

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

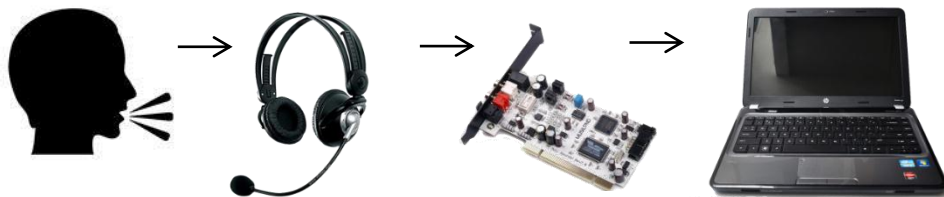
Pada sub bab ini yaitu pembahasan langkah awal sebelum membuat sistem dan hasil proses sistem serta semua elemen yang terkait. Dari mulai pencuplikan suara, metode yang digunakan untuk mengubah sinyal analog menjadi spektrum frekuensi dan juga semua kebutuhan untuk menjadikannya sebagai sistem guna mengidentifikasi suara berdasarkan intonasi dan gaya bahasa yang berbeda.

3.2 Hasil Analisis Sistem

Adapun hasil analisa pada sistem identifikasi suara dapat dibagi menjadi beberapa tahap, sebagai berikut

3.2.1 Cara pencuplikan suara

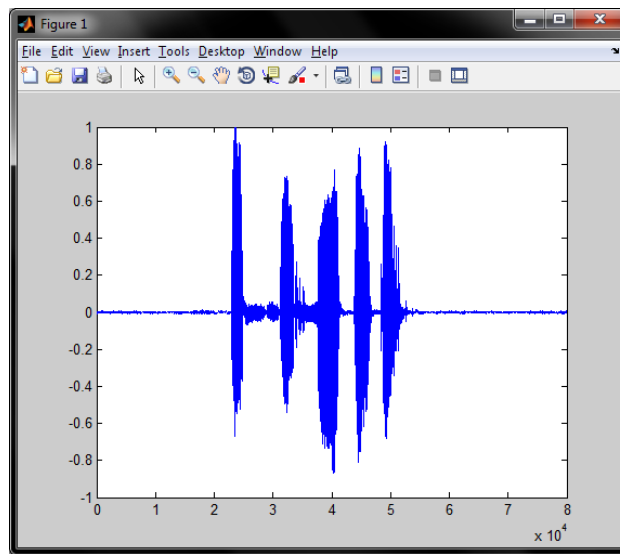
Pencuplikan suara menggunakan microphone headset , kemudian tersambung pada soundcard dan proses penyimpanan data menggunakan Matlab R2008b. proses perekaman dan penyimpanan data diatas adalah sebagai berikut:



Gambar 3.1 Diagram Alir Sistem

Gambar 3.1 menunjukkan diagram alir sistem yang akan dibuat.

Dari proses tersebut akan di dapat sample suara dengan ekstensi file *.WAV dan masih bersifat sinyal analog seperti pada gambar 3.2. Dalam proses perekaman harus dalam keadaan sepi dengan asumsi memperkecil adanya noise pada sample suara tersebut. Jarak antara microphone dengan mulut sekitar ± 5 cm.



Gambar 3.2 hasil sinyal analog ucapan ‘informatika’ oleh suara laki - laki.

Proses pencuplikan suara dilakukan oleh satu orang laki – laki dan satu orang perempuan sebanyak masing – masing 90 data dengan 30 Data yang benar, 30 Data yang salah dan untuk pelatihan 30 data uji dengan intonasi dan gaya bahasa yang berbeda. Dengan parameter target yaitu 1 adalah benar dan 0 adalah salah. Penamaan pada data dapat dilihat dalam tabel 3.1 dan selengkapnya disajikan pada lampiran.

Tabel 3.1 Penamaan data

No	Data Benar dengan target (1)		
1	Akuntansi	16	Aakuntannnsiii
2	Aakuntansi	17	AaaAkuntansi
3	Aakuuntansi	18	AaaAkuUntansi
4	Aakuunntansi	19	AaAkuUnnNtansi
5	Aakuunntaansi	20	AaAkuUnnNtaAnsi
6	Aakuunntaanssi	21	AaAkuUnnNtaAnnNsi
7	Aakuunntaanssii	22	AaAkuUnnNtaAnnNsiii
8	Akuuntansi	23	Akuntansi ?
9	Akunntansi	24	Akuntansi !
10	Akuntaansi	25	Akuntansi.....
11	Akuntannnsi	26	Akun?tansi
12	Akuntansiii	27	Akuntan?si
13	Aakunntansi	28	A?kuntansi
14	Aakuntaansi	29	A?kuntan?si
15	Aakuntannnsi	30	aaaA?kuunn?tan?si?

