Forecasting Natural Disasters of Tornados Using mHGN

Benny Benyamin Nasution, Rahmat Widia Sembiring, Bakti Viyata Sundawa, Nogivenname Gunawan, Afritha Amelia, Nogivenname Ismael, Handri Sunjaya, Suhaili Alifuddin, Morlan Pardede, Nogivenname Junaidi, et al.

Forecasting Natural Disasters of Tornados using mHGN

Benny Benyamin Nasution, Rahmat Widia Sembiring, Bakti Viyata Sundawa,
Gunawan, Afritha Amelia, Ismael, Handri Sunjaya, Suhaili Alifuddin,
Morlan Pardede, Junaidi, Muhammad Syahruddin, Zulkifli Lubis

Politeknik Negeri Medan, Medan 20155, Indonesia
`benny.nasution@polmed.ac.id`

Abstract. Lots of damages, losses, and costs have been the major concern, why handling natural disasters of tornados is very important. Several attempts using different approaches have been carried out, but up to now the results are not yet satisfactory. More promising approaches through a kind of artificial intelligent forecaster have been started for a while, but the results are still not satisfactory either. The capability of mHGN as a pattern recognizer has opened up a new possibility of recognizing a pattern of tornado many hours earlier. Therefore, it can be used to forecast a tornado more efficiently. The results taken from a simulated circumstances of a multidimensional pattern recognition have shown, that the 91% of accuracy can be regarded as satisfactory. Though, several modifications related to the data representation within the mHGN architecture need to be implemented. The deployment of mHGN in several risky areas of tornados can then be expected as a tool for reducing those damages, losses, and costs.

Keywords: Graph Neuron, Hierarchical Graph Neuron, mHGN, Natural Disaster Forecast, Tornado Forecast.

1   Introduction

Different types of natural disasters have struck many countries since millions of years and caused various problems. Natural disasters have caused not only financial problems but also casualties. Yet, people living in those hit areas have not found effective and efficient ways how to cope with it. Developed countries such as USA [1] and Japan [2] are not excepted. Those countries face natural disasters every year [3] and suffer from them. The situation is worse in some developing countries, such as Nepal and Tahiti, where people generally do not know what to do before, during, and after a natural disaster has occurred.

The most difficult part to handle natural disasters is that they come in random times. Although some natural disasters such as volcano eruptions, earthquakes are not coming every day or every month, people cannot prepare the best way to face them. Two tsunami disasters in 2004 (Indonesia) and in 2011 (Japan) are two evidences that people are not adequately nor properly prepared. Due to the randomness of occurrences of natural disasters, it becomes more difficult to handle those frequent ones like tornados,

landslides, and flooding. Not only handling natural disasters is difficult, predicting how bad the damages and costs is still a very challenging task.

To reduce damages, losses, and costs after such unpredictable occurrences people have tried to be prepared as good as possible. Disaster management is the general terminology researchers use for the activity of preparing a number of things before, during, and after a natural disaster has occurred. Additionally, since a few centuries back, researchers have been interested in discovering ways to forecast the upcoming of a natural disaster. Some of them are still at the stage of now-casting [4-8], not yet forecasting. According to their methodologies, the most difficult part of forecasting natural disasters lies in the mathematical formulas. At the moment, the success-rate of such forecasters is around up to 80%.

As it is still difficult to have a measure of disaster forecast based on mathematical formulas, it is a great opportunity to figure out other solving methods, such us through utilizing artificial intelligent technologies. Although mathematical functions that can determine the condition of a natural disaster are not yet discovered, air-temperature, wind-speed, wind-direction, and air-pressure that constitute a natural disaster, such as tornado, are all caused by physical states [9]. It means that the condition of a tornado is generally determined by particular physical patterns. So, time-series of several physical values of air-temperature, wind-speed, wind-direction, and air-pressure will determine particular tornado condition.

Multidimensional Hierarchical Graph Neuron (mHGN) has been proven to be capable of working as a pattern recognizer. The latest architecture to prove its capability was the one that uses five-dimension 5X5X5X15X15 neurons. The architecture has been tested to recognize 26 patterns of five-dimensional alphabetical figures. Despite of 10% of distortion of the figures, the architecture was able to recognize in average more than 90% of those distorted patterns. This experiment result is a positive indication that mHGN has a potential to be developed as a disaster forecaster. The architecture can then be used as an additional tool for reducing the number of damages, losses, and costs when a tornado strikes.

2 Natural Disaster Forecast

Several countries have faced natural disasters more than the others. Every year, the US suffers from tornado more than other countries do. The number of tornados occurred within a year varies, so is the severity of damages, losses, and costs. However, this does not mean that only the US must concern with the occurrence of tornados. When the circumstances of developing a tornado in an area have reached, it is very likely that the area will be hit by a tornado. The likelihood of the occurrence of a natural disaster varies, but the possibility is still there in most parts of the world. For instance, the tsunami that hit Indonesia in 2004 had never been experienced by Indonesians for hundreds of years. This situation applies for other natural disasters.

Many countries under the coordination of the United Nation's UNISDR have worked together to handle natural disasters around the world. This means that any disaster that strikes a country is no longer the concern of the country itself, but it is automatically

the world's concern. Such situation has helped researchers in gaining data from various sensors spread around the world. Many researchers have started investigating new approach in forecasting natural disasters. Several issues related to this need to be discussed further.

