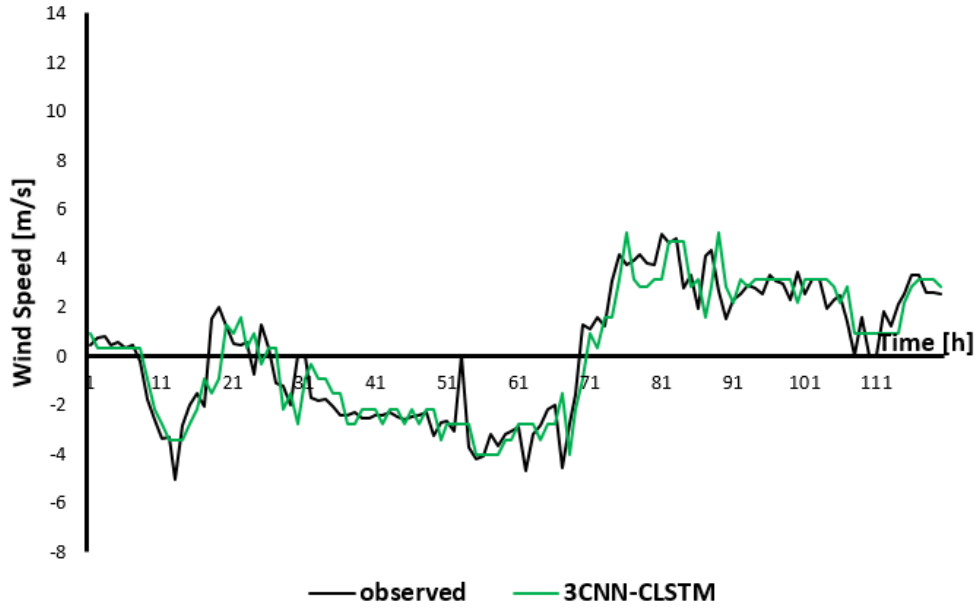


(a) v_X



(b) v_Y

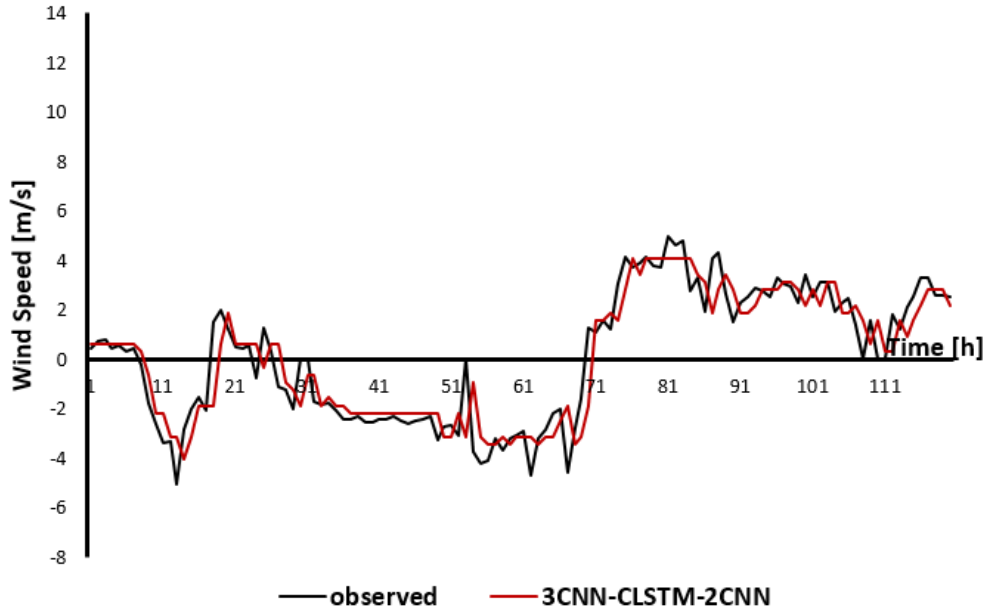
Figure 4.2: Forecasting result of $v_X(t)$ and $v_Y(t)$ of DCLSTM model



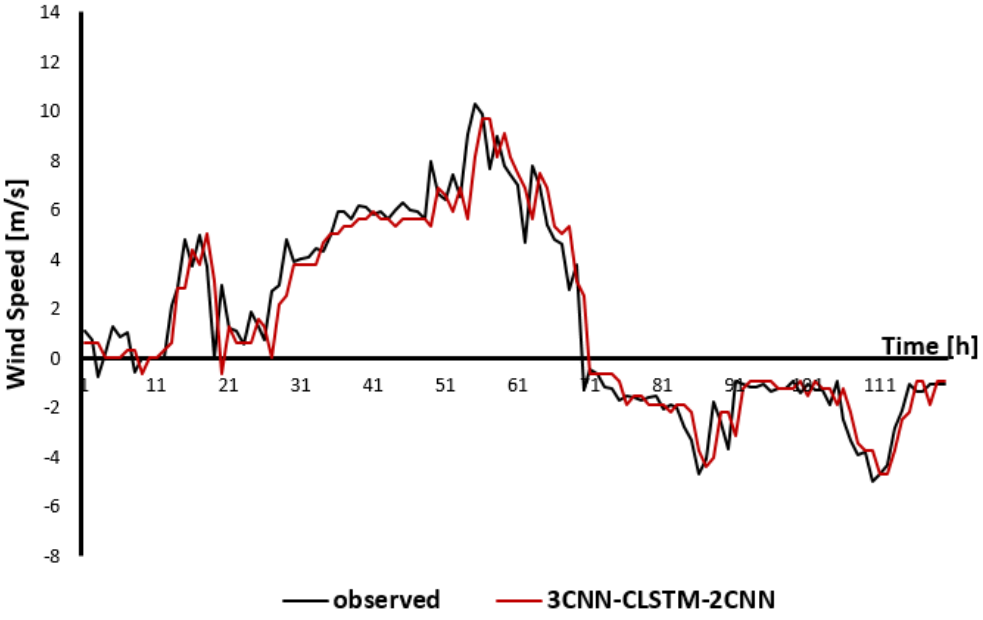
(a) v_X

(b) v_Y

Figure 4.3: Forecasting result of $v_X(t)$ and $v_Y(t)$ of 3CNN-CLSTM model



(a) v_X



(b) v_Y

Figure 4.4: Forecasting result of $v_X(t)$ and $v_Y(t)$ of 3CNN-CLSTM-2CNN model

FC-LSTM model is the lowest forecasting accuracy both in v_X and v_Y. DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models can improve forecasting accuracy FC-LSTM model. 3CNN-CLSTM-2CNN model is the highest forecasting accuracy and it indicates the best forecasting model.

Table 4.2: RMSE of v_Y for each month

Description	RMSE [°]			
	FC-LSTM	DCLSTM	3CNN-CLSTM	3CNN-CLSTM-2CNN
Dec.	1.3477	1.2999	1.1065	**1.0709**
Jan.	1.3904	1.2525	1.0259	**0.9783**
Feb.	1.5440	1.2917	1.1516	**1.0684**
Mar.	1.5471	1.4776	1.2533	**1.2382**
Apr.	1.8947	1.5910	1.3222	**1.3130**
May	1.8076	1.5948	1.3519	**1.3019**
Jun.	1.5043	1.4700	1.2011	**1.1993**
Jul.	1.5095	1.4782	1.2894	**1.2520**
Aug.	1.7020	1.5751	1.4041	**1.3662**
Sep.	1.6770	1.3044	1.1995	**1.1259**
Oct.	1.5312	1.2025	0.9895	**0.9661**
Nov.	1.4805	1.2640	1.0946	**1.0056**
Total	1.5857	1.4080	1.2062	**1.1657**

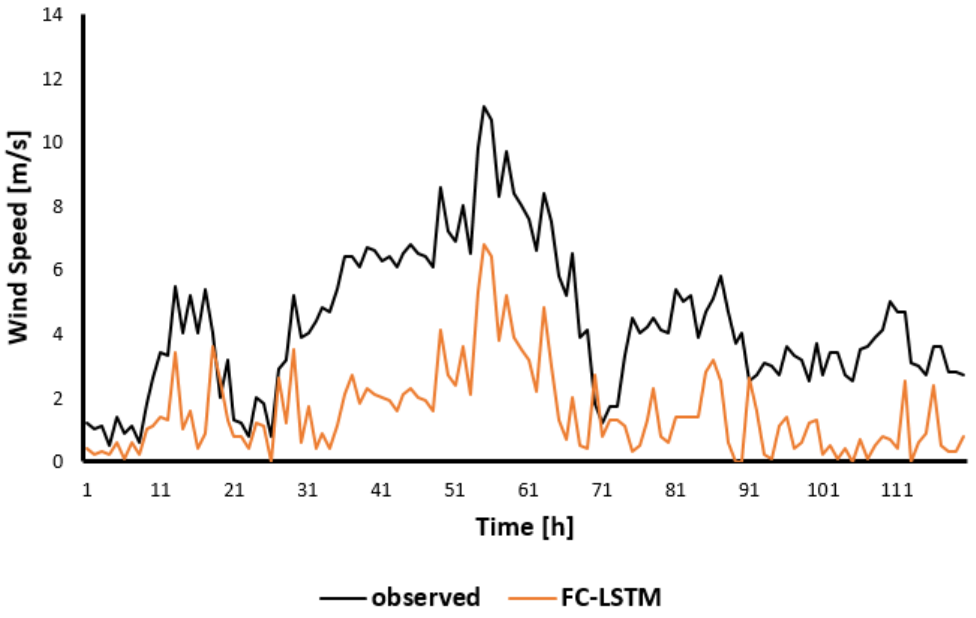
Table 4.3 shows the improvement rate of DCLSTM and 3CNN-CLSTM, 3CNN-CLSTM-2CNN models is compared FC-LSTM model for each month (Dec. 2018 - Nov. 2019). In v_X, the minimum forecasting accuracy can improve uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models are 0.87%, 14.74%, 19.81% severally. In v_Y, the minimum forecasting accuracy can improve uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models are 2.07%, 14.58%, 17.06% severally. The maximum forecasting accuracy in v_X can improve uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models are 24.75%, 34.01%, 38.04% severally. The maximum forecasting accuracy in v_Y can improve uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models are 22.22%, 35.38%, 36.91% severally. 3CNN-CLSTM-2CNN model can improve forecasting accuracy than DCLSTM and 3CNN-CLSTM models effectively both of v_X and v_Y which is denote benchmark forecasting model. The end of Table 4.6 shows a total improvement rate of DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models which 3CNN-CLSTM-2CNN model can improve forecasting accuracy more than 10% from DCLSTM model and more than 2% from the CNN-CLSTM model both in v_X and v_Y.

Table 4.3: Improvement rate of v_X and v_Y

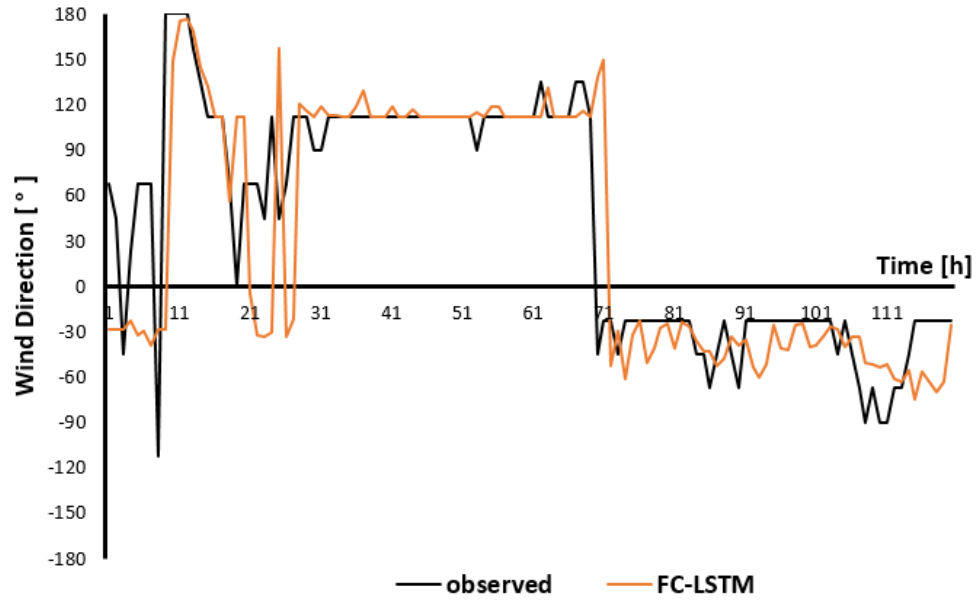
Description	Improvement rate [%]					
	v_X			v_Y		
	DCLSTM	3CNN-CLSTM	3CNN-CLSTM 2CNN	DCLSTM	3CNN-CLSTM	3CNN-CLSTM 2CNN
Dec.	21.90	31.04	32.83	3.55	17.90	20.54
Jan.	16.88	24.53	29.79	9.92	26.22	29.64
Feb.	7.98	14.74	19.81	16.34	25.41	30.80
Mar.	12.56	16.28	21.24	4.49	18.99	19.97
Apr.	13.51	23.71	29.31	16.03	30.22	30.70
May	9.06	21.61	25.67	11.77	25.21	27.98
Jun.	6.99	17.92	24.80	2.28	20.16	20.28
Jul.	0.87	16.96	20.11	2.07	14.58	17.06
Aug.	24.75	34.01	38.04	7.46	17.50	19.73
Sep.	9.12	20.25	27.00	22.22	28.47	32.86
Oct.	13.05	22.95	27.81	21.47	35.38	36.91
Nov.	15.57	22.14	27.36	14.62	26.07	32.08
Total	13.09	22.50	27.28	11.21	23.93	26.49

4.3 Forecasting results of wind speed and direction

The forecasting results of wind speed and direction are obtained Eqs. 4.3 and 4.4. The forecasted and observed wind speed and direction for five days on Oct., 1-5, 2019 as shown in Figures 4.5 - 4.8. The direction 180° = (-180)° that is graph of wind direction in Figures 4.5 (b) - 4.8 (b). In Figure 4.5, FC-LSTM model has some delay both of wind speed and direction with rapid change. The delay in FC-LSTM model can reduced effectively uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models. It confirmed DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models can extract spatio-temporal feature maps of wind vector efficiently. The forecasting result of wind speed and direction by 3CNN-CLSTM-2CNN model is approaching observed data rather than other models that confirm the best forecasting model.

