

PROTOTIPE SISTEM PRAKIRAAN CUACA BERDASARKAN SUHU DAN KELEMBAPAN DENGAN METODE LOGIKA *FUZZY* DAN *BACKPROPAGATION* BERBASIS MIKROKONTROLER

Ratna Aisuwarya¹, Dodon Yendri, Werman Kasoep², Kiki Amelia³, Adi Arga Arifnur⁴

^{*1234}Jurusan Sistem Komputer, Universitas Andalas,
Jl. Kampus Unand Limau Manis, Padang 25163
E-mail: aisuwarya@fmipa.unand.ac.id

ABSTRAK

Penelitian ini bertujuan untuk membuat suatu prototipe sistem monitoring cuaca dengan menggunakan platform *Arduino*, *web* dan perangkat *mobile* berbasis *Android*. Terdapat dua sistem pengujian yang akan dibandingkan data keakuratannya. Pada sistem pertama, data dikumpulkan secara *realtime* dan ditampilkan pada laman *web* yang bisa diakses langsung. Identifikasi keakuratan data dengan prakiraan yang diinformasikan BMKG dilakukan dengan perhitungan menggunakan *toolbox fuzzy logic*. Pada sistem kedua, metode Jaringan Syaraf Tiruan (*JST*) *Backpropagation* digunakan sebagai algoritma pembelajaran untuk mendapatkan bobot yang sesuai. Data suhu dan kelembapan yang didapatkan dari BMKG Kota Padang digunakan sebagai data input yang di-*training* oleh *toolbox JST Backpropagation* pada Aplikasi Matlab. Bobot yang sesuai dari hasil *training JST* digunakan sebagai faktor penimbang pada proses *testing JST* di *Arduino*. Hasil proses *testing* mengindikasikan prakiraan cuaca yang dikirim langsung ke *handphone Android*. Hasil pengujian kedua sistem diperoleh tingkat keberhasilan prakiraan cuaca sebesar 60,94% dengan logika *Fuzzy* dan 78,6% dengan metode *JST Backpropagation*.

Kata kunci: Cuaca, Logika *Fuzzy*, *JST Backpropagation*, *Arduino*

ABSTRACT

This research aims to create a weather monitoring system prototype using the *Arduino* platform, *web* and *mobile* devices based on *Android*. There are two test systems that will be compared to data accuracy. In the first system, the data is collected in real-time and displayed on a *web* page that can be accessed directly. The identification accuracy of the data with forecasts that are informed by BMKG performed by calculation using *fuzzy logic toolbox*. In both systems, methods of Artificial Neural Network (*ANN*) *Backpropagation* is used as a learning algorithm to obtain the corresponding weights. Temperature and humidity data obtained from BMKG Padang City is used as input data in-training by *Backpropagation ANN toolbox* in Matlab Application. The corresponding weights of the *ANN* training results are used as a factor weighing on the testing process *JST* in *Arduino*. The results of the testing indicate that weather forecasts sent directly to *android* phone. The success rate of test results obtained by both systems are 60.94%, and 78,6% respectively for *Fuzzy logic* and *Backpropagation ANN* method.

Keywords : Weather Prediction, *Fuzzy Logic*, *ANN Backpropagation*, *Arduino*

PENDAHULUAN

Kota Padang adalah kota terbesar sekaligus ibu kota dari Provinsi Sumatera Barat yang berada pada ketinggian antara 0 sampai 1.853 m (meter) di atas permukaan laut. Kota Padang mempunyai iklim tropis dengan suhu udara yang cukup tinggi, yaitu antara 23 - 32 °C (Celsius) pada siang hari dan 22 - 28 °C pada

malam hari, dengan kelembapan berkisar antara 78 - 81%.^[1]

Suhu dan kelembapan adalah dua indikator yang sering digunakan untuk memperkirakan cuaca. Dua indikator ini dapat diukur dengan sensor DHT11. Dengan mengukur perubahan suhu dan kelembapan tersebut, dapat diperkirakan apakah cuaca di

Kota Padang dalam kondisi hujan, berawan, maupun cerah.

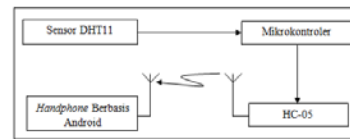
Berdasarkan permasalahan tersebut, maka penelitian ini dilakukan dengan 2 cara, yaitu pertama pendeteksian suhu dan kelembapan menggunakan sensor DHT11. Hasil pengukuran diunggah ke *website* Xively dan akan dibuat alat sistem monitoring cuaca yang dikombinasikan dengan mikrokontroler Arduino. Kedua, perangkat penampil hasil pengukuran berupa *handphone* berbasis Android sehingga menghasilkan sistem monitoring yang portabilitas. Dengan melakukan monitoring ini diharapkan dapat memberi informasi keadaan cuaca ke semua warga yang tinggal di Kota Padang sehingga ada antisipasi terhadap dampak kondisi lingkungan yang ekstrim.