The randomness of the occurrence of a natural disaster is not only in terms of the location, but also of the time and the severity. Two obvious examples are the tsunami in Indonesia in 2004 and the blizzard in Afghanistan in 2008. A number of researchers in opinion that the severity and the average magnitude of natural disasters have increased since the last decade. However, it is still not clear how severe future natural disasters might be. The impossibility to measure, or to predict the severity of natural disasters, that potentially will occur in the future, has been the major cause of the difficulties in anticipating their occurrences. Several other researchers have come up with the idea that, one way to deal with the randomness of the occurrence of natural disasters is through a disaster forecaster.

Several researchers have investigated natural disaster forecasting through a kind of early warning system [3] and now-casting [1, 4, 6, 7, 9, 10]. The forecasting approach that they [2, 5, 8] have attempted is able to forecast the disaster within one hour time frame. SuzukiI, Michihiro, and Honma [8] have shown their success in predicting Haneda's gust wind disaster of April 18, 2008. Despite the difficulties in finding appropriate equations, Sorensen [3] admits that his early warning system has been built utilizing a number of fields of science. He [3] further advises that early warning systems will be effective if they integrate the subsystems of detection of extreme events, management of hazard information, and public response.

It seems to be that researchers have tried to find an appropriate approach for working on three areas: natural disaster forecaster, now-casters, or early warning systems. However, they [3, 9, 11] also still integrate their system with disaster management systems. Even Doong et al. [11] suggest that the success of a disaster mitigation concept lies in the quality of the disaster management. This shows that their approach alone is not yet adequate to handle natural disasters. The potential reason to this case is the fact that a system for handling natural disaster requires very complicated mathematical analysis. So many parameters and values need to be considered and included in their calculation [1, 5, 6], and it is time consuming [10], but the system must run fast [1]. The other thing that needs to be considered when deploying such systems is the cost of using high quality sensors [5]. This causes the condition that gaining important measured data in several important areas is challenging [5].

Despite those efforts of researchers, Sorensen [3] argues that in terms of prediction and forecasting, no radical breakthroughs have occurred in the past twenty years. Most natural disaster researchers are working on current technologies that are not focusing on the forecasting techniques. Rather, they are concerned with how natural disaster alerts can be disseminated to the public [3]. While investigating natural disaster issues, special attention has been taken for people with disabilities. Most difficult part in facing a natural disaster is about how to handle people when a natural disaster occurs. Additionally, most common recommendation for an early warning system is "how to evacuate."

Although the randomness of the occurrences of a natural disaster has caused difficulties in handling it, the development of every natural disaster still follows natural science characteristics and rules. Each part of a natural disaster—for example a tornado—owns specific patterns and characters. For instance, a tornado develops its twist through hot and cold winds that move from the opposite directions. Not only the opposite winds play a role in developing a tornado, specific air pressure and air temperature are also significant contributors for a tornado's development.

The steps that a tornado builds its strong winding wind can be treated as a pattern. So, the recorded data from previous tornado disasters must be kept properly. The data is the important source of clue for researchers to analyse the pattern of a tornado. When patterns of tornados can be recorded, it is a strong possibility that when one of the patterns turns up, a system that can recognize patterns can be used to recognize a tornado early before it becomes a strong and destructive one. Such patterns are the most important part of mHGN for forecasting tornados hours before they strike.

3   Multidimensional Hierarchical Graph Neuron (mHGN)

The need to solve multidimensional problems has been discussed since a long time ago. People are aware that to handle complex problems, values taken from numerous dimensions must be considered and calculated. Otherwise, the result that comes up after the calculation analysing just a few parameters cannot be considered correct. In most cases, such a condition has produced very high false positive and true negative error rate. Another issue related to solving multidimensional problems is the solving method that will be implemented. In a complex system, not only the number of dimensions is large, but how all the dimensions are interrelated to each other, or independent on one another, is often not clear.

Natural disaster system is a good example as a multidimensional system. Therefore, forecasting natural disasters is also a type of solving a multidimensional problem. Not only the location or the latitude determines the condition of natural disasters, air-temperature, air-pressure, air-humidity, wind-direction, and wind-speed also play a big role in causing natural disasters of tornados. A problem that still exists is the interdependency amongst those tangible and intangible values (industrial development, people movement, etc.). It is cto figure out a formula that constitutes such interdependency. This is a strong indication that such multidimensional problems may be solved using artificial intelligent approaches such as mHGN.

3.1   Experiment Results

For the experiment, each GN is operated by a thread. Various 2D-, 3D-, 4D- and 5D-pattern recognition have been scrutinized. The compositions used in the experiment are: 15X15 mHGN, 5X15X15 mHGN, 5X5X15X15 mHGN, and 5X5X5X15X15 mHGN respectively. For instance, in the 15X15 pattern recognition the composition requires: $225 + 195 + 165 + 135 + + 105 + 75 + 45 + 15 + 13 + 11 + 9 + 7 + 5 + 3 + 1 = 1009$ neurons per value of data. As for creating patterns, binary data is used, then two values

(i.e. 0 and 1) of data are required. Therefore, 2018 neurons are deployed in the 15X15 mHGN composition. So, 2018 threads have been run in parallel during this 2D pattern recognition. By using threads, the activity of neurons is simulated so that the functionalities are close to the real neuron functionalities.

The experiment has worked on all the patterns of 26 alphabetical figures. Following the composition of the neurons, the alphabet patterns consist of 15X15 pixels. For the training purpose, the mHGN is first fed one-cycle with all the 26 non-distorted patterns. The order of the patterns during the training phase has been determined randomly. Then, to acquire the recognition results the mHGN is fed with a lot of randomly distorted patterns of alphabets. The recognizing accuracy is taken by calculating the average value of the results.