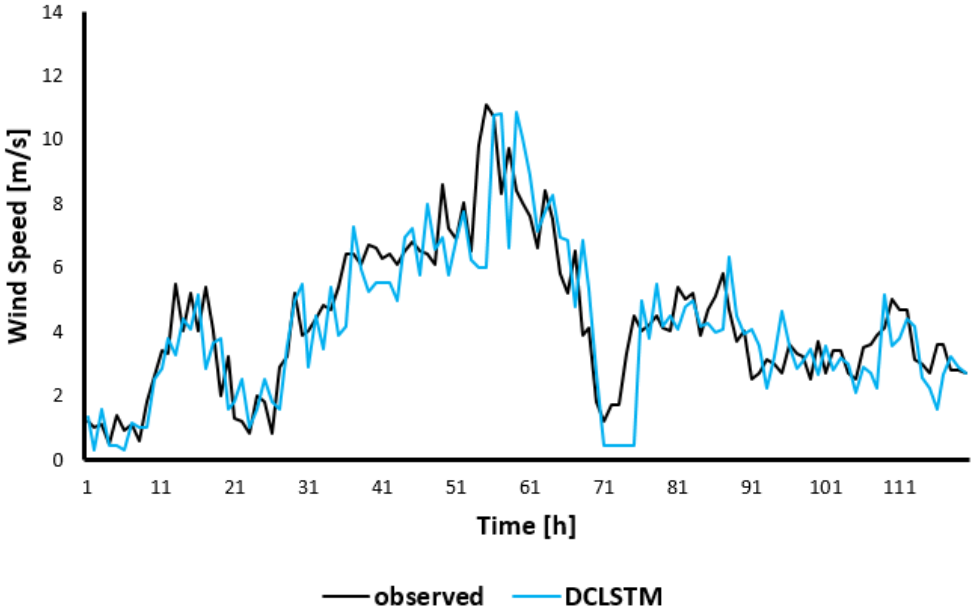


(a) Wind speed

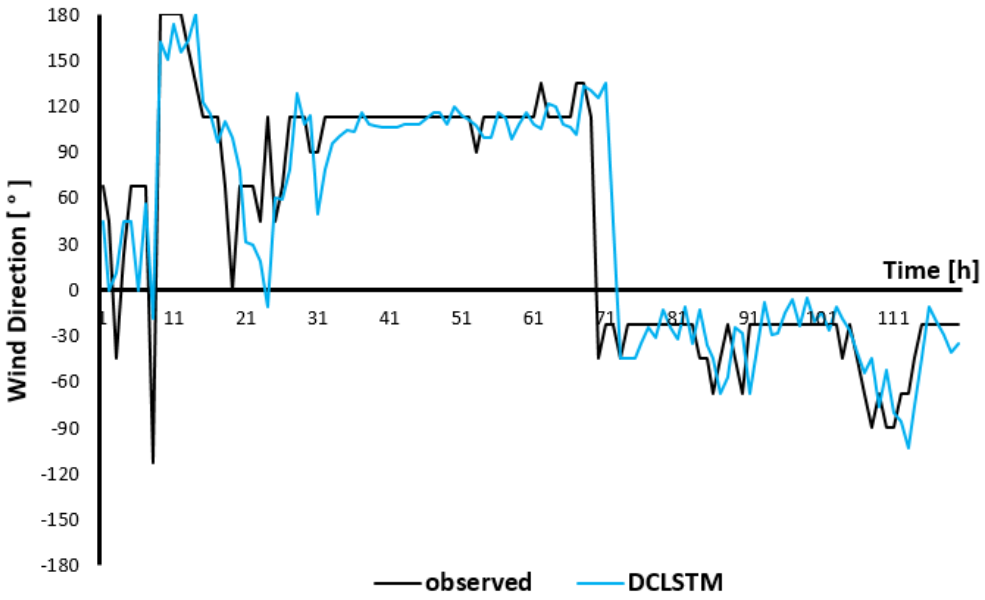


(b) Wind direction

Figure 4.5: Forecasting result of wind speed and direction of FC-LSTM model

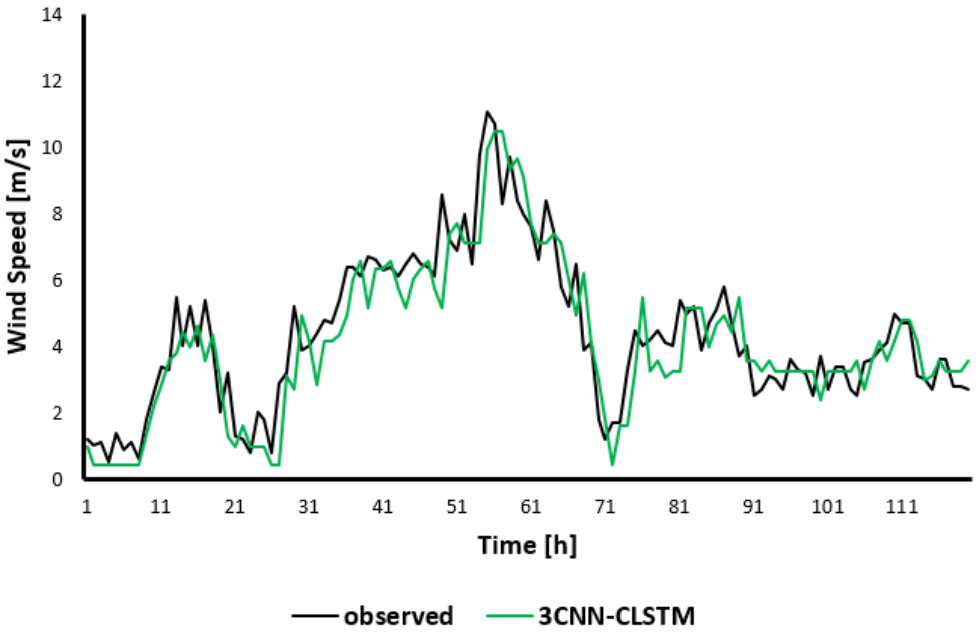


(a) Wind speed

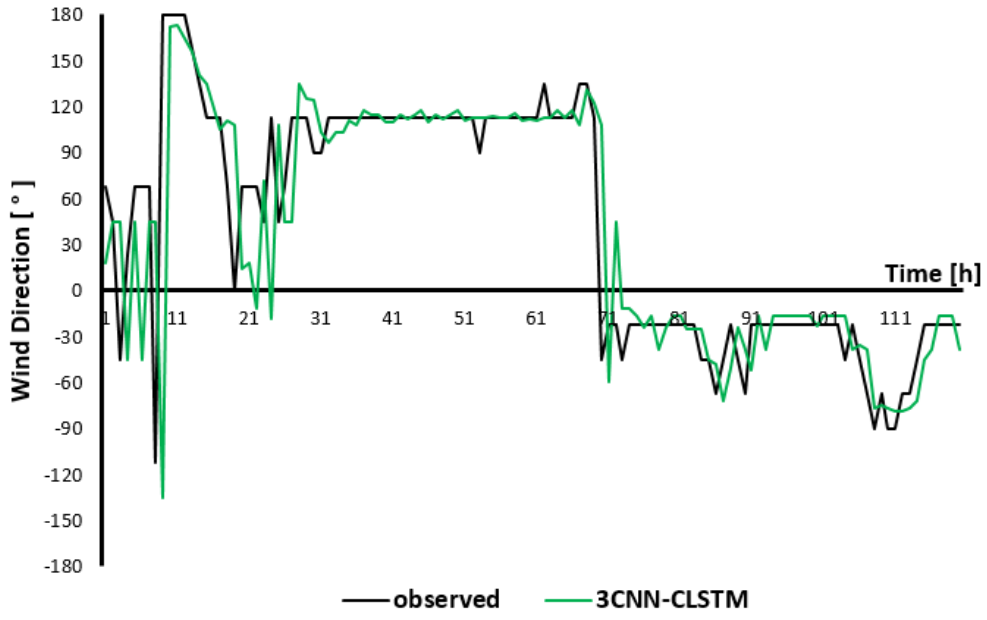


(b) Wind direction

Figure 4.6: Forecasting result of wind speed and direction of DCLSTM model

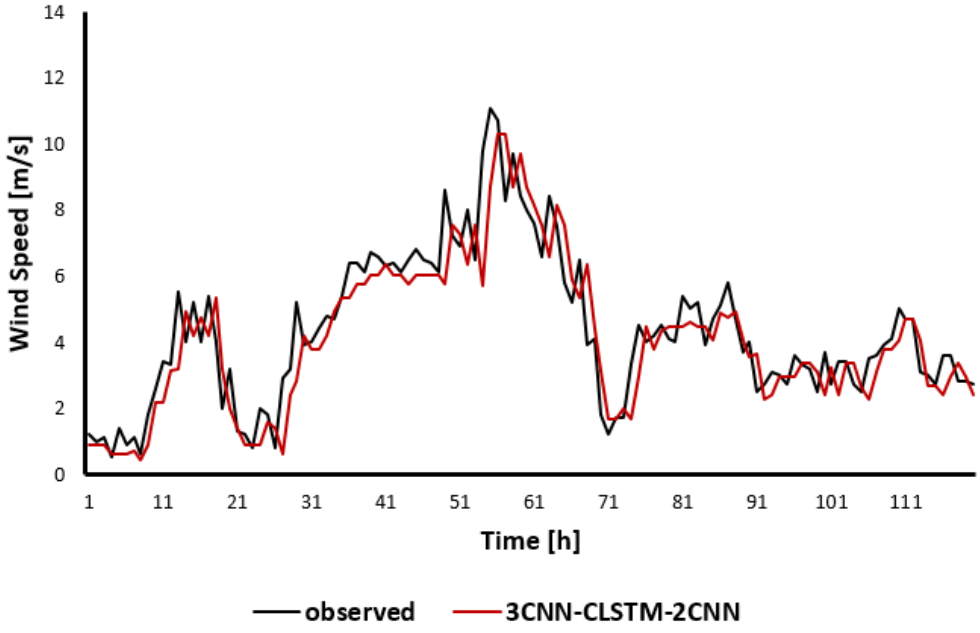


(a) Wind speed

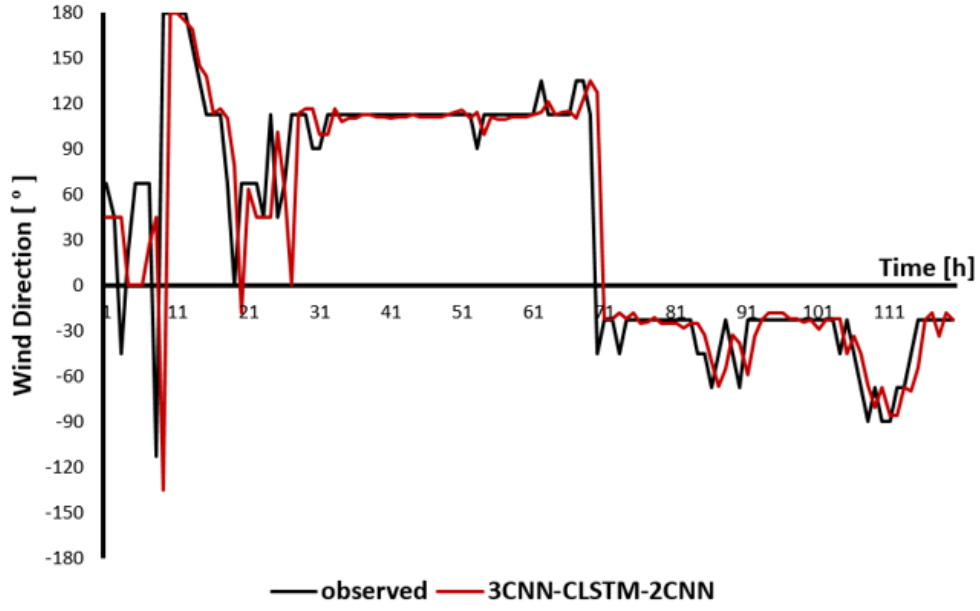


(b) Wind direction

Figure 4.7: Forecasting result of wind speed and direction of 3CNN-CLSTM model



(a) Wind speed



(b) Wind direction

Figure 4.8: Forecasting result of wind speed and direction of 3CNN-CLSTM-2CNN model

Forecasting error (RMSE) wind speed and direction of each forecasting model from Dec. 2018 – Nov. 2019 as shown in Table 4.4 and 4.5. The lowest forecasted accuracy both in wind speed and direction is the FC-LSTM model. DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models can improve forecasting accuracy FC-LSTM model. The highest

Table 4.4: RMSE of wind speed for each month

Description	RMSE [m/s]			
	FC-LSTM	DCLSTM	3CNN-CLSTM	3CNN-CLSTM-2CNN
Dec.	1.5711	1.1757	1.0250	**1.0000**
Jan.	1.4709	1.1174	0.9937	**0.9614**
Feb.	1.5818	1.2080	1.0793	**1.0621**
Mar.	1.7188	1.3588	1.2529	**1.1866**
Apr.	1.9013	1.3054	1.1046	**1.0859**
May	1.7774	1.3210	1.1040	**1.0734**
Jun.	1.5231	1.2369	1.0496	**0.9943**
Jul.	1.4555	1.1673	1.0437	**0.9884**
Aug.	1.8305	1.2110	1.0733	**1.0166**
Sep.	1.6754	1.1463	1.0604	**0.9749**
Oct.	1.5668	1.1071	0.9011	**0.8989**
Nov.	1.4473	1.1302	0.9865	**0.9445**
Total	1.6336	1.2098	1.0592	**1.0180**

forecasting accuracy is 3CNN-CLSTM-2CNN model and it indicates the best forecasting model.

Table 4.6 shows the improvement rate of DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models by FC-LSTM model for each month (Dec. 2018 - Nov. 2019). The minimum forecasting accuracy in wind speed can improve uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models are 18.79%, 27.11%, 30.96% successively. The minimum forecasting accuracy in wind direction can improve uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models are 6.42%, 12.85%, 15.22% successively. In wind speed, the maximum forecasting accuracy can improve uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models are 33.84%, 42.49%, 44.46% successively. In wind direction, the maximum forecasting accuracy can improve uses DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models are 22.77%, 27.65%, 32.89% successively. 3CNN-CLSTM-2CNN model can improve forecasting accuracy than DCLSTM and 3CNN-CLSTM models effectively both of wind speed and direction which is denote benchmark forecasting model. The end of Table 4.6 shows a total improvement rate of DCLSTM, 3CNN-CLSTM, 3CNN-CLSTM-2CNN models which 3CNN-CLSTM-2CNN model can improve forecasting accuracy more than 10% from DCLSTM model and more than 2% from the 3CNN-CLSTM model both in wind

Table 4.5: RMSE of wind direction for each month

Description	RMSE [°]			
	FC-LSTM	DCLSTM	3CNN-CLSTM	3CNN-CLSTM-2CNN
Dec.	45.4026	42.4878	38.6295	**38.4926**
Jan.	48.1570	39.8971	34.8401	**33.1539**
Feb.	48.3768	44.2642	42.1610	**39.2633**
Mar.	51.8657	47.2563	45.0555	**43.9047**
Apr.	58.2948	52.5143	45.7598	**43.7169**
May	55.9057	51.4154	47.4950	**44.6378**
Jun.	61.8276	55.3180	51.0181	**49.6847**
Jul.	69.5587	62.2201	56.1623	**54.8614**
Aug.	70.8288	65.1014	61.4489	**58.9434**
Sep.	55.6669	42.9920	40.2862	**37.3594**
Oct.	42.7311	38.9775	35.6747	**33.1543**
Nov.	46.9310	40.0034	36.6338	**34.4474**
Total	55.3933	49.3297	45.3598	**43.4435**

speed and direction.

Table 4.6: Improvement rate of wind speed and direction

Description	Improvement rate [%]					
	Wind speed			Wind direction		
	DCLSTM	3CNN-CLSTM	3CNN-CLSTM 2CNN	DCLSTM	3CNN-CLSTM	3CNN-CLSTM 2CNN
Dec.	25.17	34.76	36.35	6.42	14.92	15.22
Jan.	24.03	32.44	34.64	17.15	27.65	31.15
Feb.	23.63	31.77	32.85	8.50	12.85	18.84
Mar.	20.94	27.11	30.96	8.89	13.13	15.35
Apr.	31.34	41.90	42.89	9.92	21.50	25.01
May	25.68	37.89	39.61	8.03	15.04	20.16
Jun.	18.79	31.09	34.72	10.53	17.48	19.64
Jul.	19.80	28.29	32.09	10.55	19.26	21.13
Aug.	33.84	41.37	44.46	8.09	13.24	16.78
Sep.	31.58	36.71	41.81	22.77	27.63	32.89
Oct.	29.34	42.49	42.63	8.78	16.51	22.41
Nov.	21.91	31.84	34.74	14.76	21.94	26.60
Total	25.94	35.16	37.68	10.95	18.11	21.57

CHAPTER 5

Regional Dedendence of 3CNN-CLSTM-2CNN Model in Four Cities

This chapter describes the forecasting results of the 3CNN-CLSTM-2CNN model in four cities: Tokushima, Takamatsu, Hiketa, and Choshi. Firstly, this chapter explains about characteristics of each city which can be used to based consideration of choosing these cities. Every city has different characteristics of wind speed and direction (wind information) that affected by several factors like seasons and topography. The condition of wind information must be observed and analyze for the planning of wind power generation. The wind information is taken by AMeDAS at one hour interval. Afterward, section 5.2 presents forecasting results of v_X and v_Y in four cities. The end of chapter presents forecasting results of wind speed and direction in four cities.

5.1 Wind speed and direction condition in four cities

Tokushima city is located in Tokushima prefecture, Takamatsu and Hiketa cities are located in Kagawa prefecture, and Choshi city is located in Chiba prefecture. These cities are good wind positions due to near onshore, near the sea, and near wind power generation. This research used four years of data by AMeDAS from Dec 2015 - Nov 2019. So that, the annual wind condition of four cities are showed from 2015 - 2019. Figure 5.1 shows the annual average wind speed in four cities (2015 - 2019). From Figure 5.1, the high value of average wind speed is Choshi city, the low value of average wind speed is Takamatsu city, and in each city, every year the value of the average wind speed is slightly change.

The maximum wind speed of each city is shown in Figure 5.2. The maximum wind speed is used to specify the value of v_{max} in Eqs. 3.3 and 3.4. The value of v_{max} is needed

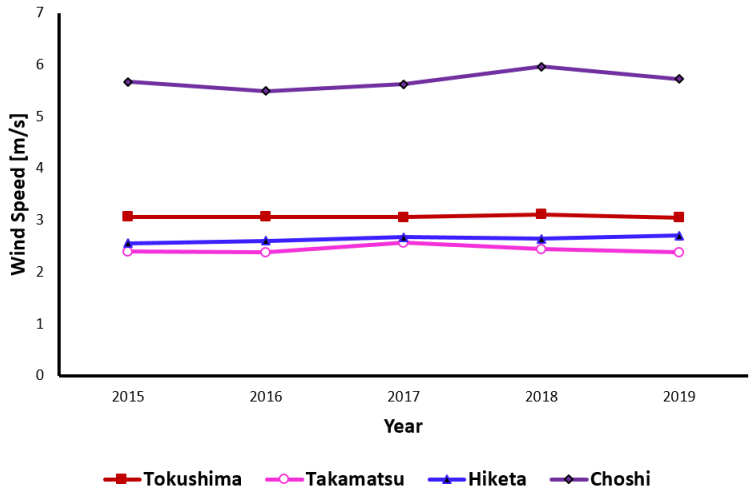


Figure 5.1: Annual average wind speed in four cities (2015 - 2019)

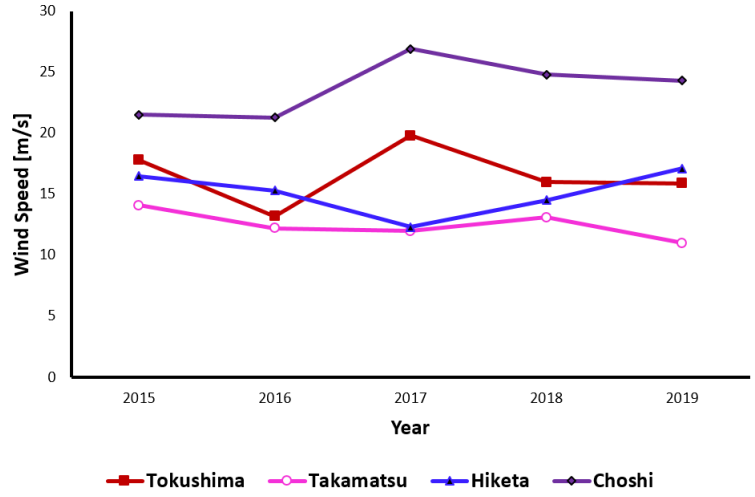


Figure 5.2: Maximum wind speed in four cities

to determine the wind vector in each city. From Figure 5.2, the value of v_{max} in Tokushima, Takamatsu, and Hiketa are 20 m/s and the value of v_{max} in Choshi city is 30 m/s.