Jaringan syaraf tiruan (JST) adalah sistem komputasi yang arsitektur dan operasinya diilhami dari pengetahuan tentang sel syaraf biologis di dalam otak. JST dengan layar tunggal memiliki keterbatasan dalam pengenalan pola. Kelemahan ini bisa diatasi dengan menggunakan *Backpropagation* yang memiliki satu/beberapa layar tersembunyi diantara layar masukan dan layar keluaran. Dengan adanya layar tersembunyi pada *Backpropagation* dapat menyebabkan tingkat *error* pada *Backpropagation* lebih kecil dibanding tingkat *error* pada jaringan layar tunggal. Hal inilah yang membuat JST semakin diminati orang apalagi setelah berhasil ditemukannya berbagai aplikasi yang dapat diselesaikan dengan *Backpropagation*.

METODE

Jenis penelitian yang digunakan adalah penelitian eksperimental, yaitu membuktikan hasil penelitian dengan cara melakukan percobaan. Gambar 1 menunjukkan cara kerja dari sistem yang akan dibuat. Sistem ini bekerja mulai dari pengukuran terhadap suhu dan kelembapan dengan menggunakan sensor DHT11. Data hasil pengukuran tersebut diteruskan ke mikrokontroler untuk diproses dengan metode *fuzzy* dan JST hasil *training* dari data BMKG. Kemudian mikrokontroler akan mengirimkan data digital hasil pemrosesannya ke *handphone* berbasis Android melalui komunikasi serial *Bluetooth* dan menampilkan hasilnya secara *realtime* melalui web. Aplikasi Android menampilkan data hasil pengukuran

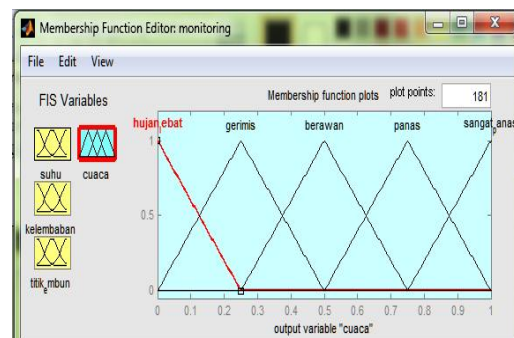
suhu dan kelembapan serta prakiraan cuaca sebagai hasil keluaran yang telah diolah oleh mikrokontroler.



Gambar 1. Diagram Blok Sistem

1. Perancangan Rule Fuzzy

Fuzzy rule yang digunakan dalam sistem monitoring sebagai bentuk penarikan hasil kesimpulan dengan input suhu, kelembapan, dan titik embun lebih sangat sederhana, sistem akan langsung mendapatkan nilai sebagai input dikarenakan input hanya mendapatkan satu variabel masukan yaitu tingkat *range* yang dibaca sensor DHT11, tingkat *range* yang direpresentasikan dengan 5 himpunan *fuzzy* dengan 3 himpunan seperti pada gambar 2, yaitu Suhu; Kelembapan; dan Titik Embun. Output data himpunan terdiri dari 5 output yaitu; Hujan lebat, Gerimis, Berawan, Panas, dan Sangat panas.



Gambar 2. Fungsi Keanggotaan Fuzzy

Jika input *Hujan lebat*, maka *fuzzy* input akan bernilai >0 , $\text{input} \leq 0,25$; jika input *Gerimis*, maka *fuzzy* input akan bernilai $>0,25$, $\text{input} \leq 0,50$; jika input bernilai *Berawan* maka *fuzzy* input akan bernilai $>0,50$, $\text{input} \leq 0,75$; jika input bernilai *Panas* maka *fuzzy* input akan bernilai $>0,75$, $\text{input} \leq 1$; jika input bernilai *Sangat Panas* maka *fuzzy* input akan bernilai 1. Derajat keanggotaan dapat dilihat pada tabel 1.

If data1 hujanlebat then (input >0) and (input ≤ 0,25)

Else data1 gerimis then (input >0,25) and (input ≤ 0,50) Else data1 berawan then (input >0,50) and (input ≤0,75)

Else data1 panas then input (input >0,75) and (input ≤1) Else data1 sangat panas then input = 1

Himpunan *fuzzy* celcius akan memiliki domain [0-50°C] dengan derajat keanggotaan celcius “Normal” tertinggi (= 1) terletak pada nilai 50°C. Apabila celcius semakin kurang dari 40°C, maka rentang *celcius* akan semakin Sedang. Namun apabila input nilai *celcius* semakin melebihi 40°C maka rentang *celcius* akan semakin mendekati “Tinggi”. Himpunan *fuzzy Celcius* “Normal” direpresentasikan dengan fungsi keanggotaan segitiga dengan derajat keanggotaan semakin naik apabila kadar *Celcius* semakin mendekati 40°C.

Himpunan *fuzzy* kelembapan akan memiliki domain [20%-90%] dengan derajat keanggotaan kelembapan “Normal” tertinggi (= 1) terletak pada nilai 90%. Apabila kelembapan semakin kurang dari 80%, maka rentang kelembapan akan semakin Sedang, dan keluar dari semesta pembicaraan data penelitian. Namun apabila input nilai kelembapan semakin melebihi 80% maka rentang kelembapan akan semakin mendekati “Tinggi”. Himpunan *fuzzy* kelembapan “Normal” direpresentasikan dengan fungsi keanggotaan segitiga dengan derajat keanggotaan semakin naik apabila kadar kelembapan semakin mendekati 80%.