For the sake of the experiment, 20 distorted patterns for each alphabetical figure have been prepared. After acquiring the results, the experiment is repeated 10 times with the same steps, but each time the mHGN is trained with 26 patterns of alphabetical figures with randomly different order. So, for each alphabetical figure for particular percentage of distortion, in total 200 distorted patterns have been prepared as testing patterns.

There are 7 levels of distortion that have been tested, they are: 1.3%, 2.7%, 4.4%, 6.7%, 8.0%, 8.9%, and 10.7%. These levels have been so chosen based on the number of distorted pixels. The sizes of pixels represent the factor and the non-factor of the dimension of the patterns. By doing so, we can observe all the possibilities of distortion. So, in total there are 5200 (26 x 20 x 10) randomly distorted testing patterns. The following Figure 1 shows 5 samples of different orders of the patterns:

1	E	N	W	L	I	S	P	G	H	J	D	Y	A	X	Q	R	C	M	F	V	O	T	U	K	Z	B
2	R	P	J	S	O	Q	D	V	C	K	L	E	F	G	X	Y	A	T	Z	B	U	W	T	H	M	N
3	G	B	H	R	Z	C	I	Y	X	S	J	K	D	A	N	T	Q	V	E	W	F	U	P	O	L	M
4	L	N	I	F	R	X	B	K	O	C	T	Z	A	Y	G	V	U	H	P	J	Q	S	W	E	D	M
5	C	E	T	U	N	R	H	Y	G	D	B	K	F	M	I	X	V	S	Q	J	Z	W	O	A	L	P

Fig. 1. Five different randomly ordered alphabets.

The following shows some results taken from testing 4.4% randomly distorted patterns, and the mHGN was previously stored with alphabetical figure patterns, and the order was IEFXMQYJHPDKTORZCUALBGVWNS. The value on the right side of each alphabet show the portion (percentage) of the pattern that is recognizable as the corresponding alphabet (see Fig. 2.).

The following shows 10 samples of distorted patterns of the alphabetical figure of "A" taken from the experiment t of recognizing 5.8% randomly distorted patterns (see Fig. 3.).

After collecting the results taken from testing 5200 patterns we can summarize how accurate the mHGN is, in recognizing different levels of distortion of 26 alphabets. The summary is taken based on the average accuracy values from all the steps. The following shows the summarized result taken from testing distorted patterns using five-dimensional 5X5X5X15X15 mHGN (see Fig. 4.).

Patterns Stored	Distorted Pattern	Recognised patterns and their recognized portion (%) from 20 different randomly distorted patterns																				Recognised Correctly
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
I	A	A 9	A 8	A 8	A 54	A 13	A 3	A 8	A 13	A 16	A 6	A 15	A 14	A 15	A 14	A 2	A 1	A 54	A 17	A 15		20
E	B	B 3	B 4	B 36	B 37	B 11	B 36	B 11	B 11	B 37	B 10	B 5	B 36	B 13	B 10	B 12	B 11	B 6	B 9	B 11	B 36	20
F	C	C 40	C 7	C 38	C 15	C 8	C 39	C 8	C 7	C 8	C 15	C 7	C 0	C 14	C 15	C 40	C 14	C 1	C 14	C 39		20
X	D	D 26	D 10	D 4	D 27	D 9	D 10	D 10	D 2	D 26	D 12	D 25	D 11	D 5	D 12	D 6	D 9	D 9	D 10	D 11	D 10	20
M	E	E 10	E 6	E 30	E 29	E 12	E 11	E 11	E 10	E 10	E 29	E 11	E 10	E 10	E 11	E 12	E 29	E 31	E 6	E 10	E 9	20
Q	F	F 20	F 4	F 19	F 3	F 20	F 9	F 7	F 8	F 4	F 1	F 7	F 21	F 19	F 22	F 8	F 9	F 21	F 8	F 4	F 8	19
Y	G	G 51	G 14	G 51	G 13	G 15	G 51	G 14	G 50	G 7	G 51	G 14	G 15	G 6	G 15	G 15	G 7	G 3	G 14	G 6	G 7	20
J	H	H 8	H 14	H 54	H 14	H 15	H 56	H 15	H 14	H 7	H 54	H 13	H 3	H 16	H 55	H 14	H 54	H 9	H 15	H 55	H 54	20
H	I	I 54	I 6	I 1	I 53	I 6	I 14	I 14	I 55	I 15	I 15	I 54	I 14	I 14	I 15	I 54	I 6	I 15	I 14	I 16	I 5	20
P	J	J 54	J 5	J 5	J 14	J 15	J 13	J 54	J 14	J 55	J 6	J 55	J 15	J 15	J 6	J 54	J 55	J 14	J 13	J 14	J 3	20
D	K	K 4	K 7	K 8	K 8	K 6	K 6	K 8	K 5	K 8	H 22	K 9	K 6	K 7	K 6	K 4	K 2	H 22	K 9	K 7	K 5	18
K	L	L 9	L 11	L 5	L 38	L 11	L 11	L 11	L 39	L 6	L 40	L 12	L 39	L 10	L 6	L 5	L 10	L 7	L 10	L 12	L 38	20
T	M	M 22	M 19	M 21	M 4	M 21	M 22	M 8	M 4	M 8	M 6	M 7	M 6	M 5	M 7	M 5	M 4	M 7	M 7	M 6	M 7	20
N	N	N 19	N 4	N 3	N 7	H 19	N 8	N 8	N 4	N 7	N 6	N 7	N 8	H 18	N 8	N 6	N 7	N 8	N 5	N 7	N 5	18
R	O	O 54	O 16	O 15	O 14	O 14	O 7	O 56	O 14	O 54	O 8	O 3	O 16	O 14	O 16	O 55	O 54	O 6	O 15	O 7	O 54	20
Z	P	P 10	P 32	P 10	P 9	P 9	P 10	P 5	P 6	P 5	P 33	P 10	P 0	P 8	P 10	P 10	P 32	P 5	P 9	P 11	P 11	20
C	Q	Q 44	Q 15	Q 4	Q 15	Q 45	Q 15	Q 16	Q 13	Q 14	Q 13	Q 6	Q 14	Q 15	Q 15	Q 15	Q 45	Q 6	Q 44	Q 14	Q 45	20
U	R	R 11	R 12	R 12	R 5	R 12	R 11	R 38	R 5	R 6	R 10	R 7	R 37	R 10	R 37	R 37	R 38	R 11	R 2	R 12		20
A	S	S 14	S 15	S 8	S 14	S 13	S 13	S 3	S 7	S 13	S 14	S 46	S 45	S 13	S 15	S 46	S 7	S 7	S 13	S 14	S 15	20
L	T	T 14	T 15	T 14	T 15	T 15	T 16	T 13	T 7	T 15	T 55	T 15	T 14	T 15	T 55	T 16	T 54	T 55	T 54	T 14	T 14	20
B	U	U 4	U 7	U 10	U 7	U 10	U 11	U 11	U 9	U 11	U 5	U 30	U 11	U 5	U 30	U 4	U 10	U 10	U 10	U 11	U 2	20
G	V	V 56	V 14	V 54	V 15	V 54	V 8	V 55	V 55	V 6	V 14	V 14	V 9	V 2	V 15	V 1	V 56	V 16	V 14	V 16	V 15	20
V	W	W 6	W 14	W 30	W 29	W 13	W 13	W 13	W 3	W 14	W 6	W 12	W 14	W 11	W 13	W 13	W 7	W 30	W 30	W 11	W 12	20
W	X	X 15	X 53	X 16	X 15	X 15	X 14	X 15	X 14	X 14	X 14	X 7	X 6	X 16	X 54	X 15	X 14	X 54	X 14	X 13	X 54	20
N	Y	Y 15	Y 16	Y 15	Y 13	Y 14	Y 51	Y 8	Y 2	Y 51	Y 13	Y 51	Y 15	Y 14	Y 13	Y 52	Y 16	Y 15	Y 15	Y 5	Y 15	20
S	Z	Z 7	Z 15	Z 15	Z 14	Z 13	Z 14	Z 16	Z 17	Z 0	Z 51	Z 14	Z 51	Z 50	Z 7	Z 6	Z 7	Z 51	Z 14	Z 13	Z 51	20