The test data is used for one year from Dec. 2018 - Nov. 2019 (see Table 3.1). Monthly average wind speed in four cities based on the test data which is Dec 2018 - Nov. 2019 as shown in Figure 5.3. Monthly average wind speed in Tokushima, Takamatsu, and Hiketa cities slightly change but the monthly average wind speed in Choshi city mostly significantly change.

The annual data and fluctuation of wind speed and direction in four cities obtained from the value of wind speed and direction one year that Dec. 2018 - Nov 2019. This wind information is needed to analyze the forecasting result in four cities. The value of fluctuation

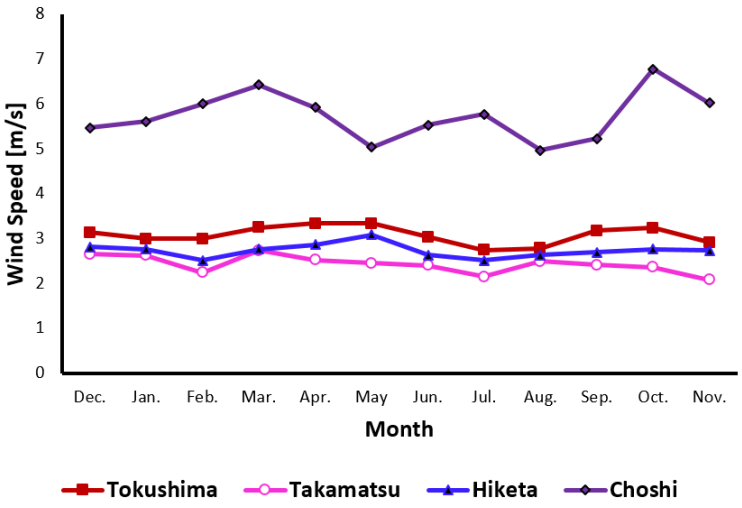


Figure 5.3: Monthly average wind speed in four cities (Dec. 2018 - Nov. 2019)

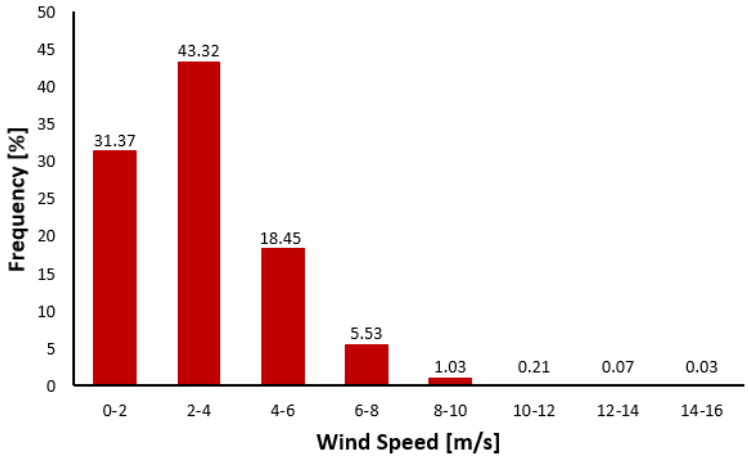
of wind speed and direction obtained from subtracting between present data and past data. The meaning negative value both in wind speed and direction that the past data is higher than the present data.

The percentage of annual wind speed and direction in Tokushima city is shown in Figure 5.4. From Figure 5.4 (a), the maximum value for annual wind speed ranges from 14 - 16 m/s and the high percentage for annual wind speed ranges 2 - 4 m/s. From Figure 5.4 (b), the high percentage for annual wind direction is 337.5° which the direction is east-south-east (ESE).

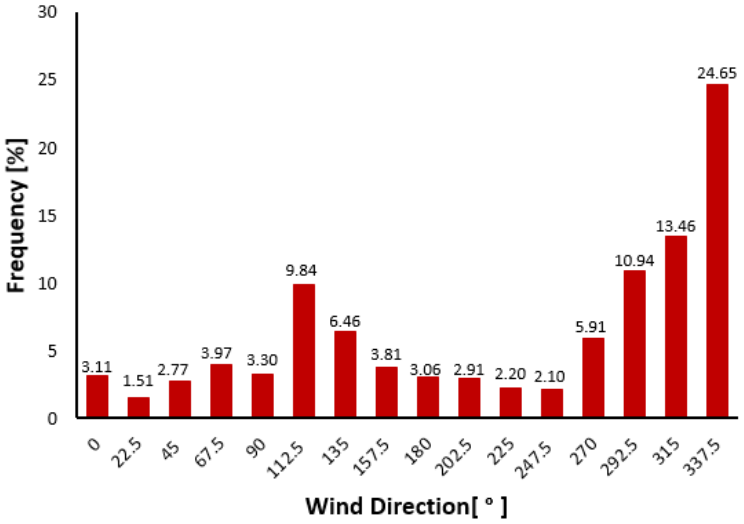
The percentage of fluctuation of wind speed and direction in Tokushima city is shown in Figures 5.5. From Figure 5.5 (a), the maximum value for fluctuation of wind speed ranges from 6 - 8 m/s, and the high percentage for fluctuation of wind speed ranges 0 - 2 m/s. From Figure 5.5 (b), the high percentage for fluctuation of wind direction is 0°.

The percentage of annual wind speed and direction in Takamatsu city is shown in Figure 5.6. From Figure 5.6 (a), the maximum value for annual wind speed ranges from 10 - 12 m/s and the high percentage for annual wind speed ranges 0 - 2 m/s. From Figure 5.6 (b), the high percentage for annual wind direction is 22.5° which the direction is east-north-east (ENE).

The percentage of fluctuation of wind speed and direction in Takamatsu city is shown in Figures 5.7. From Figure 5.7 (a), the maximum value for fluctuation of wind speed ranges from 6 - 8 m/s, and the high percentage for fluctuation of wind speed ranges 0 - 2 m/s. From Figure 5.7 (b), the high percentage for fluctuation of wind direction is 0°.



(a) Wind speed



(b) Wind direction

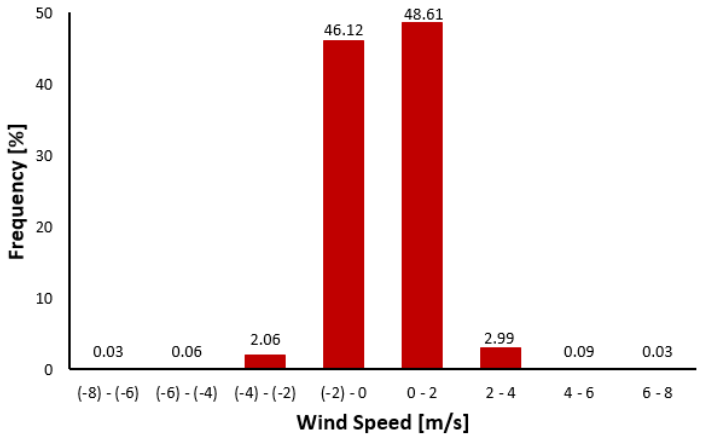
Figure 5.4: Annual wind speed and direction in Tokushima city

The percentage of annual wind speed and direction in Hiketa city is shown in Figure 5.8. From Figure 5.8 (a), the maximum value for annual wind speed ranges from 16 - 18 m/s and the high percentage for annual wind speed ranges 2 - 4 m/s. From Figure 5.8 (b), the high percentage for annual wind direction is 67.5° which the direction is north-north-east (NNE).

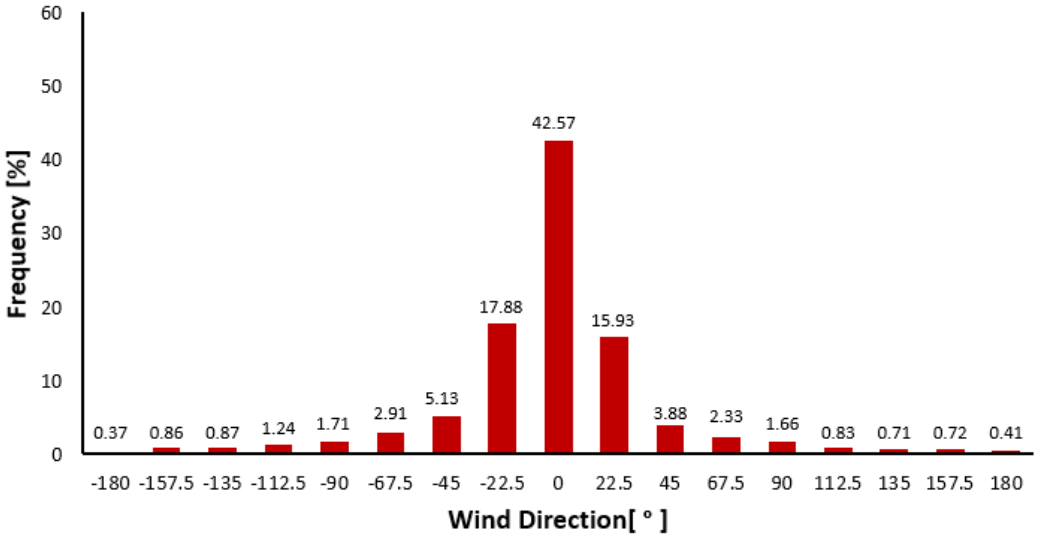
The percentage of fluctuation of wind speed and direction in Hiketa city is shown in Figures 5.9. From Figure 5.9 (a), the maximum value for fluctuation of wind speed ranges from 8 - 10 m/s, and the high percentage for fluctuation of wind speed ranges 0 - 2 m/s. From Figure 5.9 (b), the high percentage for fluctuation of wind direction is 0°.

The percentage of annual wind speed and direction in Choshi city is shown in Figure 5.10. From Figure 5.10 (a), the maximum value for annual wind speed ranges from 24 - 26 m/s and the high percentage for annual wind speed ranges 2 - 4 m/s. From Figure 5.10 (b), the high percentage for annual wind direction is 247.5° which the direction is south-south-west (SSW).

The percentage of fluctuation of wind speed and direction in Choshi city is shown in Figures 5.11. From Figure 5.11 (a), the maximum value for fluctuation of wind speed ranges from 8 - 10 m/s, and the high percentage for fluctuation of wind speed ranges 0 - 2 m/s. From Figure 5.11 (b), the high percentage for fluctuation of wind direction is 0°.