Himpunan *fuzzy* celcius akan memiliki domain [10°C -26°C] dengan derajat keanggotaan celcius “Normal” tertinggi (= 1) terletak pada nilai 26°C. Apabila celcius semakin kurang dari 24°C, maka rentang *celcius* akan semakin Sedang, dan keluar dari semesta pembicaraan data penelitian. Namun apabila input nilai *celcius* semakin melebihi 24°C maka rentang *celcius* akan semakin mendekati “Tinggi”. Himpunan *fuzzy Celcius* “Normal” direpresentasikan dengan fungsi keanggotaan segitiga dengan derajat keanggotaan semakin naik apabila kadar *Celcius* semakin mendekati 24°C.

Sistem menggunakan 5 tingkatan input data sensor *DHT11*, yang direpresentasikan ke dalam bentuk derajat keanggotaan *fuzzy* fungsi segitiga. Masing masing fungsi memiliki rentang nilai tersendiri. Variabel rentang input terbagi atas 5 pilihan himpunan *fuzzy* yaitu *Sangat Rendah*, *Rendah*, *Sedang*, *Tinggi* dan

Sangat Tinggi dan inputan yang diolah adalah 3 data, yaitu, suhu (*celcius*), kelembapan udara, dan titik embun. Inputan suhu fahrenheit tidak dipilih karena suhu yang biasa dikenali disini adalah suhu (*celcius*). Sehingga bilangan *fuzzy* untuk inputan *celcius* dapat ditampilkan

Tabel 1. Derajat Keanggotaan *Fuzzy*

Jenis cuaca	Bobot
<i>Hujan lebat</i>	input>0, input≤ 0,25
<i>Gerimis</i>	input>0,25 ,input≤ 0,50
berawan	input>0,50, input≤ 0,75
<i>Panas</i>	input>0,25, input≤ 1
Sangat Panas	input = 1

menjadi: *Sangat Rendah* = 0-10 °C; *Rendah* = 11-20 °C ; *Sedang* = 21-30 °C; *Tinggi* = 31-40 °C; dan *Sangat Tinggi* = 41-50 °C. Sesuai dengan perhitungan data akurasi sensor DHT11.

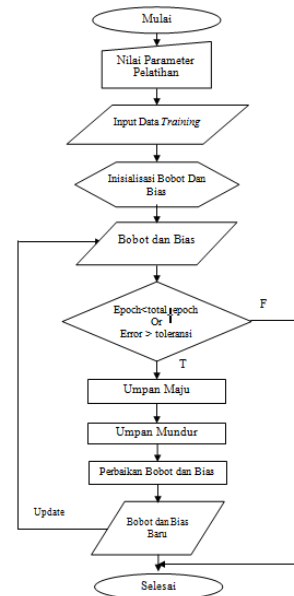
Dari data yang dihimpun maka didapatkan 9 fungsi keanggotaan himpunan *fuzzy* untuk parameter cuaca, 3 *fuzzy* fungsi keanggotaan *celcius*, 3 *fuzzy* fungsi keanggotaan kelembapan, dan 3 *fuzzy* fungsi keanggotaan titik embun. Dengan menggunakan teorema peluang yaitu kombinasi maka didapatkan 27 *rule* yang akan digunakan dalam *rule fuzzy* monitoring sensor DHT11. Setelah didapatkan 27 *rule fuzzy*, maka selanjutnya *rule fuzzy* tersebut akan diboboti yang menyebabkan masing-masing *rule* akan mempunyai pengaruh yang berbeda terhadap *output* monitoring cuaca nantinya. Agar *rule* tersebut bisa digunakan maka nilai bobot yang masih dalam bentuk nilai *crisp* harus difuzifikasi menjadi nilai probabilitas antara 0 sampai 1. Untuk probabilitas masing masing *rule* dapat dilihat pada tabel 2 berikut.

Tabel 2. *Rule* dan Derajat Keanggotaan

Rule	<i>Celcius</i>	Kelembapan	Titik embun	Nilai	Cuaca
R1	Rendah	Normal	Normal	0,23	Hujan lebat
R2	Normal	Normal	Tinggi	0,44	Gerimis
R3	Normal	Normal	Sangat Tinggi	0,55	Berawan
R4	rendah	Tinggi	Normal	0,44	Gerimis
R5	Normal	Tinggi	Tinggi	0,55	Berawan
R6	Normal	Tinggi	Sangat	0,66	Berawan