(Left margin label spanning rows: Order Type 0)

Fig. 2. The result of al the 26 alphabetical patterns that are twenty times 4.4% randomly distorted.
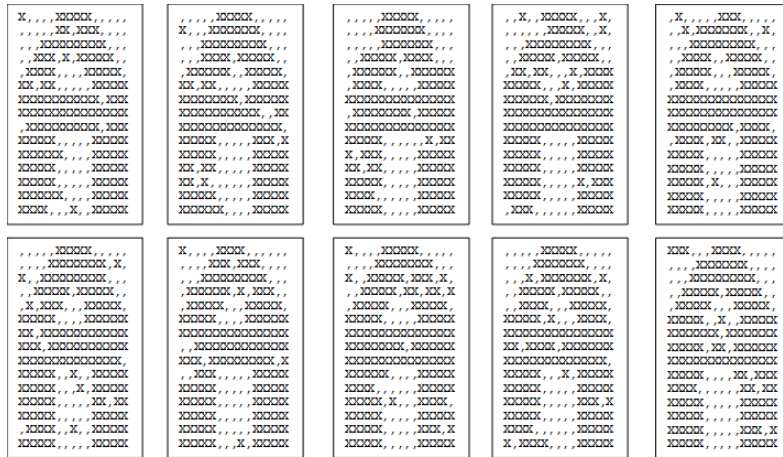

Fig. 3. Ten different randomly 5.8% distorted patterns of alphabetical figure of "A"

It can be seen from Figure 4 in the last column that the mHGN is able to recognize 91% of the 10.7% distorted patterns of 26 alphabetical figures. Some alphabetical figures of A, C, E, G, I, J, L, O, S, T, U, V, X, Y, Z, are even 100% recognizable. Other patterns of alphabetical figures of H, K, M, N, are not very well recognized because they are visually and physically very similar. In fact, if this architecture is used to recognize different states of the same alphabet, such as regular-A, bold-A, and italic-A as the same alphabet, then mHGN will be able to gain better accuracy values.

The following figure shows the differences of recognition accuracy amongst 15X15, 5X15X15, 5X5X15X15, and 5X5X5X15X15 mHGN architectures when recognizing 10.7% distorted patterns of alphabets (see Fig. 5.).