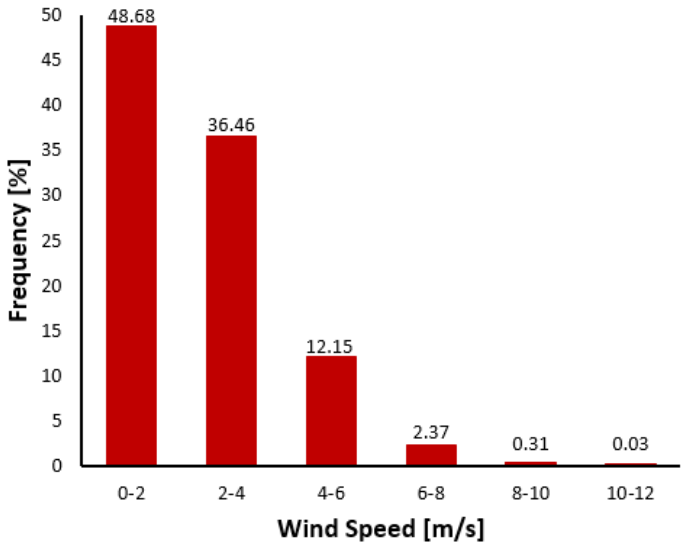


(a) Wind speed

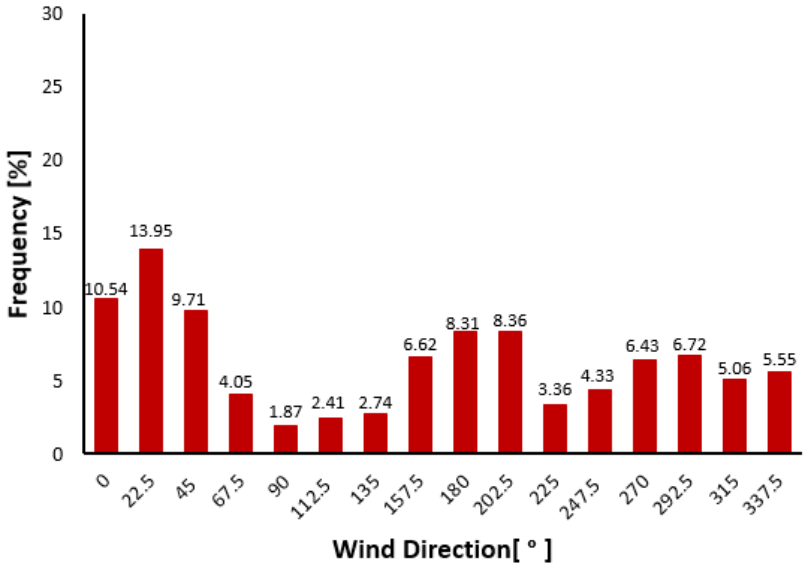


(b) Wind direction

Figure 5.5: Fluctuation of wind speed and direction in Tokushima city

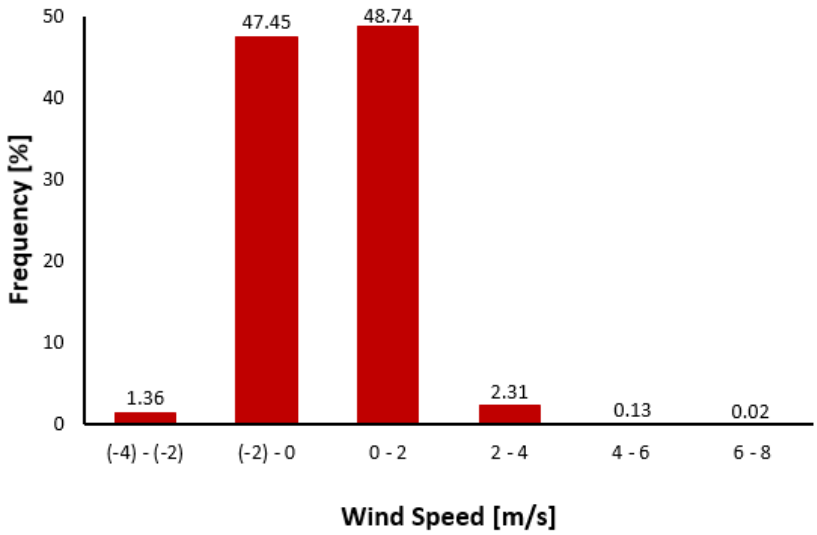


(a) Wind speed

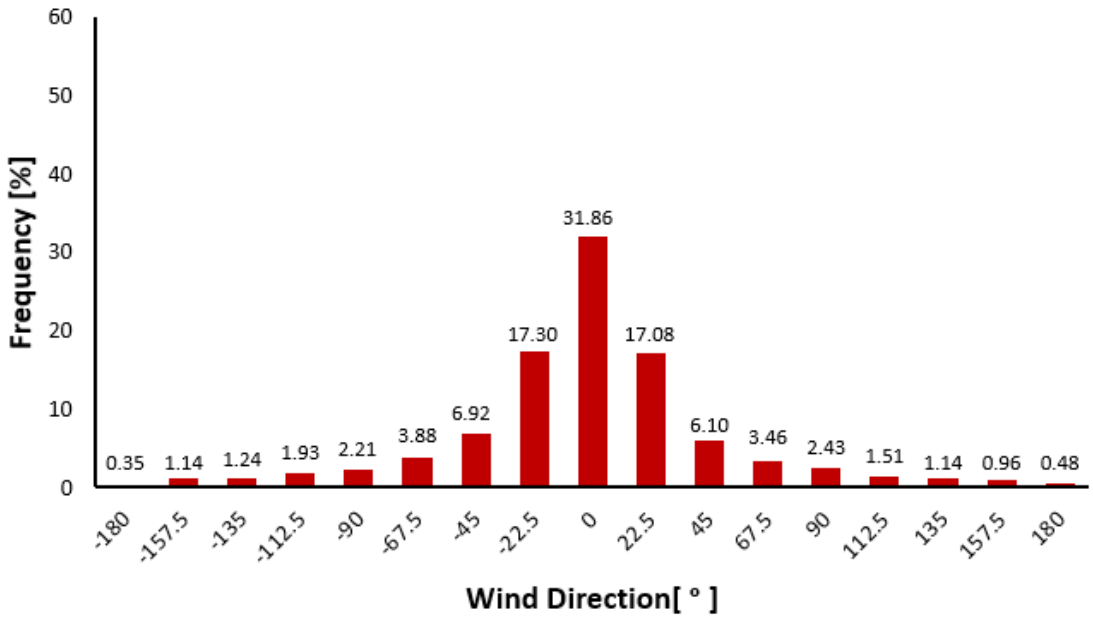


(b) Wind direction

Figure 5.6: Annual wind speed and direction in Takamatsu city

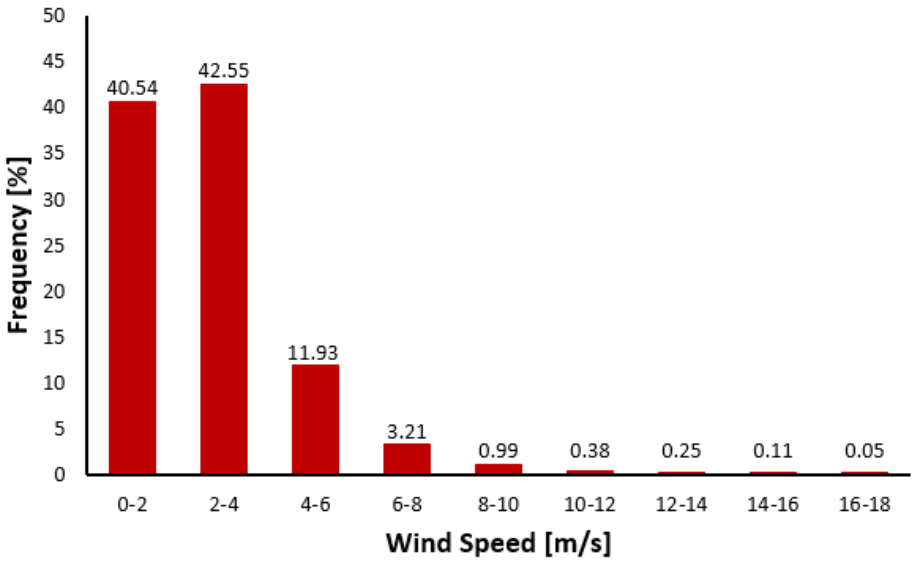


(a) Wind speed

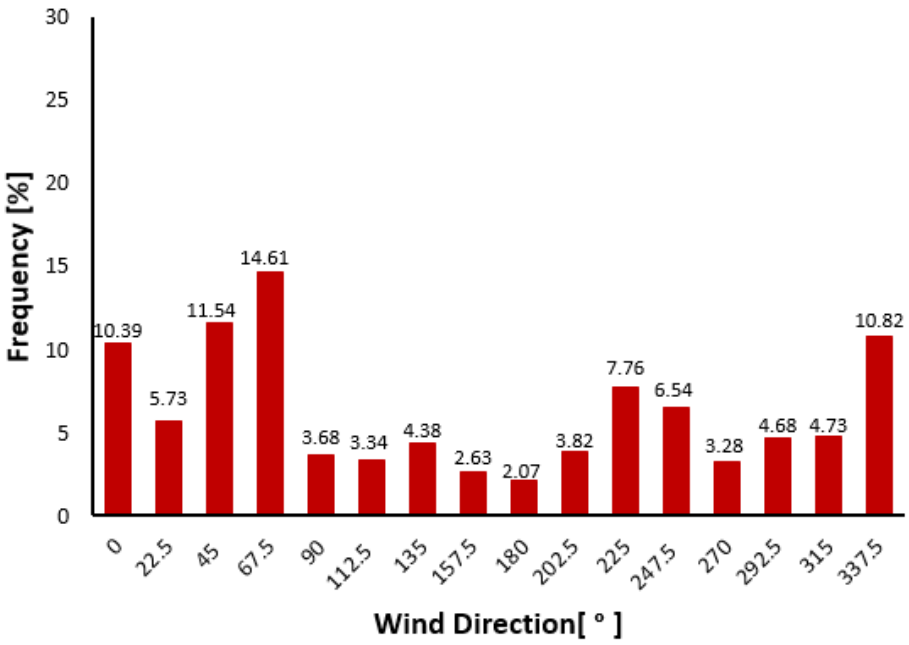


(b) Wind direction

Figure 5.7: Fluctuation of wind speed and direction in Takamatsu city

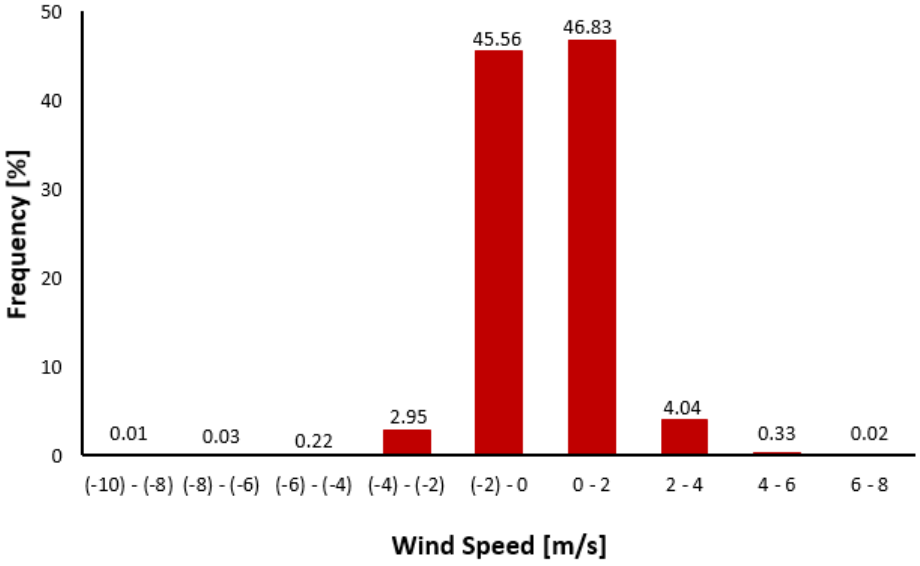


(a) Wind speed

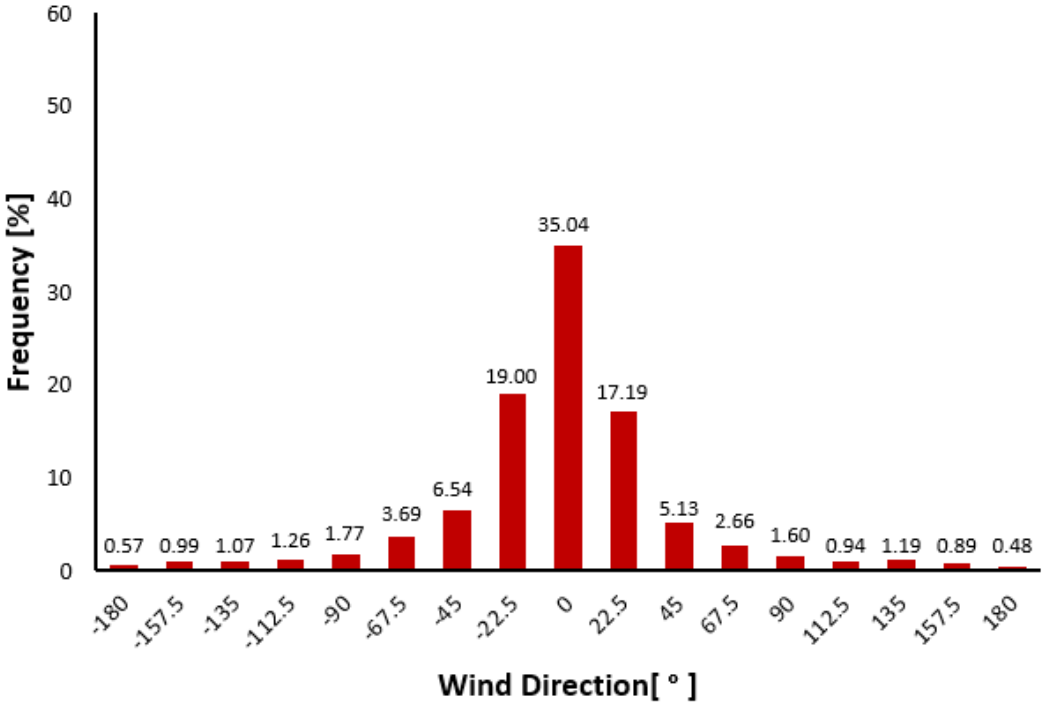


(b) Wind direction

Figure 5.8: Annual wind speed and direction in Hiketa city

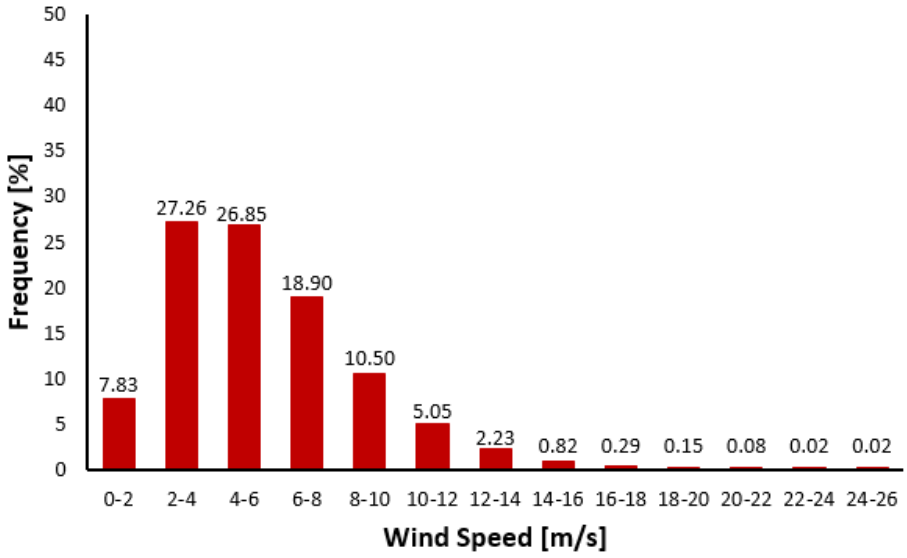


(a) Wind speed

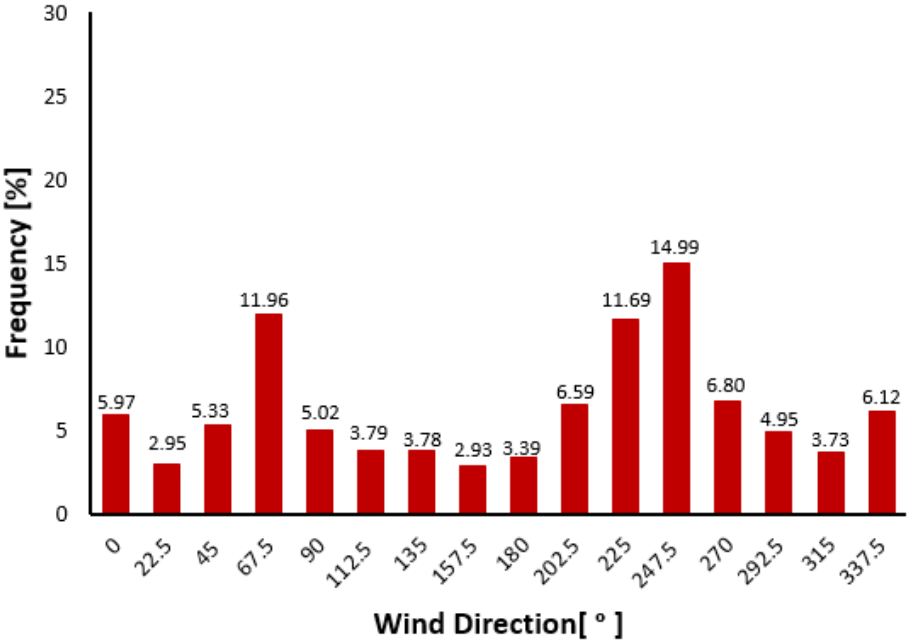


(b) Wind direction

Figure 5.9: Fluctuation of wind speed and direction in Hiketa city

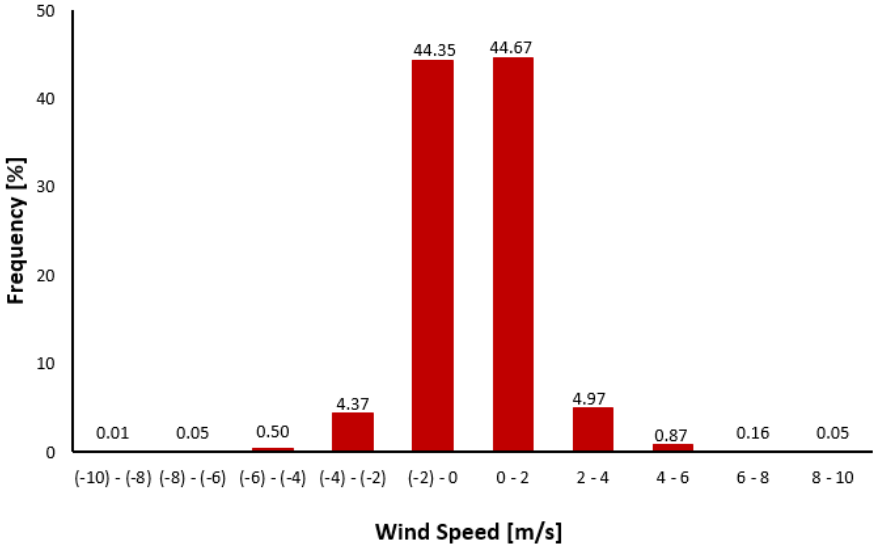


(a) Wind speed

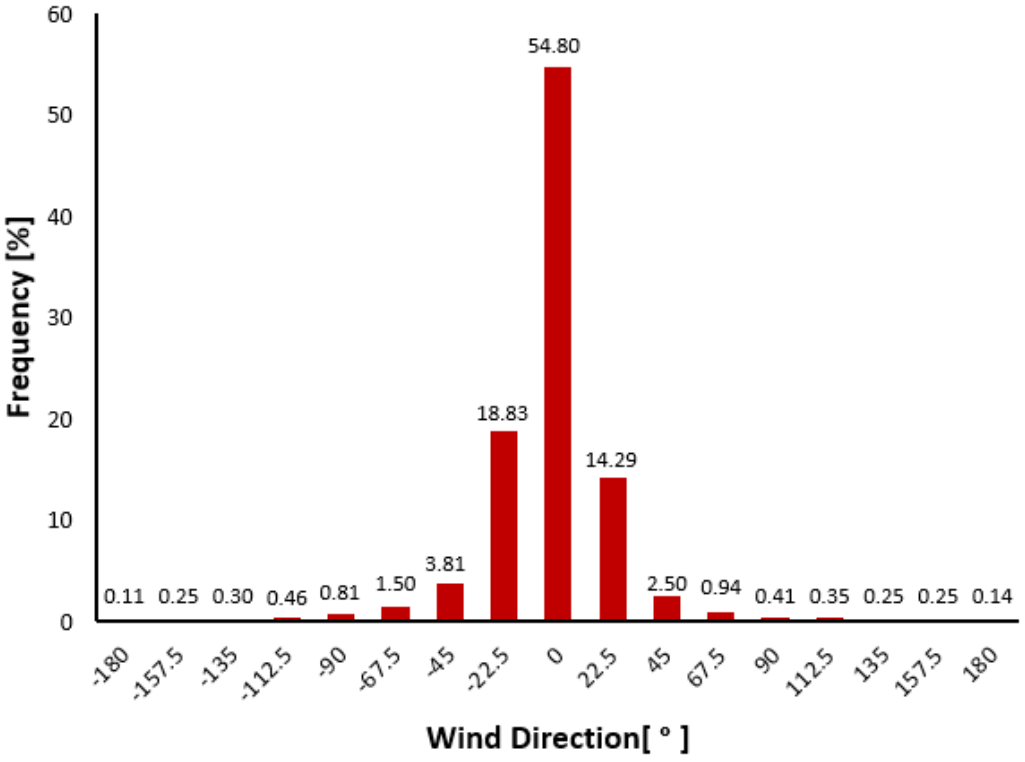


(b) Wind direction

Figure 5.10: Annual wind speed and direction in Choshi city



(a) Wind speed



(b) Wind direction

Figure 5.11: Fluctuation of wind speed and direction in Choshi city

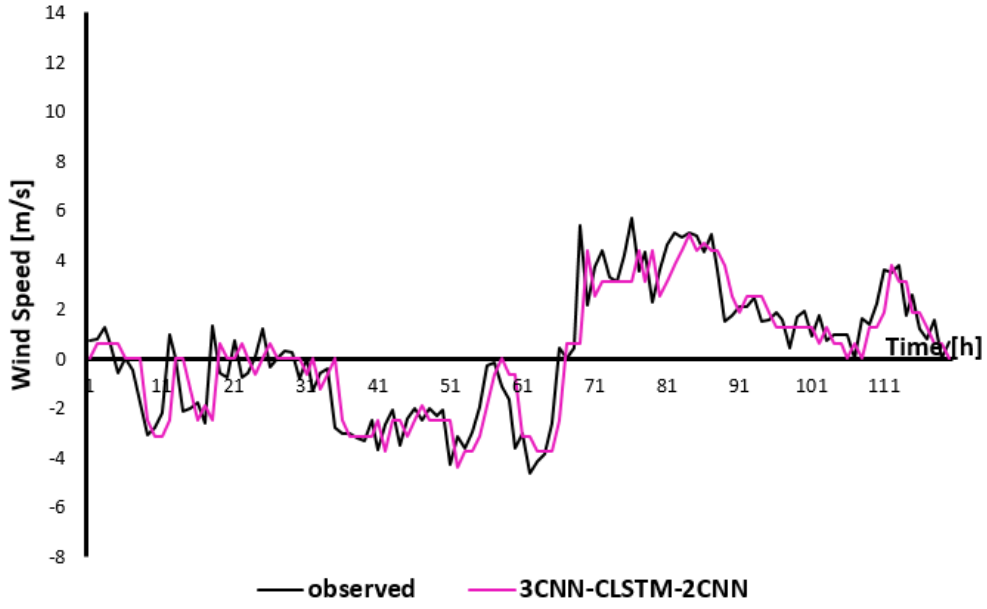
5.2 Forecasting results of v_X and v_Y in four cities

Firstly, this research calculates and analyzes forecasting results of v_X and v_Y in four cities by Eqs. 4.1 and 4.2. Forecasting results of v_X and v_Y in four cities show actual observed data and forecasted data for five days on Oct., 1-5, 2019. Forecasting results of v_X and v_Y in Tokushima city can see in Figure 4.4. Figure 5.12 shows forecasting results of v_X and v_Y in Takamatsu city. Figure 5.13 shows forecasting results of v_X and v_Y in Hiketa city. Forecasting results of v_X and v_Y in Choshi city as shown in Figure 5.14. From Figure 4.4, the different range between past data and current data is 0.0065 - 1.2887 m/s in v_X and 0 - 1.7064 m/s in v_Y. From Figure 5.12, the different range between past data and current data is 0 - 1.325 m/s in v_X and 0.0064 - 1.7731 m/s in v_Y. From Figure 5.13, the different range between past data and current data is 0.0125 - 1.2111 m/s in v_X and 0 - 2.2063 m/s in v_Y. From Figure 5.14, the different range between past data and current data is 0 - 1.0474 m/s in v_X and 0 - 1.1313 m/s in v_Y.