			Tinggi		
R7	Normal	Sangat Tinggi	Normal	0,55	Berawan
R8	Normal	Sangat Tinggi	Tinggi	0,66	Berawan
R9	Normal	Sangat Tinggi	Sangat Tinggi	0,77	Panas
R10	Tinggi	Normal	Normal	0,44	Gerimis
R11	Tinggi	Normal	Tinggi	0,55	Berawan
R12	Tinggi	Normal	Sangat Tinggi	0,66	Berawan
R13	Tinggi	Tinggi	Normal	0,55	Berawan
R14	Tinggi	Tinggi	Tinggi	0,66	Berawan
R15	Tinggi	Tinggi	Sangat Tinggi	0,77	Panas
R16	Tinggi	Tinggi	Normal	0,66	Berawan
R17	Tinggi	Sangat Tinggi	Tinggi	0,77	Panas
R18	Tinggi	Sangat Tinggi	Sangat Tinggi	0,88	Sangat Panas
R19	Sangat tinggi	Normal	Normal	0,55	Berawan
R20	Sangat tinggi	Normal	Tinggi	0,66	Berawan
R21	Sangat tinggi	Normal	Sangat Tinggi	0,77	Panas
R22	Sangat tinggi	Tinggi	Normal	0,66	Berawan
R23	Sangat tinggi	Tinggi	Tinggi	0,77	Panas
R24	Sangat tinggi	Tinggi	Sangat Tinggi	0,88	Sangat Panas
R25	Sangat tinggi	Sangat Tinggi	Normal	0,77	Panas
R26	Sangat tinggi	Sangat Tinggi	Tinggi	0,88	Sangat Panas
R27	Sangat tinggi	Sangat Tinggi	Sangat Tinggi	1,00	Sangat Panas

sambungan, baik bobot sambungan antar *input layer* dan *hidden layer* maupun antara *hidden layer* dan *output layer*. Bila terdapat lebih dari satu *hidden layer* maka juga terdapat pembobot antar *hidden layer* itu sendiri. Pelatihan akan terus dilakukan sampai *output* dan *target* sudah sesuai. Sedangkan tahap testing akan dilakukan jika proses pelatihan tersebut telah selesai. Flowchart proses pelatihan pada *Backpropagation* dapat dilihat pada Gambar 3.



Gambar 3. Flowchart Training JST

Pada Gambar 3. Proses pelatihan diawali dengan pengambilan data BMKG sebagai data sampel sebanyak 61 data yang terukur dari tanggal 1 Maret 2015 sampai dengan 30 April 2015. Data ini dimasukkan kedalam Matlab sebagai input dan target. Jika nilai *output* belum memenuhi target, maka dilakukan pelatihan beberapa *epoch* sampai hasilnya mendekati batas toleransi. Jika hal ini telah selesai dilakukan, maka nilai bobot dari *hidden dan output layer* diambil sebagai faktor penimbang untuk proses *testing*. Berikut ini sebagian sampel data training yang dapat dilihat pada Tabel 3.

Tabel 3. Sampel Data Training

Sampel Data	Input		Target		
	X1 (Suhu)	X2 (RH)	Y1	Y2	Y3
Cerah	27,825	80	1	1	0
Berawan	27,2	83	1	0	1
Hujan Ringan	26,625	85,25	1	0	0
Hujan Sedang	27	86,25	0	1	1
Hujan Lebat	26,57	88,75	0	0	1
Hujan Sangat Lebat	26,55	93	0	1	0

2. Perancangan Metode JST

Metode JST *Backpropagation* dalam sistem monitoring ini digunakan untuk menghitung output (keluaran). Proses JST meliputi 2 tahap, yaitu pelatihan (*training*) dan *testing*.

Tahap pelatihan menjelaskan langkah bagaimana suatu jaringan syaraf itu berlatih, yaitu dengan cara melakukan perubahan bobot

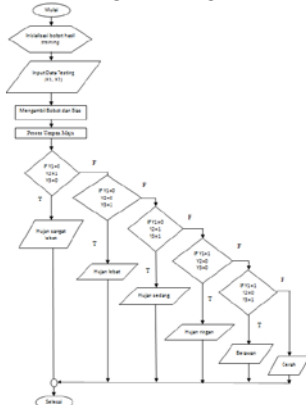
Bobot dan bias yang telah sesuai dari hasil proses *training* dapat dilihat pada Tabel 4.

Tabel 4 Bobot dan Bias Hasil *Training*

Dari Input											
Ke-Hidden Layer	X1	X2	Bias								
Z1	-0.21807	5.2606	-4.7261								
Z2	-2.5897	-4.2708	3.3582								
Z3	-4.997	-3.1212	3.2356								
Z4	7.6496	-5.862	-2.915								
Z5	4.7831	5.3205	-1.6948								
Z6	1.6854	3.7658	0.98502								
Z7	2.9947	4.3166	1.4071								
Z8	5.7148	-6.024	1.1199								
Z9	4.8387	1.9275	4.061								
Z10	5.244	-3.2248	4.6276								
Dari Hidden Layer											
Ke-Output	Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8	Z9	Z10	Bias
Y1	-1.417	-0.2888	-0.36614	0.4741	-0.14151	-0.057987	-0.40547	-0.34217	2.0303	-1.6605	-0.50087
Y2	2.1341	2.0538	-0.87843	-0.69973	0.72793	-1.9158	1.5116	0.14045	-0.8828	1.9673	-0.62248
Y3	-0.94407	-2.2302	2.2608	0.76518	0.21622	2.1363	-1.4757	0.41763	-2.155	1.3369	0.14143