5X5X5X15X15 Patterns		Distortion (%)						
		1.3	2.7	4.4	6.7	8.0	8.9	10.7
Recognition Accuracy for Each Pattern (%)	A	100	100	100	100	100	100	100
	B	100	100	100	100	98	97	94
	C	100	100	100	100	100	96	100
	D	100	100	100	100	100	100	98
	E	100	100	100	100	100	100	100
	F	100	99	94	89	83	85	74
	G	100	100	100	100	100	100	100
	H	100	100	89	67	48	50	55
	I	100	100	100	100	100	100	100
	J	100	100	100	100	100	100	100
	K	100	100	98	81	70	72	67
	L	100	100	100	100	100	100	100
	M	100	100	93	76	55	66	49
	N	100	100	97	77	63	60	55
	O	100	100	100	100	100	100	100
	P	100	99	87	79	80	81	81
	Q	100	100	100	100	100	94	99
	R	100	100	100	95	100	99	95
	S	100	100	100	100	100	100	100
	T	100	100	100	100	100	100	100
	U	100	100	100	100	100	100	100
	V	100	100	100	100	100	100	100
	W	100	100	100	100	99	98	92
	X	100	100	100	100	100	100	100
	Y	100	100	100	100	100	100	100
	Z	100	100	100	100	100	100	100
Average		100	100	98	95	92	92	91

Fig. 4. The summary of the result using 5X5X5X15X15 mHGN [12].

Comparison Result		Distortion = 10.7 %			
		15X15	5X15X15	5X5X15X15	5X5X5X15X15
Recognition Accuracy for Each Pattern (%)	A	99	100	100	100
	B	58	69	92	94
	C	67	93	94	100
	D	78	92	94	98
	E	85	80	100	100
	F	61	71	81	74
	G	87	98	100	100
	H	23	63	69	55
	I	95	100	100	100
	J	77	95	100	100
	K	68	59	84	67
	L	50	80	100	100
	M	38	36	35	49
	N	53	42	63	55
	O	100	100	100	100
	P	61	59	75	81
	Q	63	73	73	99
	R	79	90	95	95
	S	78	97	100	100
	T	93	95	100	100
	U	89	84	85	100
	V	100	100	100	100
	W	75	82	98	92
	X	85	100	100	100
	Y	100	100	100	100
	Z	99	100	100	100
Average		75	83	90	91

Fig. 5. Differences of recognition accuracy amongst four different architectures

3.2   Time-Series in Pattern Recognition

Recognizing patterns of time series problem utilizes data that have previously been recorded regularly in timely manner [12]. For instance, if the parameter that needs to be recorded is a single value, and the recording step is every six hours, then there will be 4 values recorded every day. In order to constructs the recorded values as a pattern, the data representation of the recorded values need to be developed so, that they can fit into a pattern recognition architecture. The following figure 6 shows two ways of representing recorded data for 8 levels of measurement.

0	00000000	0	00000000
1	10000000	1	00000001
2	11000000	2	00000011
3	11100000	3	00000111
4	11110000	4	00001111
5	11111000	5	00011111
6	11111100	6	00111111
7	11111110	7	01111111
8	11111111	8	11111111

Fig. 6. Two examples of data representation for 8-bit value

It can be seen from **Error! Reference source not found.** that the data is represented using binary values. The bit difference (distance) between adjacent levels is 1. Additionally, the number of bit differences between any two levels is linear with the value difference between the two levels. However, such data representation will not maximally utilize the binary combination. With 3-bit data, only 3/8 or 0.375 is the occupation rate. For 4-bit data is the occupation rate 4/16 or 0.25. The occupation rate is 5/32 or 0.15625 for 5-bit value. This shows that the above data representation will produce less occupation rate, the more bits is used. This is an indication that due to such an occupation rate the pattern recognizer will have less recognition accuracy the more bits in it is used. The following is a better data representation.

In **Error! Reference source not found.** it is shown that the number of bit differences between adjacent levels is 1. Between any two levels the bit difference is 2, and 3 between any three levels. This data representation is cyclic. It means that, if it is required the order of binary representation can be modified circularly without affecting the bit differences (distances). Using such a better data representation, for any bit data is the occupation rate constantly 0.75. With such a constant occupation rate the pattern recognizer will have constant recognition accuracy, any number of bits in it is used. The following figure shows an example of recorded data taken from a single value measurement and each value has 8 levels.

It can be seen from **Error! Reference source not found.** that the recorded values from parameter of 8 levels data construct a two-dimensional pattern of 30X8 architecture. Utilizing these recorded data, the pattern recognizer can forecast a tornado 6 hours earlier. when the same tornado will occur again. It means that if values have been recorded and the same pattern is recognized by the pattern recognizer, then the tornado is forecasted to occur again within 6-hour time.

So, to predict what will occur in 6-hour time using 30X8 mHGN architecture, the recognizer need to be fed with data measurement recorded from 7 days and 6 hours ago

until now. Not only forecasting something that will occur in 6-hour time, the 30X8 mHGN architecture can also be used to forecast something that will occur in 12-hour time. But, for this purpose the recognizer is fed with data measurement recorded from 7 days only. In this case, the pattern is not fed with 30X8 binary data, but with only 29X8 binary data. This is the same case when a pattern recognizer is fed with incomplete data (only 97% data), but the recognizer still has the capability to recognize the pattern. Similarly, to forecast something that will occur in 18-hour time, the recognizer is fed with data measurement recorded from 6 days and 18 hours ago (only 93% data). This case is shown in **Error! Reference source not found.**, that after stored with 26 patterns, 5X5X5X15X15 mHGN architecture is able to recognize 89% incomplete/distorted patterns with 91% of successful rate.