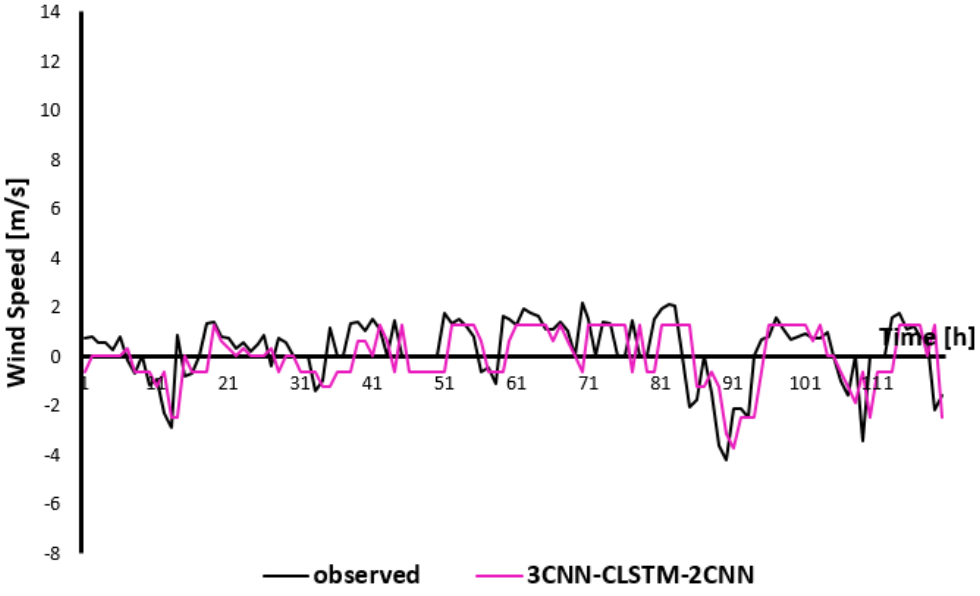
A total forecasting error (RMSE) of v_X and v_Y in four cities from Dec. 2018 – Nov. 2019 as shown in Table 5.1 and to each month as shown in Figure 5.15. The lowest forecasting accuracy is Choshi city both in v_X and v_Y. The highest forecasting accuracy is Tokushima city in v_X and Takamatsu city in v_Y.

5.3 Forecasting results of wind speed and direction in four cities

The target forecasting results of wind speed and direction are one hour ahead in four cities and calculated by Eqs. 4.3 and 4.4. Comparison actual observed and Forecasting results of wind speed and direction in four cities on five days (Oct., 1-5, 2019) are shown in Figure 4.8 (Tokushima city), Figure 5.16 (Takamatsu city), Figure 5.17 (Hiketa city), and Figure 5.18 (Choshi city). From Figure 5.16 (b), Figure 5.17 (b), and 5.18 (b), the direction of 180° = (-180°). From Figure 4.8, the different range between past and current data is 0 - 1.8 m/s in wind speed and (-45°) - 67.5° in wind direction. From Figure 5.16, the different range between past and current data is 0 - 1.42 m/s in wind speed and (-67.5°) - 135° in wind direction. From Figure 5.17, the different range between past and current data is 0-2.06 m/s in wind speed and (-143.1°) - 36.5° in wind direction. From Figure 5.18, the different range between past and current data is 0.01-1.12 m/s in wind speed and (-14°) - 16.2° in wind direction.

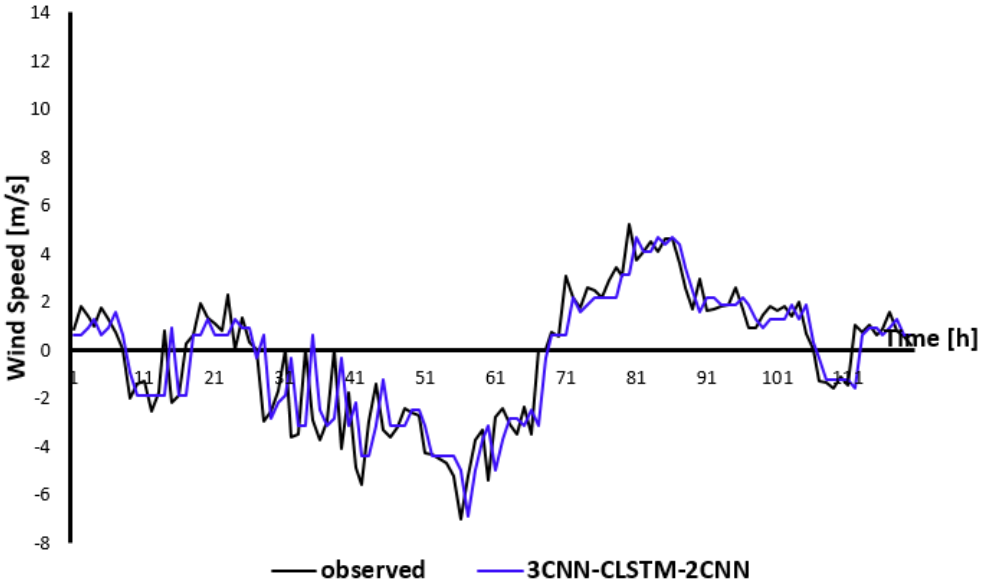


(a) v_X

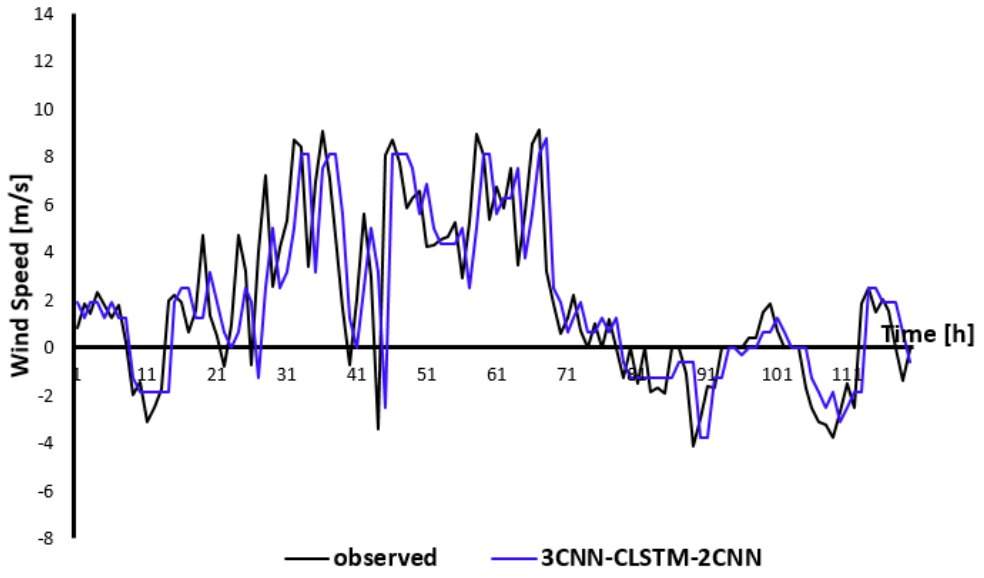


(b) v_Y

Figure 5.12: Forecasting result of v_X and v_Y in Takamatsu city

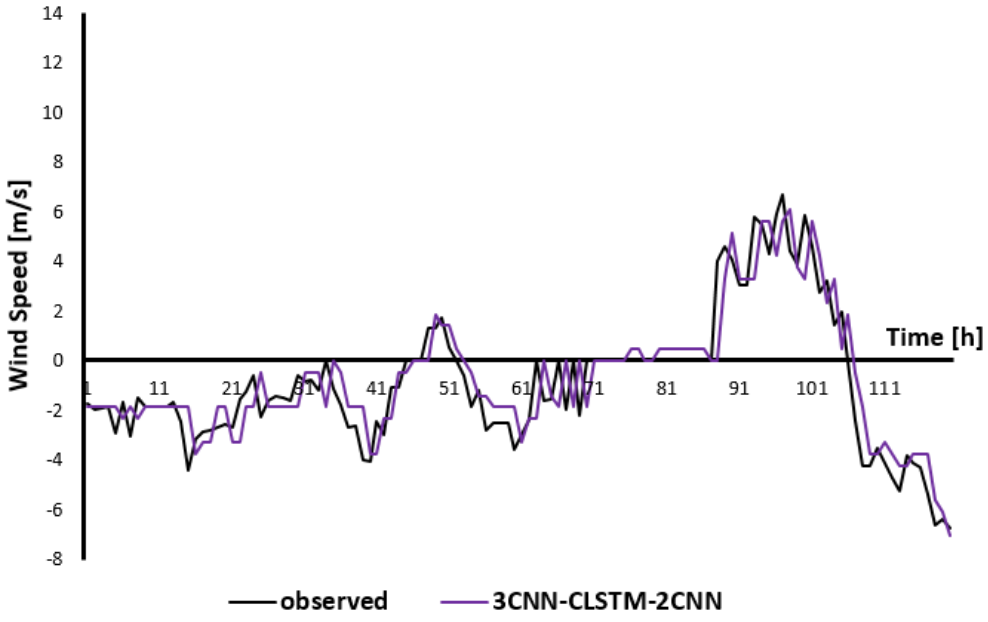


(a) v_X

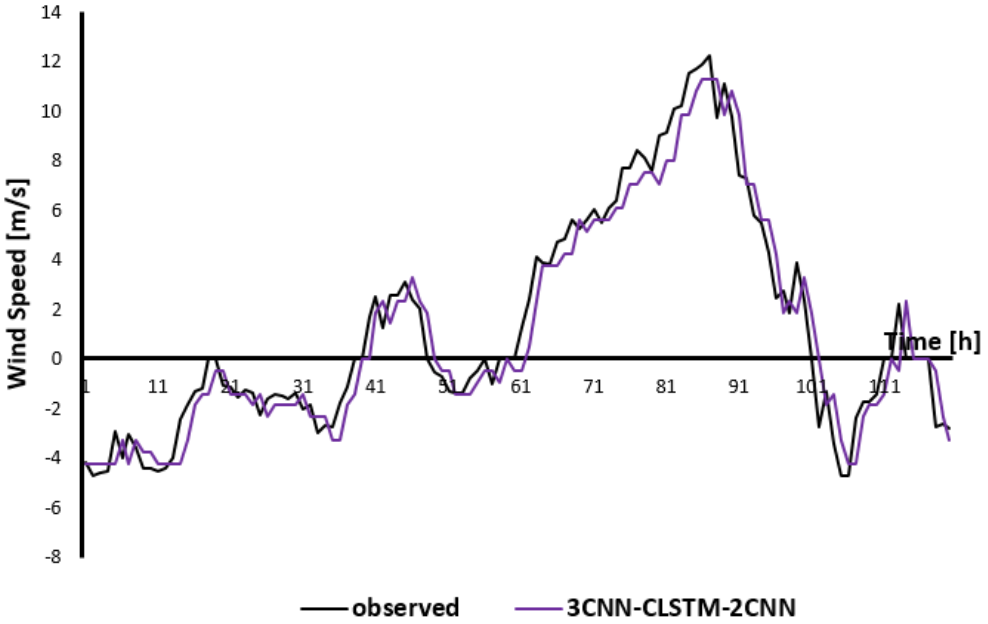


(b) v_Y

Figure 5.13: Forecasting result of v_X and v_Y in Hiketa city



(a) v_X

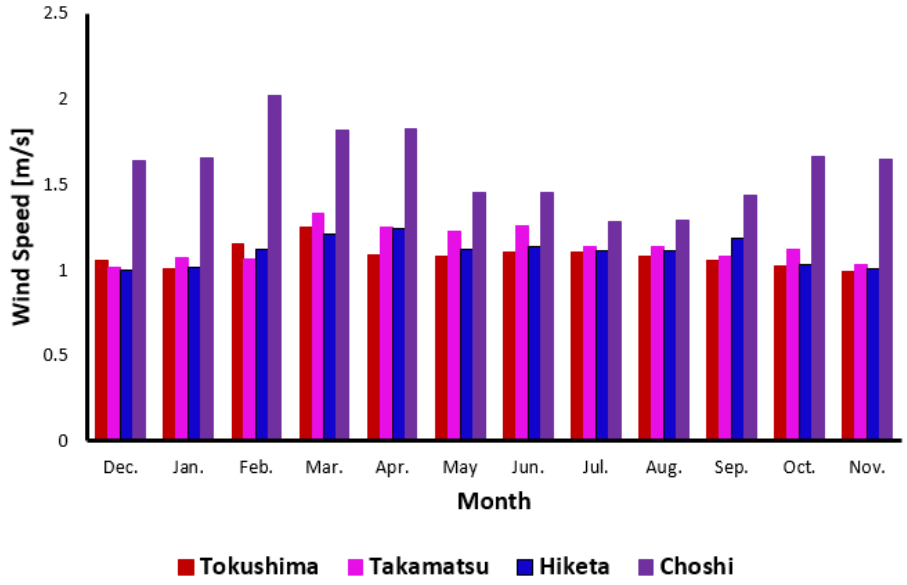


(b) v_Y

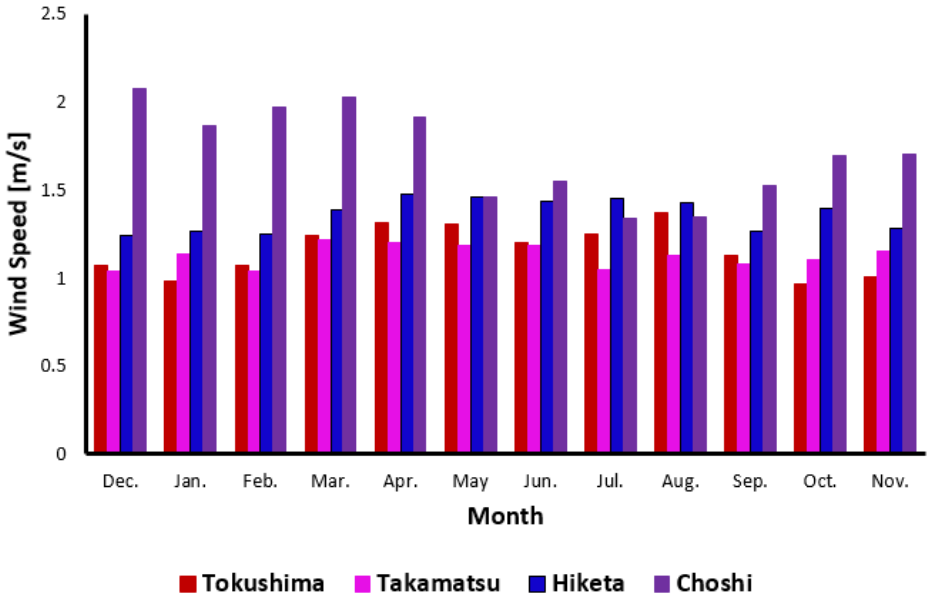
Figure 5.14: Forecasting result of v_X and v_Y in Choshi city

Table 5.1: RMSE of v_X and v_Y in four cities

Description	RMSE [m/s]			
	Tokushima	Takamatsu	Hiketa	Choshi
v_X	1.0836	1.1469	1.1084	1.6075
v_Y	1.1657	1.1264	1.3650	1.7198