Pada proses pengujian (*testing*), algoritma *Backpropagation* yang dijalankan hanyalah pada bagian proses *feedforward* (umpan maju) saja, yaitu dengan memasukkan bobot-bobot yang sesuai dari hasil proses *training* yang telah disimpan pada mikrokontroler hingga menghasilkan keluaran yang mendekati target. Data hasil pengukuran dari sensor DHT11 akan menjadi data input (X1 dan X2) pada mikrokontroler. Data input tersebut sudah dalam bentuk data digital sehingga dapat langsung diproses oleh Arduino dengan menggunakan bobot *Training* JST. Selanjutnya informasi yang dihasilkan akan ditampilkan di *handphone* berbasis Android. Diagram alir untuk proses *testing* dapat dilihat pada gambar 4 berikut.

Berdasarkan *flowchart* pada Gambar 8, variabel Y1, Y2, dan Y3 menunjukkan nilai *output* yang diinginkan jika diberi input X1 dan X2. Sistem secara keseluruhan menggunakan 2 *node* input layer, 10 *node* hidden layer dan 3 *node* output layer dengan menentukan target keluaran masing-masing.



Gambar 4. Flowchart *Testing* JST

HASIL DAN PEMBAHASAN

Setelah perancangan alat dan perancangan *software*, dihasilkan sebuah Sistem Monitoring Cuaca berbasis *handphone* Android dan Web menggunakan sensor DHT11 dan Mikrokontroler Arduino. Adapun tampilan sistem tersebut dapat dilihat pada Gambar 5.



Gambar 5. Rangkaian *Hardware* Sistem

Sebelum melakukan pengujian, alat harus diaktifkan terlebih dahulu dengan cara menghidupkan sistem melalui *powerbank*, kemudian meletakkan alat ini di ruangan terbuka. Selanjutnya menghubungkan *Bluetooth* *handphone* Android dengan *Bluetooth* HC-05. Setelah sistem diaktifkan dan *handphone* Android berhasil terhubung, pengguna diminta untuk menjalankan aplikasi monitoring cuaca pada *handphone* Android. Pada saat aplikasi berjalan, maka pengguna memilih menu “mulai” dan hasilnya akan ditampilkan di layar *handphone*. Sistem yang terhubung dengan *Smartphone* Android terlihat pada gambar 6 berikut.



Gambar 6. Sistem Terhubung dengan Android

1. Hasil Pengujian Sensor DHT11

Pengujian ini bertujuan untuk mengetahui tingkat keakuratan pembacaan suhu dan

kelembaban oleh sensor DHT11 yang terpasang di beberapa pin mikrokontroler. Pengujian dilakukan dengan cara membandingkan pembacaan sensor DHT11 dengan pembacaan suhu dan kelembaban sebenarnya yang diukur menggunakan *Vococal Digital Thermo Hygrometer*. Hasil perbandingan pengukuran dapat dilihat pada Tabel 5.

Tabel 5. Hasil Pengujian Sensor DHT11

Percobaan	Suhu dan Kelembaban (Sensor DHT11)		Suhu dan Kelembaban Sebenarnya (<i>Vococal Digital Thermo Hygrometer</i>)	
	Suhu (°C)	Kelembaban (%)	Suhu (°C)	Kelembaban (%)
1	36	37	36,2	37
2	30	44	30,2	44
3	30	43	30,4	43
4	31	43	31,1	43
5	36	37	36,8	38
6	27	56	27,0	56
7	30	47	30,2	47
8	41	36	36,8	37
9	34	45	34,4	45
10	33	41	31,7	42
11	29	48	29,3	48
12	31	46	29,6	47
13	28	49	28,0	49
14	28	50	28,0	50
15	30	44	30,2	44
16	33	46	33,4	46
17	35	43	35,6	44
18	25	50	24,9	50
19	31	56	31,0	56
20	30	57	30,0	59
21	28	48	28,1	48
22	29	44	29,3	44
23	33	46	33,4	46
24	29	44	29,0	44
25	31	43	31,4	43
26	35	38	33,9	38
27	30	44	29,0	48
28	28	48	27,8	48
29	31	43	31,2	43
30	30	44	30,1	44

Berdasarkan Tabel 5 dapat dilihat bahwa hasil pengukuran yang diperoleh sensor DHT11 hampir menyamai dengan hasil pengukuran yang sebenarnya yang dilakukan dengan menggunakan *Vococal Digital Thermo Hygrometer*. Terdapat sedikit kesalahan sensor dalam mengukur suhu dan kelembaban karena perbedaan nilai berkoma yang cukup jauh. Berdasarkan percobaan yang telah dilakukan, tingkat keakuratan sensor DHT11 pada 30 data uji sebesar :

$$\text{Akurasi} = \frac{22}{30} \times 100\% = 73,33\%$$

2. Tampilan Sistem Monitoring Web

Pada menu utama yang terlihat di web *xively*, terdapat 4 pilihan menu yang dapat dilihat oleh *user* yaitu tampilan “suhu (*celcius*)”, tampilan “kelembaban”, terakhir tampilan “titik embun”. Pada tampilan ini terdapat *graph* perubahan yang terjadi sesuai dengan hasil pembacaan sensor DHT11. *Data stream* memulai *upload* data ke *xively*, dengan syarat terpenuhi *Ethernet Client* dan *IP address* yang sudah cocok. Kemudian data sensor yang terbaca terlihat di *port com8* akan sesuai dengan data yang diterima *xively*, dengan *temperatur celcius* terbaca 31°C *humidity* atau kelembaban terbaca 63%, dan *dewpoint* (titik embun) sebesar 23,11 °C. Kemudian *xively client* melaporkan kembali ke *port com8* bahwa data sensor SUKSES terbaca di-*xively*.