3-bit		4-bit		5-bit	
1	010	1	0010	1	10001
2	011	2	0011	2	10000
3	001	3	0001	3	10010
4	101	4	0101	4	00010
5	100	5	0100	5	00011
6	110	6	0110	6	00001
		7	1110	7	00101
		8	1111	8	00100
		9	1101	9	00110
		10	1001	10	01110
		11	1000	11	01111
		12	1010	12	01101
				13	01001
				14	01000
				15	01010
				16	11010
				17	11011
				18	11001
				19	11101
				20	11100
				21	11110
				22	10110
				23	10111
				24	10101

Fig. 7. Three examples of a better data representation for 3-, 4-, and 5-bit value

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1
0 0 0 0 1 1 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 0 0 0 1 1 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

2    1    1    1    3    3    2    1    1    2    2    3    4    5    5    6    7    8    7    6    5    5    4    3    3    3    4    5    5    4
18.00 00.00 06.00 12.00 18.00 00.00 06.00 12.00 18.00 00.00 06.00 12.00 18.00 00.00 06.00 12.00 18.00 00.00 06.00 12.00 18.00 00.00 06.00 12.00 18.00 00.00 06.00 12.00 18.00 00.00
```
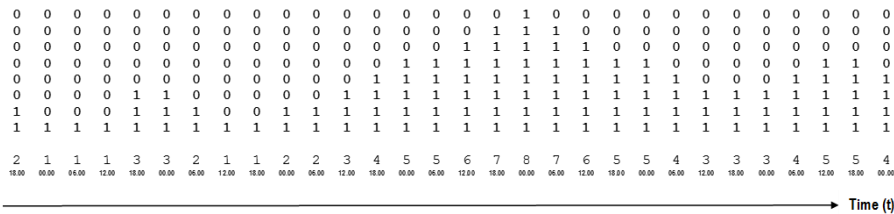
Time (t)

Fig. 8. Data of 8 level value build a 2D-Pattern

4   Multidimensional Graph Neuron for Tornado Forecasting

In the previous section, time series value is described and represented so, that it can be forecasted through utilizing a pattern recognition, such as mHGN. In case of tornado forecasting, single parameter in a location, such as air-pressure, is not the only value that determine the occurrence of a tornado in the location within 6-hour time. Several other parameters, such as air-temperature, wind-speed, wind-direction, and air-humidity, play a big role in the occurrences as well. It means that the number of levels or a measured value will increase according to the number of parameters. In case 5 parameters need to be measured and each value contains 8 levels, the required pattern structure would be 30X40.

Also described in the previous section that measuring a parameter at particular point of location for several periods of time will generate a two dimensional pattern. If a series of points of the location need to be measured for several period of time, then the measured values will become a three dimensional pattern. The following figure 9 depicts how some part of it will look like.
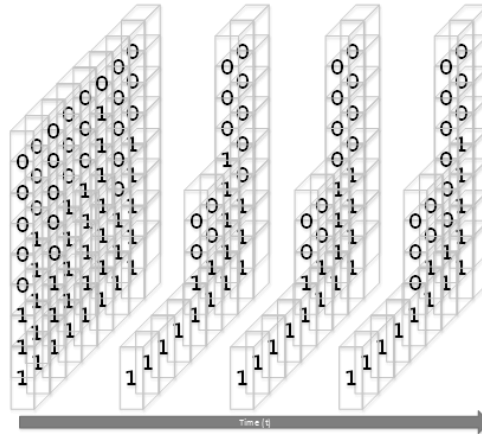
Fig. 9. A row of data of 8 level value build a 3D-Pattern

Also described in the previous section that measuring parameters at particular point of location for several periods of time will generate a two dimensional pattern. If a series and linear of locations need to be measured for several periods of time, then the measured values will become a three dimensional pattern. If the location that need to be measured is an 2D area, then the measured values will generate a 4D pattern. Furthermore, if the location that need to be measured is a 3D area, then the measured values will generate a 5D pattern.

4.1 The Architecture of mHGN for Time-Series Tornado Data

The utilization of mHGN has introduced a new approach that a local tornado forecast can be operated using small and cheap components. The values of air-temperature, air-humidity, air-pressure, wind-speed, and wind-direction can be gained through ordinary sensors. The area that is covered by those sensors can be a 3D area, because such small sensors can be easily mounted in valleys or hills, or even vehicles. The sensors can be embedded in a tiny computer, such as Raspberry Pi. The tiny computer will be responsible to run several GNs. The values taken from the sensors will then be worked out within the GNs. The connectivity of neurons is developed within a tiny computer and through the interconnectivity of the tiny computers.

During mHGN experiments, each neuron and its functionalities is operated by a thread. However, the number of thread will be tremendous, especially when the mHGN is used to work on multidimensional patterns. For example, 15X15 architecture of mHGN requires 2018 neurons. This means that the number of threads that need to be run is also 2018. Such a number of threads would be difficult to be run if the computer used for the project is a Raspberry Pi. The new approach to run neurons is through utilizing threads in which the number of threads is only the same as the size of neurons on the base level. The following figure 10 shows that instead of utilizing 25 threads the new approach to implement mHGN architecture only requires 9 threads.
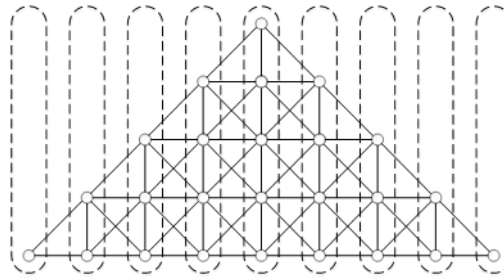
Fig. 10. The number of threads (dashed line) is the same as the neuron size on the base level