(a) v_X



(b) v_Y

Figure 5.15: RMSE of v_X and v_Y in four cities in each month

In chapter 4 and section 5.2, the error of forecasting results is evaluated by RMSE. In this section, the forecasting error of wind speed and direction are evaluated by RMSE, mean absolute error (MAE), and mean absolute percentage error (MAPE). RMSE is used for mean forecasting. MAE is used for median forecasting. MAPE is used to measure forecasting accuracy that easily explain and interpret errors in terms of percentage. Equation of RMSE can see in Eq. 3.5. Equations of MAE and MAPE as follows,

$$MAE = \frac{1}{N}\sum_{t=1}^{N}|Y_t - \hat{Y}_t| \tag{5.1}$$

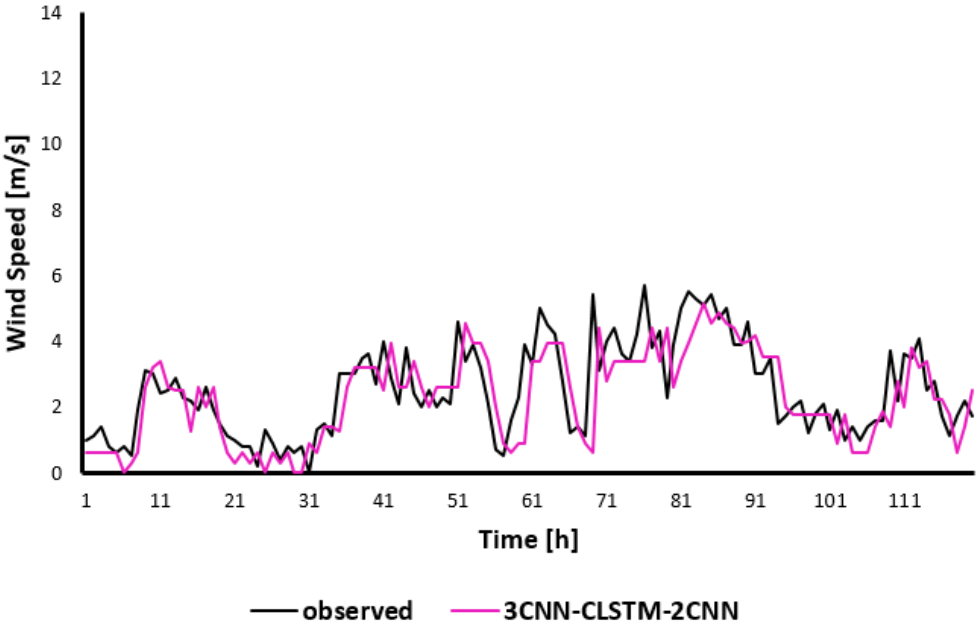
$$MAPE = \frac{1}{N}\sum_{t=1}^{N}|\frac{Y_t - \hat{Y}_r}{Y_t}| * 100 \tag{5.2}$$

where $\hat{Y}_t$ is forecasted data, Y_t is actual observed data, and N is the number of learning data.

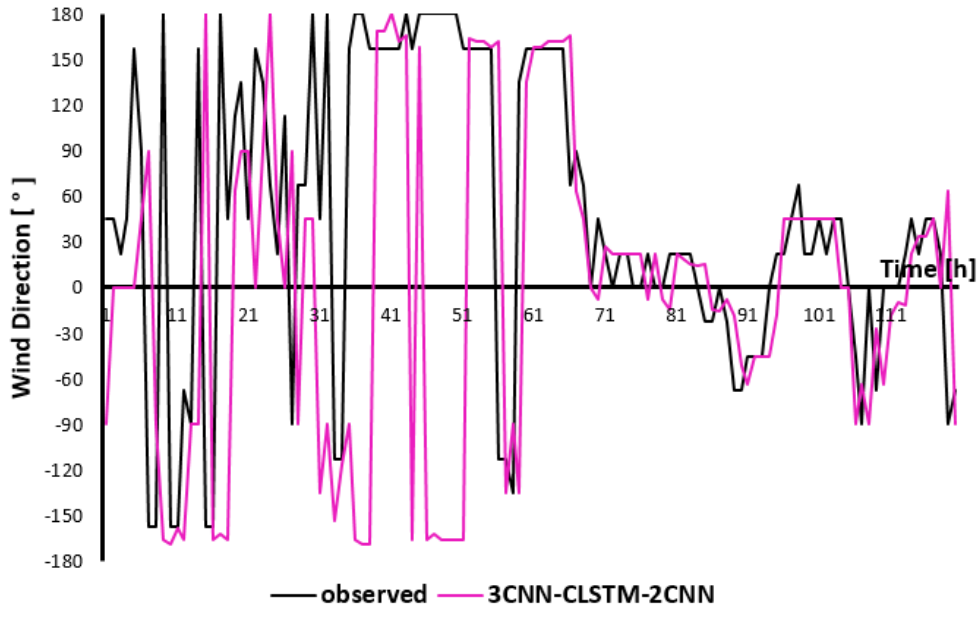
In this section, the research also calculated the persistent model. The persistent model means standard model to short-time forecasting model that forecasting output is the current value of output. The total forecasting error of wind speed and direction in four cities from Dec. 2018 – Nov. 2019 is shown in Tables 5.2 and 5.3, and to each month in Figures 5.19 - 5.22. The persistent model is evaluated by RMSE. The improvement rate is obtained from RMSE compared persistent model for each city. From fluctuation graph of wind speed in four cities (see Figures 5.5 (a), 5.7 (a), 5.9 (a), 5.11 (a)) and standard deviation (SD) in Table 5.2, in order for smallest to largest are Takamatsu, Tokushima, Hiketa, and Choshi cities. From fluctuation graph of wind direction in four cities (see Figures 5.5 (b), 5.7 (b), 5.9 (b), 5.11 (b)) and SD in Table 5.3, in order to smallest to largest are Choshi, Tokushima, Hiketa, and Takamatsu cities.

Comparison between forecasting error of wind speed in Tables 5.2 and fluctuation graph of wind speed in four cities is the same order: Takamatsu (SD = 0.9167 m/s, RMSE = 0.9755 m/s), Tokushima (SD = 1.0153 m/s, RMSE = 1.0180 m/s), Hiketa (SD = 1.1347 m/s, RMSE = 1.1520 m/s), and Choshi (SD = 1.3513 m/s, RMSE = 1.3969 m/s). Comparison between forecasting error of wind direction in Tables 5.3 and fluctuation graph of wind direction in four cities is the same order: Choshi (SD = 30.1256°, RMSE = 30.4677°), Tokushima (SD = 45.3046°, RMSE = 43.4435°), Hiketa (SD = 49.7260°, RMSE = 48.4974°), and Takamatsu (SD = 52.9903°, RMSE = 56.0314°). So that, the 3CNN-CLSTM-2CNN model can handle and suitable to apply in four cities although different characteristics of wind speed and direction. From Tables 5.2 and 5.3, the lowest forecasting accuracy is Hiketa city in wind

speed and Takamatsu city in wind direction. Afterward, the largest forecasting accuracy is Choshi city both in wind speed and direction. To calculate MAE is used the absolute value for evaluating forecasting results. Then, forecasting errors have a small variance so that MAE is slightly lower than RMSE is shown in Tables 5.2 and 5.3.

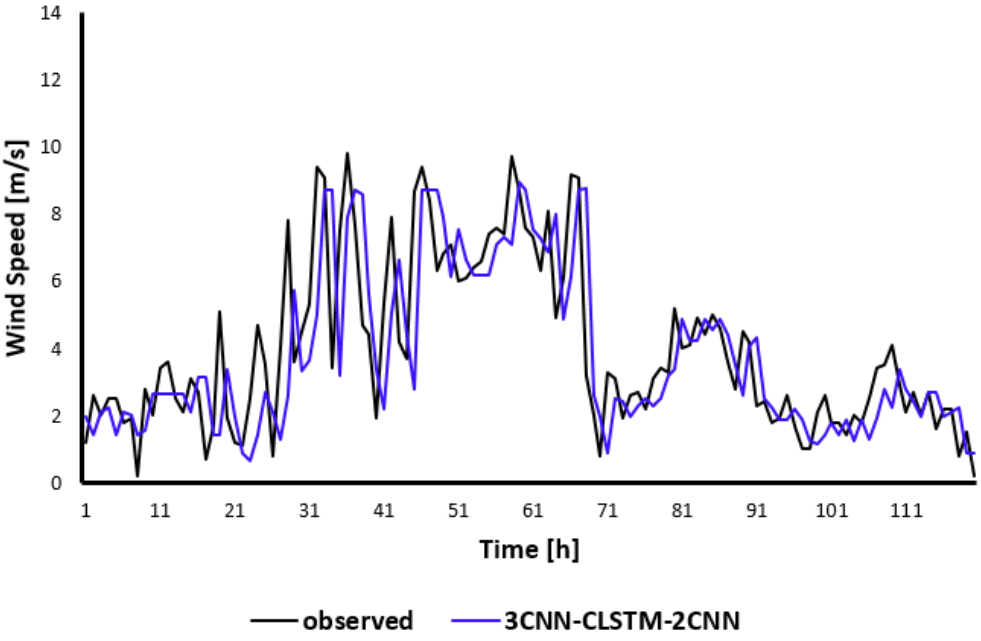


(a) Wind speed

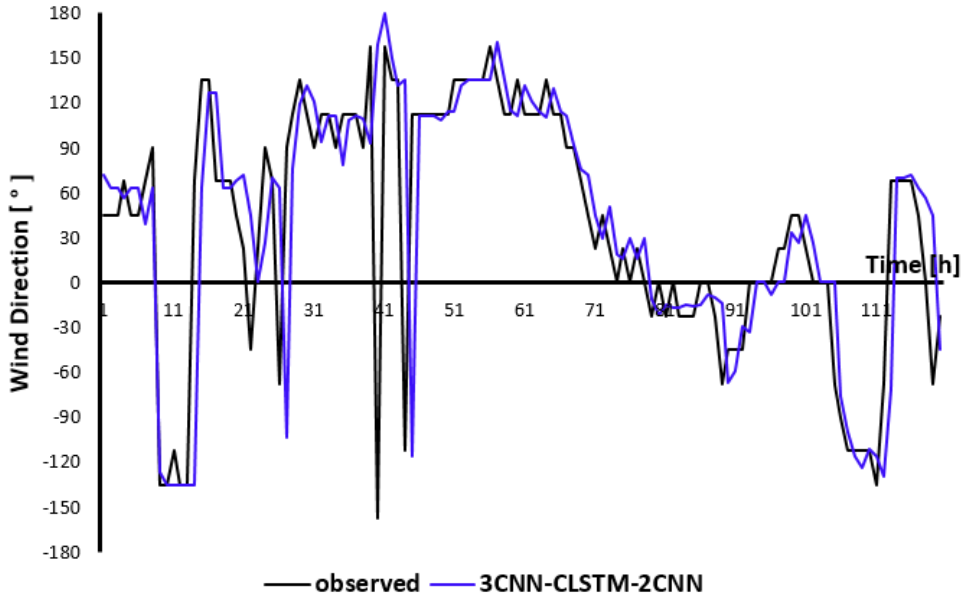


(b) Wind direction

Figure 5.16: Forecasting result of wind speed and direction in Takamatsu city

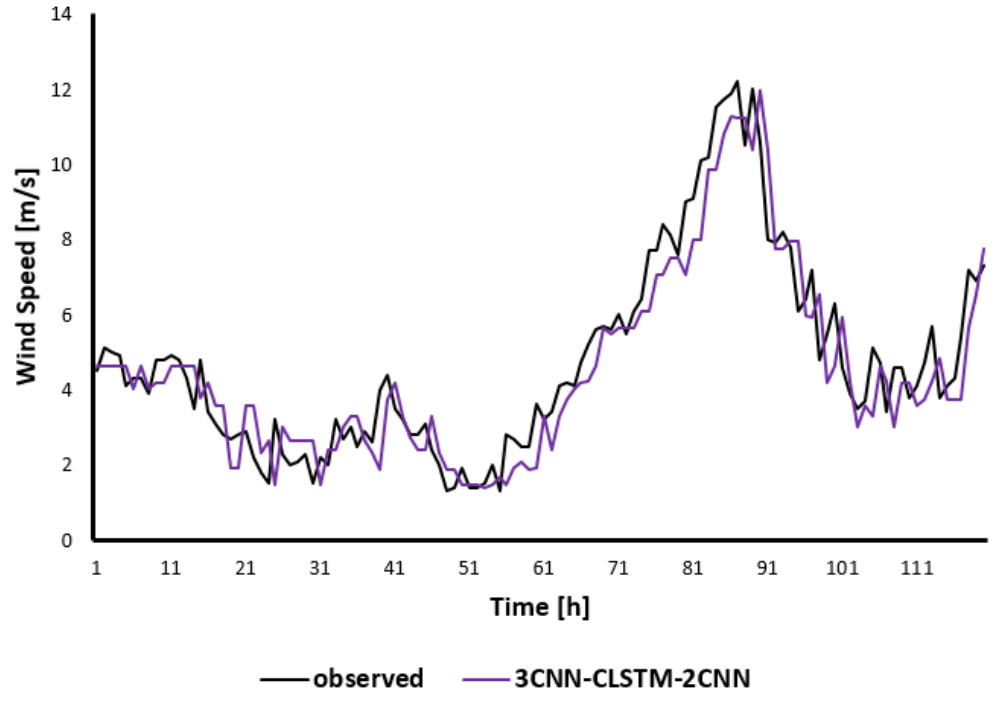


(a) Wind speed

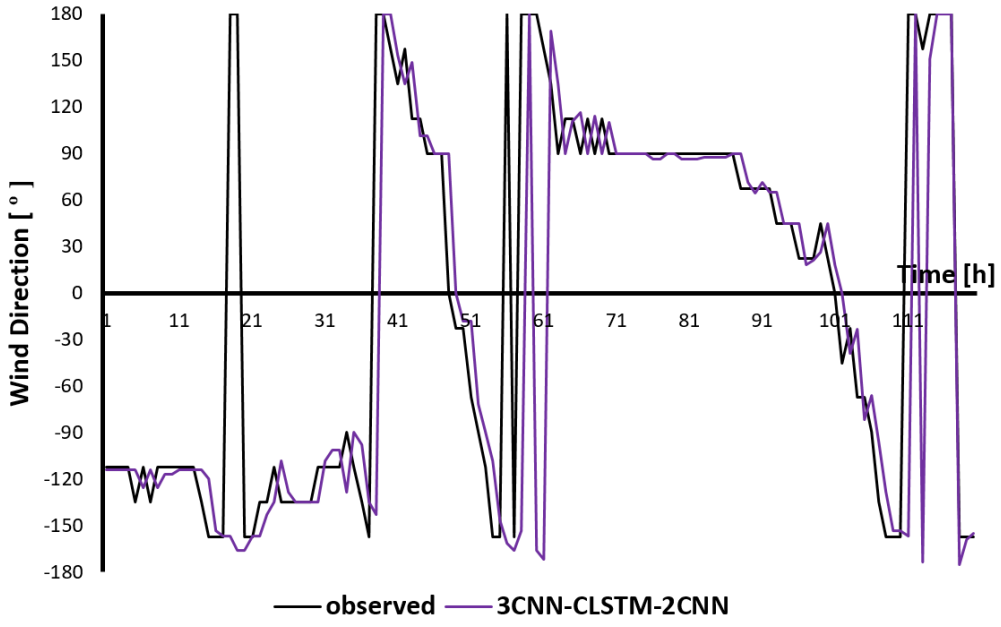


(b) Wind direction

Figure 5.17: Forecasting result of wind speed and direction in Hiketa city



(a) Wind speed



(b) Wind direction

Figure 5.18: Forecasting result of wind speed and direction in Choshi city

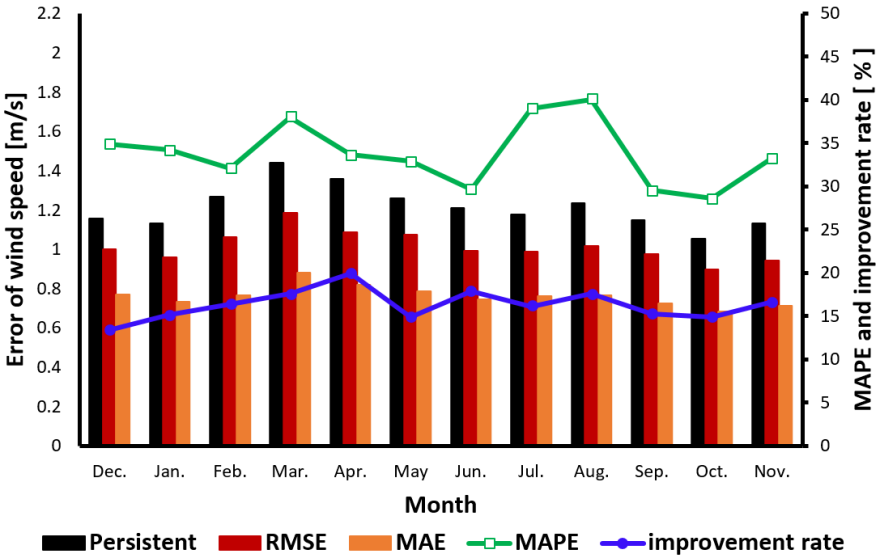
Table 5.2: SD and error of wind speed in four cities