Data log akan tersimpan secara otomatis pada database yang ada di akun web *xively*. Sehingga data yang telah dibaca oleh sensor pada setiap harinya dengan mudah ditelusuri via website tersebut. Tampilan *xively* dapat dilihat pada gambar 7 berikut.

Gambar 7. Tampilan *Realtime* Suhu

3. Hasil Pengujian Sistem dengan *Fuzzy*

Secara umum hasil perhitungan dengan menggunakan *toolbox fuzzy* pada matlab dan data yang terbaca sensor, memiliki selisih keakuratan yang relatif kecil jika dibandingkan dengan hasil prakiraan, ini terbukti dari hasil pengujian 10 data yang diambil tiap jam, seperti terlihat pada tabel 6 dibawah ini :

Tabel 6. Hasil Pengujian *Fuzzy*

Pengujian	Suhu (°C)	Kelembaban (%)	Titik embun (°C)	Akurasi data (%)	Toolbox fuzzy	Prakiraan BMKG			Keterangan
						(°C)	(%)	Ket	
1	29	60	20,43	83,3	0,47/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG
2	29	52	18,13	66,7	0,452/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG
3	35	50	22,08	75,0	0,5/ Gerimis	23-32	60-94	Hujan ringan	Tidak COCOK dengan prakiraan BMKG (35 °C)
4	32	51	20,58	75,0	0,474/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG
5	31	54	19,66	75,0	0,46/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG
6	30	57	20,53	83,3	0,472/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG
7	29	58	19,88	83,3	0,46/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG
8	28	58	19,88	83,3	0,46/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG
9	30	60	21,37	83,3	0,487/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG
10	31	64	23,37	83,3	0,5/ Gerimis	23-32	60-94	Hujan ringan	COCOK dengan prakiraan BMKG



Dari tabel 6 tersebut secara garis besar dapat dijelaskan bahwa terdapat pengujian yang tidak sesuai dengan data prakiraan BMKG, data yang tidak sesuai tersebut adalah pengujian data input yang ke 3, data tersebut tidak sesuai dengan data prakiraan BMKG karena celcius

$$\text{Persentase akurasi data} = \frac{\text{Jumlah Persentasi Tiap Pengujian}}{\text{Banyaknya Pengujian}}$$

yang didapat lebih dari 32°C. Namun pada perhitungan *fuzzy* sudah memenuhi bobot bahwa pada pengujian data input sistem monitoring tersebut dikondisikan “gerimis”

$$\frac{\text{Akurasi Pengujian 1} + \text{Akurasi Pengujian 2} + \text{Akurasi Pengujian 3} + \text{Akurasi Pengujian 4} + \text{Akurasi Pengujian 5} + \dots + \text{Akurasi Pengujian 10}}{10}$$

$$= \frac{83,3\% + 66,7\% + 75,0\% + 75,0\% + 75,0\% + 83,3\% + 83,3\% + 83,3\% + 83,3\% + 83,3\%}{10}$$

atau “hujan ringan”. Hal ini sebetulnya disebabkan karena letak sensor yang dipindahkan ke seng, dengan keadaan yang panas, sehingga sensor menangkap panas dari seng tersebut.

Untuk menggambarkan keberhasilan sistem dengan menggunakan logika *fuzzy*, dihitung dengan perhitungan kesalahan absolut relatif fraksional sebagai berikut:

Dengan demikian berdasarkan tingkat keberhasilan pengujian data 1 sampai 10, maka rata rata keberhasilan adalah 80,15 % seperti terlihat pada perhitungan dibawah ini.

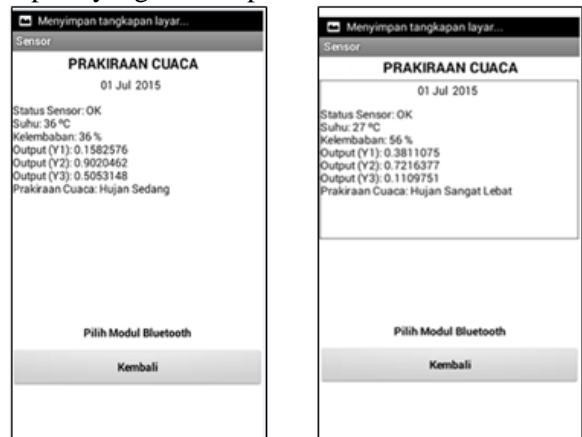
$$= 80,15\%$$

4. Hasil Pengujian Sistem dengan JST

Pengujian dilakukan dengan cara mengambil 1 data dalam satu hari dengan total data sebanyak 14 data yang diambil mulai tanggal 22 Mei 2015 – 07 Juni 2015. Hal ini dilakukan untuk melihat prakiraan cuaca harian yang dihasilkan oleh sistem berdasarkan

pengolahan dengan bobot JST BMKG yang sudah disimpan dalam mikrokontroler.