In short, to build a tornado forecast for particular location, five parameters need to be measured. They are: air-temperature, air-humidity, wind-speed, wind-direction, and air-pressure. So, if one parameter is represented through 8-bit binary data, then for the measurement of 5 parameters 41-bit data is needed (the dimension must be odd number). For the time series, 21 series of measurement will be carried out. For an area that needs to be protected by mHGN, 3X3X3 measurement points will be deployed. So, the mHGN dimension will be 3X3X3X41X21.

The positions of the 3X3X3 GNs will form a cylinder shape. In the cylinder, there will be three layers of circles. Each layer contains 9 GNs, in which 8 GNs will be on the border of the circle, and one GN will be located in the centre of the circle. The following figure 11 shows the architecture of the positions of the sensors.

The cylinder shape of the architecture has been chosen so, that mHGN still has an ability to recognize the same tornado pattern but developed with the direction different

from the ones already stored. For the purpose of training, patterns from the previous tornados will be stored in the mHGN. Each pattern of a tornado will then be stored in mHGN eight times, following the number of eight major compass directions. It will look like as if the mHGN has stored 8 patterns of tornados. By having eight patterns for each tornado stored in mHGN, whenever the same characteristics of a tornado turn up but from different direction from the already stored ones, mHGN will be able to recognize it.
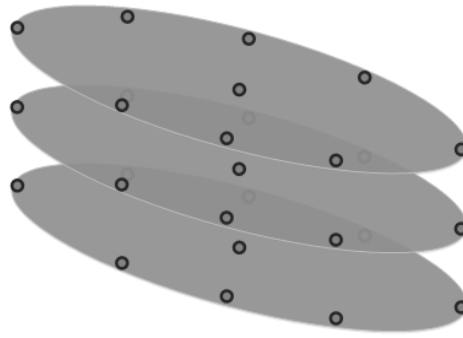
Fig. 11. The architecture of 3X3X3 sensors

4.2   Case Studies: Joplin's and Hackleburg–Phil Campbell's Tornados

Two deadliest tornados occurred quite recently are the tornado that struck Joplin, Missouri on May 22, 2011 and the one in Hackleburg–Phil Campbell, Alabama on April 27, 2011. To store the circumstances, several parameters in these areas need to be stored in mHGN. Fortunately, the National Oceanic and Atmospheric Administration (NOAA) provides lots of data of: air-temperature, air-humidity, air-pressure, wind-speed, wind-direction in most areas of the US. These data will be the major source for mHGN to store previous occurrences of tornados. In the case of Joplin, the following are several locations of stations that have recorded those data from their sensors including the map in the state of Missouri (see Fig. 12 and Fig. 13).

1	Boonville	10	Excelsior Springs
2	Branson	11	Farmington
3	Branson West	12	Fort Leonard Wood
4	Camdenton	13	Hannibal
5	Cape Girardeu	14	Jefferson City
6	Chillicothe	15	CRN: Joplin
7	CRN: Chillicothe	16	Joplin
8	Clinton	17	Kaiser Lake Ozark
9	Columbia		

Fig. 12. Several weather stations in the State of Missouri

In order to collect the suitable data that fit with the architecture of mHGN, the locations of the chosen weather stations that the data will be taken from, must build a figure like a circle, and the middle weather station must be located in the area in which a tornado has hit. The following is an excerpt of the data taken from a weather station Joplin in the State of Missouri (see Fig. 14).
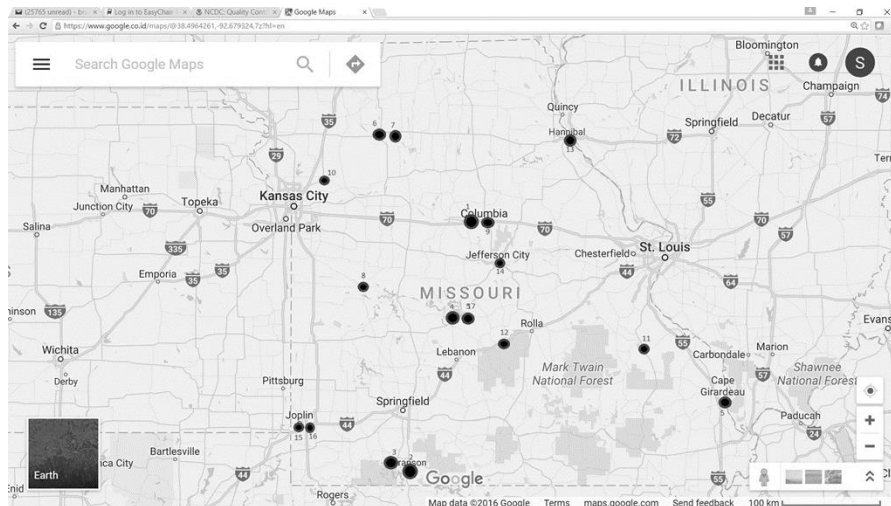

Fig. 13. The locations (bold circles) of several weather stations in the State of Missouri

Fig. 14. An excerpt of the data taken from a weather station in the State of Missouri

5   Discussion

As is the case with pattern recognition of alphabets, patterns are more or less different to one another. However, in time series measurement data patterns, which are constructed from the measured values of the sensors, can be very similar to one another. Therefore, data representation of measured values before data is fed to the architecture of mHGN plays a big role in having very accurate results. False positive and true negative rate will also be indications to determine the quality of mHGN in forecasting natural disastesr.

The data that will be used to validate this work will be the data taken from different cities and different countries. As mHGN is trained one-cycle only, it is a challenge to choose which data is the right data for the training purpose, or the data is the consolidated data from a number of occurrences. When the appropriate training data has been applied, mHGN will then have a capability to forecast the tornado

6   Conclusion