Description	SD [m/s]	Forecasting error				improvement
		Persistent	RMSE [m/s]	MAE [m/s]	MAPE [%]	rate[%]
Tokushima	1.0153	1.2188	1.0180	0.7627	33.8639	16.48
Takamatsu	0.9167	1.1802	0.9755	0.7333	41.8929	17.34
Hiketa	1.1347	1.3767	1.1520	0.8258	41.3432	16.32
Choshi	1.3513	1.8349	1.3969	1.0154	22.6225	23.87

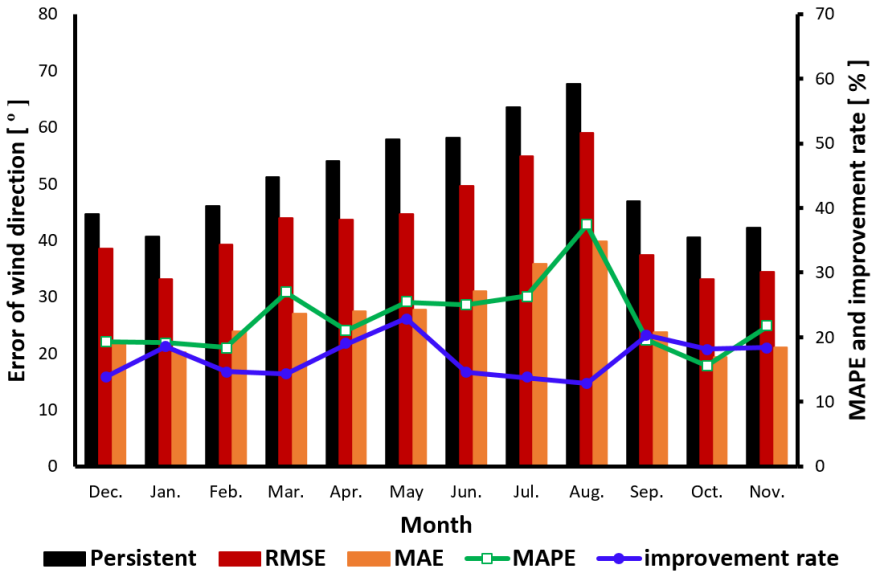
Table 5.3: SD and error of wind direction in four cities

Description	SD [°]	Forecasting error				improvement
		Persistent	RMSE [°]	MAE [°]	MAPE [%]	rate[%]
Tokushima	45.3046	51.9300	43.4435	26.6390	23.0317	16.34
Takamatsu	52.9903	63.3147	56.0314	38.3026	55.6538	11.50
Hiketa	49.7260	59.5280	48.4974	30.9133	41.2880	18.53
Choshi	30.1256	39.7542	30.4677	17.5200	20.3945	23.36

The result of MAPE in Tables 5.2 and 5.3 shows the lowest result of MAPE in Choshi city and the highest result of MAPE in Takamatsu city both in wind speed and direction. In this case, the lower result of MAPE is good so that the forecasting result is more approaching actual observed data. From Figure 5.19, the range of forecasting accuracy in Tokushima city is 13.43 - 19.98% in wind speed and 12.87 - 22.86% in wind direction. From Figure 5.20, the range of forecasting accuracy in Takamatsu city is 11.86 - 19.75% in wind speed and 8.65 - 13.73% in wind direction. From Figure 5.21, the range of forecasting accuracy in Hiketa city is 6.20 - 19.15% in wind speed and 14.39 - 24.51% in wind direction. From Figure 5.22, the range of forecasting accuracy in Choshi city is 20.16 - 26.25% in wind speed and 17.42 - 26.49% in wind direction.

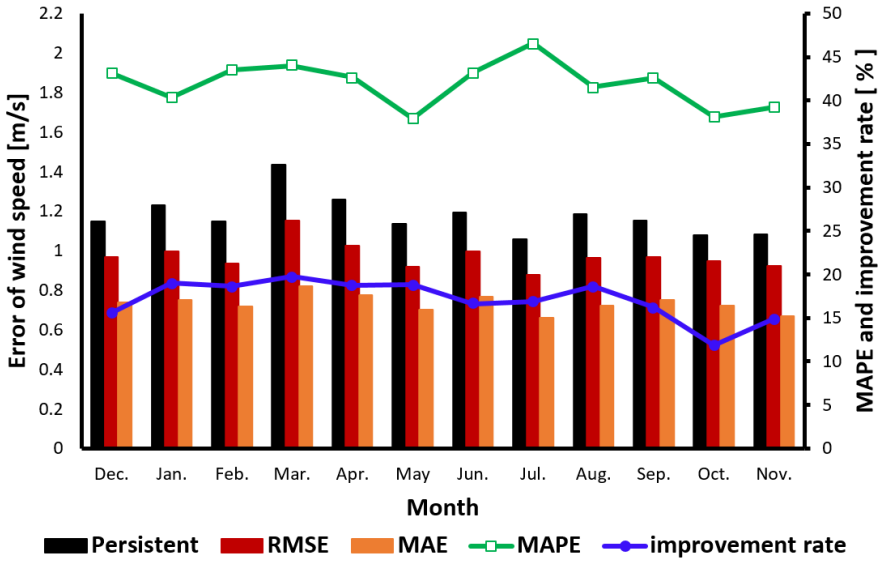


(a) Wind speed

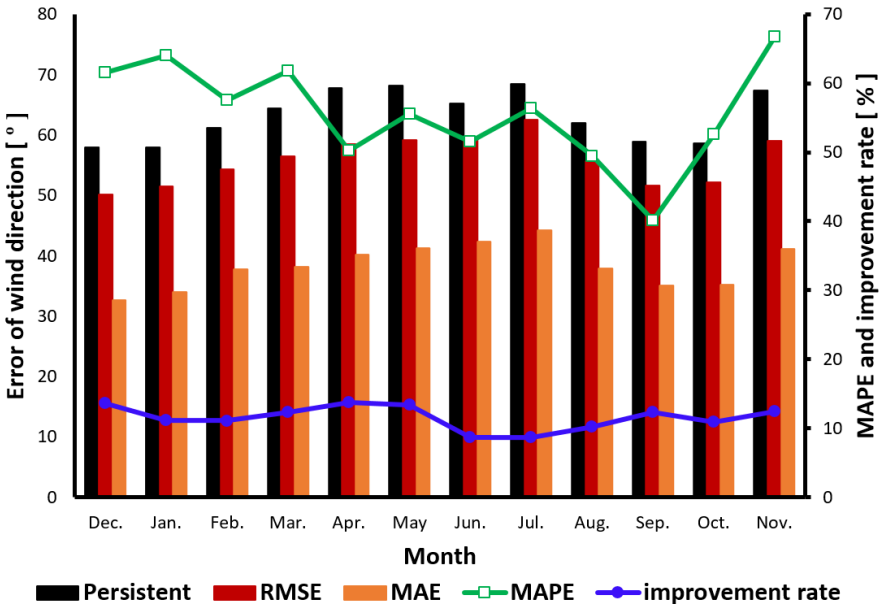


(b) Wind direction

Figure 5.19: Forecasting error of wind speed and direction in Tokushima city

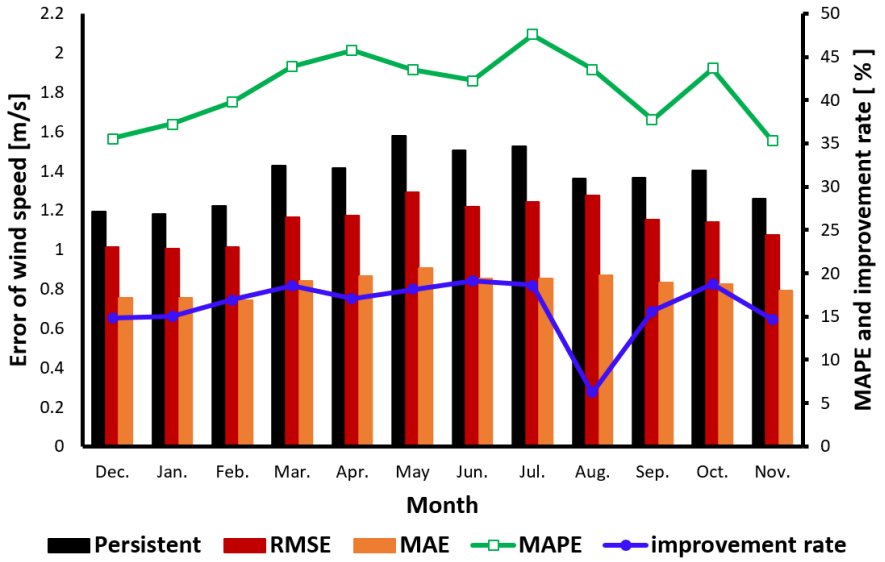


(a) Wind speed

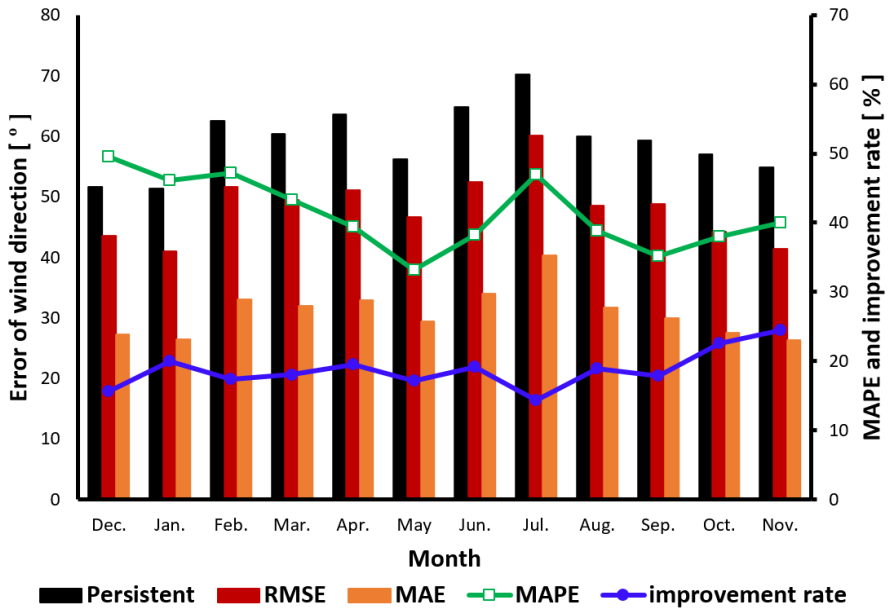


(b) Wind direction

Figure 5.20: Forecasting error of wind speed and direction in Takamatsu city

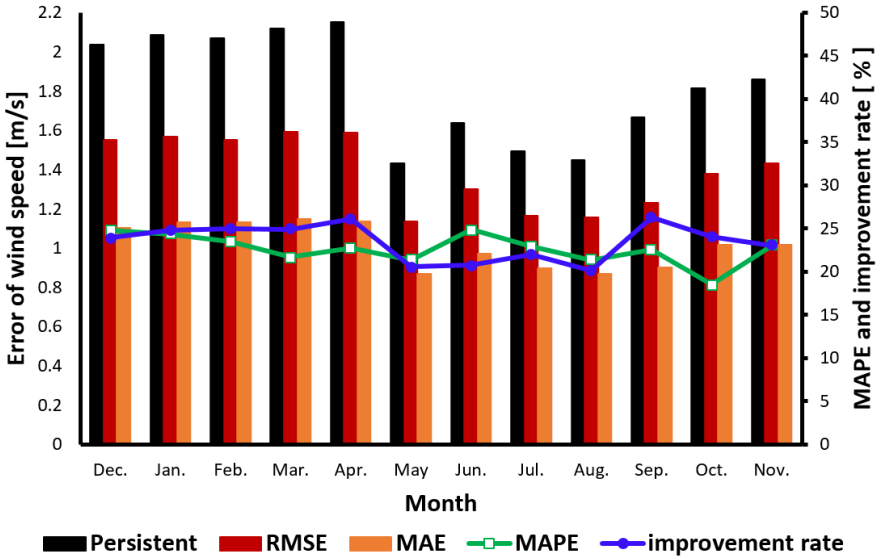


(a) Wind speed

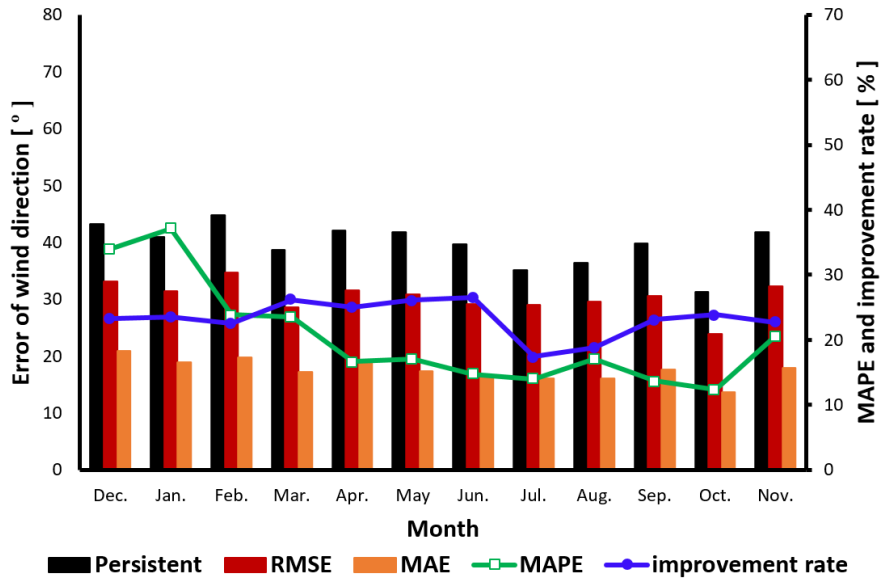


(b) Wind direction

Figure 5.21: Forecasting error of wind speed and direction in Hiketa city



(a) Wind speed



(b) Wind direction

Figure 5.22: Forecasting error of wind speed and direction in Choshi city

CHAPTER 6

Conclusions

In this doctoral thesis, short-term forecasting wind speed and direction by a deep neural network (DNN) are mainly discussed. The proposed forecasting model using the 3CNN-CLSTM-2CNN model. The input of the forecasting model utilizes wind speed and direction which represent an image on the 2D-coordinate system and make it to the sequential image. The data of wind speed and direction are obtained by AMeDAS, Japan at one hour interval. The prediction period of the forecasting model is one hour ahead. For verifying the effectiveness of the proposed forecasting model is evaluated using RMSE and compared by FC-LSTM, DCLSTM, and 3CNN-CLSTM models. 3CNN-CLSTM-2CNN model is the highest forecasting accuracy and the forecasting result is approached actual observed data than other models. DCLSTM, 3CNN-CLSTM, and 3CNN-CLSTM-2CNN models can resolve spatio-temporal sequence and improve forecasting accuracy better than the FC-LSTM model. In comparison FC-LSTM model, the 3CNN-CLSTM-2CNN model can improve forecasting accuracy than DCLSTM, and 3CNN-CLSTM models effectively. 3CNN-CLSTM-2CNN model can improve forecasting accuracy more than 10% from DCLSTM model and more than 2% from the CNN-CLSTM model both in wind speed and direction.

3CNN-CLSTM-2CNN model is compared in four cities (Tokushima, Takamatsu, Hiketa, Choshi) to confirm an applicability to different characteristics of wind conditions. 3CNN-CLSTM-2CNN model was evaluated by RMSE, MAE, and MAPE between forecasted data and actual observed data in four cities. For calculating MAE is used the absolute value for evaluating forecasting results. Afterward, forecasting errors have a small variance so that MAE is slightly lower than RMSE. In the forecasting error of MAPE, the lower result of MAPE is good caused the forecasting result is more approaching actual observed data. The order of fluctuation wind speed and forecasting accuracy of wind speed in four

cities are same: Takamatsu (SD = 0.9167 m/s, RMSE = 0.9755 m/s), Tokushima (SD = 1.0153 m/s, RMSE = 1.0180 m/s), Hiketa (SD = 1.1347 m/s, RMSE = 1.1520 m/s), and Choshi (SD = 1.3513 m/s, RMSE = 1.3969 m/s). The order of fluctuation wind direction and RMSE of wind direction in four cities are same: Choshi (SD = 30.1256°, RMSE = 30.4677°), Tokushima (SD = 45.3046°, RMSE = 43.4435°), Hiketa (SD = 49.7260°, RMSE = 48.4974°), and Takamatsu (SD = 52.9903°, RMSE = 56.0314°). So that, the 3CNN-CLSTM-2CNN model can handle and suitable to apply in four cities although different characteristics of wind speed and direction.

The improvement rate of the 3CNN-CLSTM-2CNN model is obtained by comparison between RMSE of the 3CNN-CLSTM-2CNN model and the persistent model in each city. The lowest forecasting accuracy is Hiketa city in wind speed and Takamatsu city in wind direction. The largest forecasting accuracy is Choshi city both in wind speed and wind direction. The lowest result of MAPE in Choshi city both in wind speed and direction. The highest result of MAPE in Takamatsu city both in wind speed and direction.