Proses pengujian (*testing*) dilaksanakan pada bagian *feedforward* (umpan maju) saja, yaitu dengan memasukkan bobot-bobot JST yang sesuai dari hasil proses *training* hingga menghasilkan nilai *output* (Y1, Y2, dan Y3). Dengan adanya nilai *output* tersebut, maka sistem dapat menentukan prakiraan cuaca yang akan muncul berdasarkan pengkondisian nilai *output* dengan fungsi aktivasi algoritma *backpropagation*. Sedangkan nilai input (X1 dan X2) yang masuk ke mikrokontroler berasal dari data hasil pengukuran dengan sensor DHT11 dalam bentuk data suhu dan kelembapan. Hasil *testing* akan langsung ditampilkan di aplikasi sistem monitoring cuaca setelah tahap umpan maju selesai dilaksanakan seperti yang terlihat pada Gambar 8



Gambar 8. Tampilan Hasil Prakiraan

Dari Gambar 8 dapat dilihat hasil monitoring yang tampil berupa status sensor, suhu, kelembapan, dan prakiraan cuaca yang didapatkan dari *output* JST. Beberapa langkah dan persamaan *feedforward* yang digunakan pada *testing* JST untuk mendapatkan *output* yaitu :

- a. Langkah 1: Masing-masing unit input (Xi, i =1,...n) menerima sinyal input Xi dan sinyal tersebut disebarkan ke unit bagian atas lapisan tersembunyi (*hidden units*)
- b. Langkah 2: Masing-masing *hidden* menjumlahkan faktor penimbang melalui persamaan berikut :

$$Z_{net_j} = V_{jo} + \sum_{i=1}^n X_i V_{ji} \dots\dots\dots(1)$$

$$Z_j = f(Z_{net_j}) \dots\dots\dots(2)$$

$$Z_j = \frac{1}{1 + e^{-Z_{net_j}}} \dots \dots \dots (3)$$

Kemudian mengirim sinyal tersebut ke semua unit di atasnya (*output unit*).

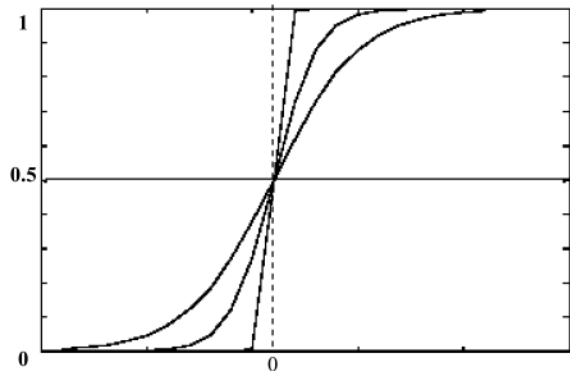
c. Langkah 3: Masing-masing unit output ($Y_k, k=1,2,3,\dots,m$) dijumlahkan faktor penimbang melalui persamaan berikut:

$$Y_{net_k} = W_{ko} + \sum_{j=1}^n Z_j W_{kj} \dots \dots \dots (4)$$

Menghitung sesuai dengan fungsi aktivasi sigmoid biner yaitu :

$$Y_k = f(Y_{net_k}) = \frac{1}{1 + e^{-Y_{net_k}}} \dots \dots \dots (5)$$

Nilai *output* yang didapatkan dari persamaan-persamaan diatas berkisar antara 0 hingga 1. Hal ini dikarenakan fungsi aktivasi sigmoid biner hanya memiliki nilai 0 dan 1. Oleh karena itu, berdasarkan target yang sudah dibuat sebelumnya pada *Neural Network Matlab*, maka nilai *output* (Y_k) bernilai 1 apabila Y_k atau $F(Y_{netk}) \geq 0,5$ dan bernilai 0 apabila Y_k atau $F(Y_{netk}) < 0,5$ seperti yang didasarkan pada Gambar 9



Gambar 9. Grafik Fungsi Aktivasi Sigmoid Biner

Berikut ini adalah 14 data dari hasil *testing* JST yang dapat dilihat pada Tabel 7 berikut.