From the experiment results it is shown that mHGN has the capability to recognize multidimensional patterns. For simulating a tornado forecast, we have presented results of up to 5D architecture. As already discussed in [13] and [14] there is no modification required if the architecture needs to be extended to bigger sizes of patterns. In the future this capability will be improved to the extent so, that multi oriented of multidimensional patterns will also be recognizable. At this stage it is also observed that mHGN still use a single cycle memorization and recall operation. The scheme still utilizes small response time that is insensitive to the increases in the number of stored patterns.

References

1. Cheng, R. T., Smith, R. E.: A Nowcast Model for Tides and Tidal Currents in San Francisco Bay, California. In: Ocean Community Conference, Baltimore, USA (1998).
2. Ogasawara, J., Tanimoto, K., Imaichi, O., Yoshimoto, M.: Disaster Prevention and Response Support Solutions. Hitachi Review 63 (1), 236-243 (2014).
3. Sorensen, J. H.: Hazard Warning Systems: Review of 20 Years of Progress. Natural Hazards Review, pp. 119-125 (May 2000).
4. Kirschbaum, D. B., Adler, R., Hong, Y., Kumar, S., Peters-Lidard, C., Lerner-Lam, A.: Advances in landslide nowcasting: evaluation of a global and regional modeling approach. Environment Earth Science 1 (1), 1-14 (2011).
5. Bellaire, S., Jamieson, B.: Nowcast with a Forecast – Snow Cover Simulations on Slopes. Journal of Something 1 (1), 1-7 (2012).
6. Li, P., Lai, E.: Applications of Radar-based Nowcasting Techniques for Mesoscale Weather Forecasting in Hong Kong. Meteorological Applications 11 (1), 253-264 (2004).
7. Roy, S. S., Lakshmanan, V., Bhowmik, S. R., Thampi, S.: Doppler Radar based Nowcasting of Cyclone Ogni. Journal of Earth System Science 119 (2), 183-199 (2009).
8. SuzukiI, Y., Michihiro, Y., Honma, M.: Development of Strong Wind Nowcasting System. In: International Symposium on Extreme Weather and Cities, Tokyo, Japan (2013).

9. Winterrath, T., Rosenow, W.: The Radar- based Precipitation Nowcasting System RADVOR of Deutscher Wetterdienst for the support of meteorological and hydrological alert systems. In: 8th European Conference on Radar in Meteorology and Hydrology, Germany (2014).

10. Kryvasheyeu, Y., Chen, H., Obradovich, N., Moro, E., Hentenryck, P. V., Fowler, J.: Nowcasting Disaster Damage. In: arXiv preprint arXiv:1504.06827 (2015).

11. Doong, D. J., Chuang, L. Z. H., Wu, L. C., Fan, Y. M., Kao, C. C., Wang, J. H.: Development of An Operational Coastal Flooding Early Warning System. Natural Hazards and Earth System Sciences 12 (1), 379–390 (2012).

12. Nasution, B. B., Sembiring, R. W., Sundawa, B. V., Gunawan, Amelia, A., Ismael, Sunjaya, H., Alifuddin, S., Pardede, M., Junaidi, Syahruddin M., Lubis, Z.: Realtime Weather Forecasting using Multidimenssional Hierarchical Graph Neuron (mHGN). In: The 16th International Conference on Neural Networks (NN '15), Rome, Italy (2015).

13. Nasution, B. B., Khan, A. I.: A Hierarchical Graph Neuron Scheme for Real-time Pattern Recognition. IEEE Transactions on Neural Networks, pp. 212-229 (2008).

14. Nasution, B. B.: Towards Real Time Multidimensional Hierarchical Graph Neuron (mHGN). In: The 2nd International Conference on Computer and Information Sciences 2014 (ICCOINS 2014), Kuala Lumpur, Malaysia (2014).

15. Anderson, K.: Predicting the Weather: Victorians and the Science of Meteorology, The University of Chicago Press, Chicago and London (2005).

16. Duchon, C., Hale, R.: Time Series Analysis in Meteorology and Climatology: An Introduction, John Wiley & Sons Ltd, Oxford, UK (2012).

17. Gluhovsky, A.: Subsampling Methodology for the Analysis of Nonlinear Atmospheric Time Series. Lecture Notes in Earth Sciences 1 (1), 3-16 (2000).

18. Inness, P., Dorling, S.: Operational Weather Forecasting. John Wiley & Sons Ltd, West Sussex, UK (2013).

19. Lynch, P.: The origins of computer weather prediction and climate modelling. Journal of Computational Physiscs 227 (2008), 3431–3444 (2007).

20. Miksovsky, J., Pisoft, P., Raidl, A.: Global Patterns of Nonlinearity in Real and GCM-Simulated Atmospheric Data. Lecture Notes in Earth Sciences 1 (1), 17-34 (2000).

21. Vasquez, T.: Weather Forecasting Handbook, Weather Graphics Technologies, Garland, USA (2002).

22. Percival, D. B.: Analysis of Geophysical Time Series Using Discrete Wavelet Transforms: An Overview. Lecture Notes in Earth Sciences 1 (1), 61-80 (2000).

23. Lutgens, F. K., Tarbuck, E. J.: The Atmosphere: An Introduction to Meteorology, Pearson, Glenview, USA (2013).

24. Nebeker, F.: Calculating the Weather: Meteorology in the 20th Century. Academic Press, San Diego, USA (1995).

25. Yorke, S.: Weather Forecasting Made Simple. Andrews UK Limited, Newbury, Berkshire, UK (2011).