3CNN-CLSTM-2CNN and 3CNN-CLSTM models improve forecasting accuracy than DCLSTM due to these models uses CNN that can easier for training and reduce training time. 3CNN-CLSTM-2CNN model can improve accuracy than 3CNN-CLSTM model due to the last process of 3CNN-CLSTM-2CNN model uses 2DCNN that can easier for training and effectively to process sequential time.

◇ ◇ ◇ ◇ REFERENCES ◇ ◇ ◇ ◇

[1] REN21: "Renewable 2020 Global Status Report" , [cited 2021, Mar 5], available from: https://www.ren21.net/gsr-2020/ (2021).

[2] A.P. Sari, H. Suzuki, T. Kitajima, T. Yasuno, and D.A. Prasetya: "Prediction model of wind speed and direction using deep neural network", Journal of Electrical Engineering, Mechatronics and Computer Science (JEEMECS), vol. 3, no. 1, pp. 1 – 10, (2020).

[3] N. L. Panwar, S.C. Kaushik, and S. Kothari: "Role of renewable energy sources in environmental protection: A review", Elsevier Journal : Renewable and Sustainable Energy Reviews, vol. 15, no. 3, pp. 1513 – 524, (2011).

[4] G.V. Laksmi and P.P. Divakar, "Role of renewable energy sources in environmental protection", Proc. Two-Day International Conference on "Materials for Energy and Environmental Protection"(ICMEEP-18), pp. 12 – 17, (2018).

[5] A.P. Sari, H. Suzuki, T. Kitajima, T. Yasuno, D.A. Prasetya, and A. Rabi': "Prediction of wind speed and direction using encoding-forecasting network with convolutional long short-term memory", Proc. 2020 59th Annual Conference of the Society of Instrument and Control Engineers (SICE), pp. 958 – 963, (2020).

[6] Statista: "Trends of installed wind power capacity in the world", [cited 2021, Mar 5], available from: https://www.statista.com/statistics/268363/installed-wind-power-capacity-worldwide/, (2021).

[7] Renewable energy institute: "Trends of installed wind power capacity in Japan", [cited 2021, Mar 5], available from: https://www.renewable-ei.org/en/statistics/re/?cat=wind, (2021).

[8] Wikiwand: "Wind power capacity and production", [cited 2021, May 7], available from: https://www.wikiwand.com/en/Wind-power, (2021).

[9] Q. Zhu, J. Chen, D. Shi, L. Zhu, X. Bai, X. Duan, and Y. Liu: "Learning temporal and spatial correlations jointly: a unified framework for wind speed prediction", Journal of IEEE Transaction on Sustainable, vol. 11, no. 1, pp. 509 – 523, (2020).

[10] H. Liu, X. Mi, and Y. Li: "Smart deep learning based wind speed prediction model using wavelet packet decomposition, convolutional neural network and convolutional long short term memory network", Journal of Elsevier: Energy Conversion and Management, vol. 166, pp. 120 – 131, (2018).

[11] H. Liu, X. Mi, Y. Li, Z. Duan, Y. Xu: "Smart wind speed deep learning based multi-step forecasting model using singular spectrum analysis, convolutional gated recurrent unit network and support vector regression", Journal of Elsevier: Renewable Energy, vol. 143, pp. 842 – 854, (2019).

[12] U. Cali and V. Sharma: "Short-term wind power forecasting using long-short term memory based recurrent neural network model and variable selection", International journal of Smart Grid and Clean Energy, vol. 8, no. 2, pp. 103 – 110, (2019).

[13] L. Chen, Z. Li, and Y. Zhang: "Multiperiod-ahead wind speed forecasting using deep neural architecture and ensemble learning", Research article Hindawi: Mathematical Problems in Engineering, vol. 2019, pp. 1 – 14, (2019).

[14] A. Marndi, G.K. Patra, and K.C. Gouda: "Short-term forecasting of wind speed using time division ensemble of hierarchical deep neural networks", Research article: Bulletin of Atmospheric Science and Technology, vol. 1, pp. 91 – 108, (2020).

[15] Q. Zhu, J. Chen, L. Zhu, X. Duan, and Y. Liu: "Wind speed prediction with spatio-temporal correlation: a deep learning approach", Journal of energies, vol. 11, no. 4:705, pp. 1 – 18, (2018).

[16] A. Daraeepour and D.P. Echeverri: "Day-ahead Wind Speed Prediction by a Neural Network-based Model", Proc. Innovative Smart Grid Technologies Conference, pp. 1 – 5, (2014).

[17] H. Ramchoun, M.A.J. Idrissi, Y. Ghanou, M. Ettaouil: "Multilayer perceptron: architecture optimization and training", International Journal of Interactive Multimedia and Artificial Intelligence, vol. 4, no. 1, pp. 26 – 30, (2016).

[18] A.P. Sari, H. Suzuki, T. Kitajima, T. Yasuno, D.A. Prasetya, and Nachrowie: "Prediction model of wind speed and direction using convolutional neural network - long short term memory", Proc. 2020 IEEE International Conference on Power and Energy (PECon), pp. 358 – 363, (2020).

[19] S.Raschka and V. Mirjalili: "Python machine learning second edition: machine learning and deep learning with python, scikit-learn and tensorflow", Packt Publishing: Mumbai, (2017).

[20] A. Fandango: "Mastering tensorflow 1.x", Packt Publishing: Mumbai, (2018).

[21] M.Z. Alom, T.M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M.S. Nasrin, M. Hasan, B.C.V. Essen, A.A.S. Awwal, and V.K. Asari: "A state-of-the-art survey on deep learn-

ing theory and architectures", Journal of Electronics, vol. 8:292, pp. 1 – 67, (2019).

[22] Y. Zhang, H. Yuan, J. Wang, and X. Zhang: "YNU-HPCC at EmoInt-2017: Using a CNN-LSTM Model for Sentiment Intensity Prediction", Proc. 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 200 – 204, (2017).

[23] X. Shi, Z. Chen, H. Wang, D.Y. Yeung, W.K. Wong, and W.C. Woo: "Convolutional LSTM network : A machine learning approach for precipitation nowcasting", Journal of arXiv, Vol. abs/1506.04214, pp. 1 – 9, (2015).

[24] S. Liang, L. Nguyen, F. Jin, "A multi-variable stacked long-short term memory network for wind speed forecasting ", Journal of arXiv, vol. 1811.09735v1, pp. 1 – 4, (2018).

[25] W. Byeon, T.M. Breuel, F. Raue, and M. Liwicki, " Scene labeling with LSTM recurrent neural networks ", Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp. 3547 – 3555, (2015).

[26] R. Fukuoka, H. Suzuki, T. Kitajima, A. Kuwahara, T. Yasuno and K. Takigawa, " Prediction of rapid wind speed fluctuation by LSTM using observed data of multiple points ", Proc. 2019 RISP International Workshop on Nonlinear Circuits, Communications and Signal Processing (NCSP2019), pp. 76 – 79, (2019).

[27] Z.C. Lipton, D.C. Kale, C. Elkan, and R. Wetzel: "Learning to diagnose with LSTM recurrent neural networks", Journal of arXiv, vol. 1511.03677v7, pp. 1 – 18, (2017).

[28] T.N. Sainath, O. Vinyals, A. Senior, and H. Sak: "Convolutional, long short-term memory, fully connected deep neural networks", Proc. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4580 – 4584, (2015).

[29] X. Chen, L. Ma, W. Jiang, J. Yao, and W. Liu:" Regulazing RNNs for caption generation by reconstructing the past with the present", IEEE explore, CVPR paper: provided Computer Vision Foundation, pp. 7795 – 8003, (2018).

[30] S. Agethen and W.H. Hsu: "Deep multi kernel convolutional LSTM Networks and an attention based mechanism for videos", IEEE Transactions on Multimedia, vol. 22, pp. 819 – 829, (2019).

[31] A.P. Sari, H. Suzuki, T. Kitajima, T. Yasuno, D.A. Prasetya, and A. Rabi': "Deep convolutional long short-term memory for forecasting wind speed and direction", SICE Journal of Control, Measurement, and System Integration (SICE JCMSI), vol. 14, no. 2, pp. 30 – 38, (2021).

[32] R. Castro, Y.M. Souto, E. Ogasawara, F. Porto, and E. Bezerra: "STConvS2S: Spatiotemporal convolutional sequence to sequence network for weather forecasting", Journal of arXiv, vol. 1912.00134v4, (2020).

[33] O. Avci, O. Abdeljaber, S. Kiranyaz, and D. Inman, “ Structural Damage detection in real-time: implementation of 1D convolutional neural networks for SHM applications,” Proc. 35th IMAC, A Conf. Expo. Struct. Dyn. 2017, Springer International Publishing, pp. 49 – 54, (2017).

[34] H. Gao, S. Lin, Y. Yang, C. Li, and M. Yang, “ Convolutional neural network based on two-dimensional spectrum for hyperspectral image classification ”, Journal of sensor: Hindawi, vol. 07, pp. 1 – 13, (2018).

[35] K. Wang, X. Wang, L. Lin, M. Wang, and W. Zuo, “ 3D human activity recognition with reconfigurable convolutional neural networks, ” Proc. 22nd ACM Int. Conf. Multimedia, pp. 97 – 106, (2014).

[36] S. Zhang, S. Guo, W. Huang, M.R. Scott, and L. Wang, “ V4D:4D Convolutional neural networks for video-level representation learning, ” Proc. conference paper at ICLR 2020, pp. 1 – 16, (2020).

[37] R.Wang: ”Research on image generation and style transfer algorithm based on deep learning”, Open Journal of Applied Sciences, vol. 9, no. 8, pp. 661 – 672, (2019).

[38] R. Yasrab: “ECRU, An encoder-decoder based convolution neural network (CNN) for road-scene understanding”, Journal of Imaging, vol. 4:116, no. 10, pp. 1 – 19, (2018).

[39] A. Zhang, Z.C. Lipton, M. Li, and A.J. Smola: “Dive into deep learning Release 0.7”, Amazon Science, (2019).

[40] X. Yin, Y. Li, and B.S. Shin: “TGV upsampling a making-up operation for semantic segmentation”, Research Article Hindawi Computational Intelligence and Neuroscience, vol. 2019, pp. 1 – 12, (2019).

[41] Y.Y. Hong, J.J.F. Martinez, and A.C. Fajardo: “Day-ahead solar irradiation forecasting utilizing gramian angular field and convolutional long short-term memory ”, Journal of IEEE access, vol. 8, pp. 18741 – 18753, (2020).

[42] Thomas M. Breuel: “High performance text recognition using a hybrid convolutional-LSTM implementation”, Proc. 2017 14th IAPR International Conference on Document Analysis and Recognition, pp. 11 – 16, (2017).

[43] L. Zhang, G. Zhu, P. Shen, and J. Song: “Learning spatiotemporal features using 3DCNN and Convolutional LSTM for gesture recognition”, IEEE Xplore: ICCV workshop by Computer Vision Foundation, pp. 3120 – 3128, (2017).

[44] J. Feng and S. Lu: “Performance analysis of various activation function in artificial neural networks”, Journal of Physics: Conference series1237 (ICSP 2019), pp. 1 – 6, (2019).

[45] Y. Wang, Y. Li, Y. Song, and X. Rong: "The influence of the activation function in a convolution neural network model of facial expression recognition", Journal of applied sciences, vol. 10, pp. 1 – 20, (2020).

[46] C.E. Nwankpa, W.L. Ijomah, A. Gachagan, and S. Marshall: "Activation functions comparison of trends in practice and research for deep learning", Proc. 2nd International Conf. on Computational Sciences and Technologies (INCCST 20), pp. 124 – 133, (2020).

Publications

【Journal Papers】

1. Anggraini Puspita Sari, Hiroshi Suzuki, Takahiro Kitajima, Takashi Yasuno, Dwi Arman Prasetya, and Abd. Rabi': Deep Convolutional Long Short-term Memory for Forecasting Wind Speed and Direction, The SICE Journal of Control, Measurement, and System Integration (SICE JCMSI), vol. 14, no. 2, pp. 30 – 38, June 2021, (Published).

2. Anggraini Puspita Sari, Hiroshi Suzuki, Takahiro Kitajima, Takashi Yasuno, and Dwi Arman Prasetya: Prediction Model of Wind Speed and Direction using Deep Neural Network, Journal of Electrical Engineering, Mechatronic, and Computer Science (JEEMECS), vol. 3, no. 1, pp. 1 – 10, February 2020, (Published).

【International Conference Papers】

1. Anggraini Puspita Sari, Hiroshi Suzuki, Takahiro Kitajima, Takashi Yasuno, Dwi Arman Prasetya, and Abd. Rabi': Prediction of Wind Speed and Direction using Encoding-forecasting Network with Convolutional Long Short-term Memory, IEEE Xplore: Proc. 2020 59th Annual Conference of Instrument and Control Engineers of Japan (SICE), pp. 958 – 963, September 2020, (Published).

2. Anggraini Puspita Sari, Hiroshi Suzuki, Takahiro Kitajima, Takashi Yasuno, Dwi Arman Prasetya, and Nachrowie: Prediction Model of Wind Speed and Direction using Convolutional Neural Network-Long Short Term Memory, IEEE Xplore: Proc. 2020 IEEE International Conference on Power and Energy (PECon), pp. 358 – 363, December 2020, (Published).

【Domestic Conference Papers】

1. Anggraini Puspita Sari, Hiroshi Suzuki, Rui Fukuoka, Takahiro Kitajima, and Takashi Yasuno: Prediction Model of Wind Speed and Direction using Convolutional Neural Network, Proc. SICE Shikoku Conference 2019, no. SO2-05, pp. 23 – 27, December

2019, (Published).

2. Anggraini Puspita Sari, Hiroshi Suzuki, Takahiro Kitajima, and Takashi Yasuno: Prediction of Wind Speed and Direction by Deep One-Dimensional Convolutional Neural Network, 2020 Shikoku-Section Joint Convention Record of The Institutes of Electrical and Related Engineers (SJCIEE), no. 3-6, September 2020, (Published).

3. Anggraini Puspita Sari, Hiroshi Suzuki, Takahiro Kitajima, and Takashi Yasuno: Forecasting Model of Wind Speed and Direction using CNN-CLSTM, Proc. SICE Shikoku Conference 2020, no. PS1-18, pp. 69 – 72, December 2020, (Published).

【Award】

1. Best paper award: Anggraini Puspita Sari, Hiroshi Suzuki, Takahiro Kitajima, Takashi Yasuno, Dwi Arman Prasetya, and Nachrowie: Prediction Model of Wind Speed and Direction using Convolutional Neural Network-Long Short Term Memory, 2020 IEEE International Conference on Power and Energy (PECon).

Acknowledgments

In the name of Allah SWT, the Most Gracious and the Most Merciful.

Alhamdulillah, all praise and gratitude to Allah SWT for strength, grace, and His blessing that I completed this dissertation. I am thankful to Allah SWT that gave me affection, courage, and guidance to complete this research.

I realize that many people contributed their idea, energy, time for assisting this research. Firstly, I would like to express the special and deepest appreciation to my Supervisor: Prof. Takashi Yasuno and assistant of my supervisor: Dr. Hiroshi Suzuki who gave me valuable advice through meaningful discussions and taught me to the great writing of research papers. Their support, guidance, encouragement, and the invaluable help of suggestions and comments throughout have contributed success to this research.

Special thanks to Prof. Masahide Hojo and Prof. Yoshifumi Nishio who reviewed my doctoral thesis in advance, provided me their suggestions and comments at the defense presentation. I would like thanks to all members of Yasuno - Suzuki Laboratory group that made a contribution of all sorts, support, and friendship. Especially Dr. Takahiro Kitajima, Mr. Akinobu Kuwahara, and the Renewable Energy team to support, give suggestions, comments, and help along with this research. I would like thanks to teachers and staff of Tokushima university that supported and assisted in all my academic activities and my life in Tokushima city especially Ms. Mito and Ms. Shinohara.

I would like to express my deepest appreciation to the Japanese government (Monbukagakusho : MEXT) that give me sponsorship of doctoral course in Tokushima University, give me chance to study doctoral course, and support my finances until I complete the doctoral course in Tokushima University.

I would like to show the deepest gratitude to my beloved parents: the late Mr. Sujarwo and Mrs. Alimah, my beloved in-laws: the late Mr. Suhantoro and Mrs. Suhartatik Yulianing for their affection, support, and prayers. Special grateful to my beloved husband: Zainul Purwanto for support, trust, love, sacrifice, and prayer. My deepest love to my sons: Azarul Azmi and the late Azril Rafisqy for their love and prayers. They raised my morale whenever I required. I would say thank you to my brother Yanuar Fachrudin Ardiansyah and all my family for their support, love, and prayers.

I would say thank you to the Rector and all staffs of the University of Merdeka Malang, Indonesia that gave me chance, encouragement to study until completed doctoral course in Tokushima university especially Dr. Eng. Dwi Arman Prasetya and all my friends in the Department of Electrical Engineering. I would say thank you to all Indonesian colleagues

especially Squad girls (Ms. Maul, Ms. Feni and Ms. Gigie), Mrs. Basitha's family (Mr. Ario, Mrs. Basitha, and Naomi), all members of the Indonesian student association (PPIJ Tokushima) and PPIJ Tokushima's family for their support, help, nice memories along with life in Tokushima.

Finally, I would like to express thank you to every one of those who supported me, I hope that God blessing your life and best of luck in your endeavors.

September 2021
Tokushima
Anggraini Puspita Sari