Tabel 7. Output Testing JST

No.	Input	Nilai Pembacaan Sistem	Output Testing	Prakiraan Cuaca (JST Sistem)
1.	X1	29	Y1 = 0 Y2 = 1 Y3 = 0	Hujan Sangat Lebat
	X2	44		
2.	X1	30	Y1 = 0 Y2 = 1 Y3 = 0	Hujan Sangat Lebat
	X2	44		
3.	X1	0	Y1 = 0 Y2 = 0 Y3 = 0	Tidak Terdefinisi
	X2	0		
4.	X1	27	Y1 = 0 Y2 = 1 Y3 = 0	Hujan Sangat Lebat
	X2	50		

5.	X1	34	Y1 = 0 Y2 = 1 Y3 = 1	Hujan Sedang
	X2	40		
6.	X1	35	Y1 = 0 Y2 = 1 Y3 = 1	Hujan Sedang
	X2	37		
7.	X1	24	Y1 = 0 Y2 = 1 Y3 = 0	Hujan Sangat Lebat
	X2	67		
8.	X1	36	Y1 = 0 Y2 = 1 Y3 = 1	Hujan Sedang
	X2	36		
9.	X1	34	Y1 = 0 Y2 = 1 Y3 = 1	Hujan Sedang
	X2	40		
10.	X1	0.00	Y1 = 0 Y2 = 0 Y3 = 0	Tidak Terdefinisi
	X2	0.00		
11.	X1	33	Y1 = 0 Y2 = 1 Y3 = 1	Hujan Sedang
	X2	40		
12.	X1	31	Y1 = 0 Y2 = 1 Y3 = 0	Hujan Sangat Lebat
	X2	43		
13.	X1	44	Y1 = 0 Y2 = 0 Y3 = 0	Tidak Terdefinisi
	X2	30		
14.	X1	28	Y1 = 0 Y2 = 1 Y3 = 0	Hujan Sangat Lebat
	X2	48		

Berdasarkan Tabel 7. dapat dilihat bahwa terdapat 3 data pengujian yang tidak terdefinisi dan 11 data pengujian yang berhasil membaca prakiraan cuaca. Oleh karena itu dapat dihitung tingkat keberhasilan prakiraan cuaca melalui metode jaringan syaraf tiruan dengan algoritma *backpropagation* pada 14 data uji sebesar :

$$\frac{11}{14} \times 100\% = 78,6\%$$

Tingkat keberhasilan menunjukkan bahwa hampir semua data yang diambil dapat didefinisikan oleh sistem melalui persamaan *feedforward testing* dengan bobot JST. Hasil prakiraan yang tidak terbaca dapat disebabkan oleh berbagai faktor diantaranya:

- Pengukuran yang terlalu lama pada ruang terbuka
- Peletakan sistem pada tempat yang salah
- Kepala sensor yang masih terlalu panas ataupun terlalu lembab
- Perhitungan *output* antara bobot JST dengan input sensor yang tidak sesuai sehingga sistem tidak dapat menentukan prakiraan cuaca.

SIMPULAN DAN SARAN

1. Kesimpulan

Berdasarkan penelitian yang telah dilakukan maka dapat diambil beberapa kesimpulan sebagai berikut :

- a. Sistem monitoring cuaca yang dirancang menggunakan sensor *DHT11*, *arduino uno*, dan *Ethernet shield* mampu menampilkan data monitoring dengan *output realtime web* pada pengujian tiap 10 jam.
- b. Penghitungan yang dihasilkan dengan metode *fuzzy* tidak jauh berbeda dengan prakiraan yang informasikan BMKG dengan persentasi keberhasilan pengujian data input sistem monitoring adalah 80,15 %.
- c. Tingkat keberhasilan sistem monitoring cuaca berbasis android dengan metode jaringan syaraf tiruan *backpropagation* adalah sebesar 78,6 % dari 14 data uji.

2. Saran

Untuk pengembangan sistem ini dapat disarankan beberapa hal sebagai berikut :

- a. Pengembangan dan implementasi sistem monitoring cuaca, tidak hanya tergantung pada suhu celsius, kelembapan dan titik embun saja, tetapi juga ada faktor-faktor lainnya seperti, kecepatan angin.
- b. Program monitoring secara realtime web dapat dikembangkan untuk aplikasi berbasis mobile sehingga bisa diakses melalui smartphone. Kecepatan akses data juga mempengaruhi kualitas pengiriman data realtime ketika data upload.

DAFTAR PUSTAKA

- Budi. 2014. Data Suhu dan Kelembapan. Stasiun Meteorologi Klimatologi dan Geofisika II. Padang
- Budiharto, Widodo. 2006. Panduan Belajar Mikrokontroler. Bandung: Elex Media Komputindo.
- Hartono. 2007. *Jelajah Bumi dan Alam Semesta*. Jakarta: Grafindo.
- Kusumadewi, Sri dan Hari, Purnomo. 2010. *Aplikasi Logika Fuzzy Untuk pendukung keputusan*. Yogyakarta: Graha Ilmu.
- Naba, Agus.2009. *Belajar Cepat Fuzzy Logic Menggunakan Matlab,CV* Andi Offset,Yogyakarta.
- Nalwan, Andi. "Teknik rancang bangun robot, tingkat dasar",. Yogyakarta : CV. Andi Offset, 2012. p. 34,38,39,45,59.
- Sudiharto, Agus."Penerapan Dasar Transducer Dan Sensor". Yogyakarta : Kanisius, 2002. p. 2. ISBN.
- ZuhalfProf, and Zhanggischan. "Prinsip Dasar Elektro Teknik". Jakarta : PT. Gramedia Pustaka Utama, 2004. pp. 371-372.