No	Data Benar dengan target (1)		
1	Informatika	16	IinnfoorrRmaatika
2	Iinformatika	17	IinnNfoorrRmaaAtika
3	Iinnformatika	18	InformAaatikaaA
4	Iinnfoormatika	19	iiIinformatika
5	Iinnfoorrmaatika	20	iiIinnNformatika
6	Iinnfoorrmaatiika	21	iiIinformatikaaAa
7	Iinnfoorrmaatiikaa	22	iiInnNformatiilika
8	Infoormatika	23	informatika ?
9	Infoorrmaatika	24	Informatika !
10	Infoorrmaatiika	25	Informatika.....
11	Infoorrmaatiikaa	26	In?formatika
12	informaaAtika	27	Informa?tika
13	innNformatiilika	28	infor?matika
14	IinnNformaaAtikaa	29	in?forma?tika
15	InforrRmatika	30	iiInnN?formaaA?ti?ka?

No	Data Salah dengan target (0)		
1	a	16	akunNtaaN
2	kun	17	aaAkuntaaN
3	tan	18	aaAkuntaaNs
4	si	19	kuuUnntannNsi
5	akun	20	kuUnntansiii
6	akunta	21	aaAkuntaaA
7	akuntan	22	aaAtansi
8	akuntans	23	Aatannnsiii
9	kuntan	24	aakunnsiii
10	kuntans	25	kuunnsii
11	kuntansi	26	taaannnN
12	tans	27	aaAk
13	tansi	28	aanNsi
14	akuuU	29	aakuunntaassii
15	akuuUn	30	aakuutaansi

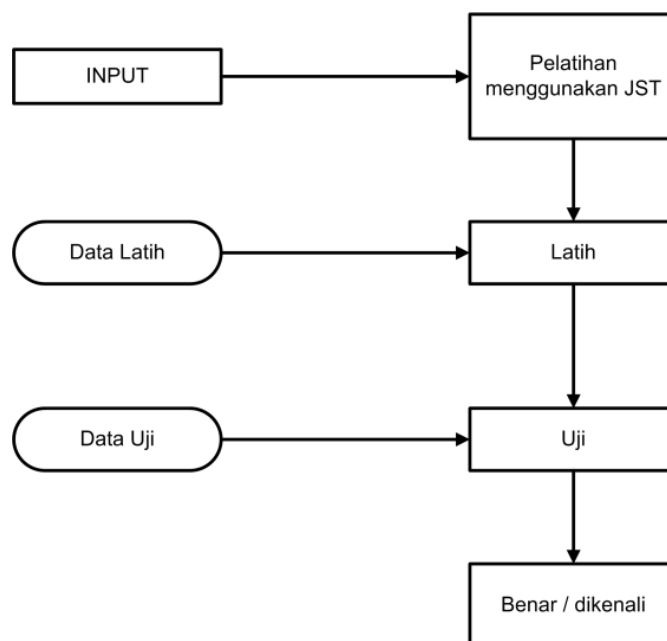
No	Data Salah dengan target (0)		
1	In	16	matika
2	For	17	tika
3	Ma	18	iinnfoorm
4	Ti	19	iinnforrmaasi
5	Ka	20	aatiikaa
6	Infor	21	innfomatika
7	Informa	22	iinnmaatiikaa
8	Informat	23	iinntiikaa
9	Informatik	24	iikaa
10	Informatiks	25	innfootiikaa
11	Format	26	innfoorrkaa
12	Formati	27	maaat
13	Formatika	28	innfo
14	Mati	29	innfoorrmaakaa
15	Matik	30	foorrtiikaa

No	Data Uji		
1	a-kuntansi	16	akutansi
2	a-kun-tansi	17	angkutansi
3	a-kun-tan-si	18	akuntansi
4	aaA-kuntansi	19	akuntansi
5	aaA-kuuN-tansi	20	autansi
6	aaA-kuuN-tanN-sii	21	akuntanci
7	aaAkuuN-tansi	22	aaAtasi
8	aaAkuuN-tanN-si	23	kuuUtaanNsi
9	aaAkuuNtanN-si	24	akuuUtannNsi
10	aaA-kuuNtanN-si	25	aaaAkutannN
11	aaA-kunnNtanNsii	26	Akuisisi
12	aAakuUun-tansii?	27	kuuiitannsi
13	aAakunntanN?-si?	28	Akreditasi
14	aaA?-kunNnN-tansi!	29	uunntaaaaa
15	akun-taaAaAnsi	30	angkutan

No	Data Uji		
1	in-formatika	16	inforR-maaA-tiiks
2	infor-matika	17	inn-fomaaA-tika
3	informa-tika	18	i-formatii-ka
4	informati-ka	19	iinN-formasikaa
5	in-for-matika	20	infor-meetikaa
6	in-for-ma-tika	21	forr-maaA
7	infor-ma-ti-ka	22	informatika-ka
8	in-forma-tika	23	in-for-ma-matika
9	in-formati-ka	24	in-in-formatika
10	iinn-formatika	25	foorrrmm
11	inforR-matika	26	formasi
12	informaaA-tika	27	Informasi
13	informatiiI-ka	28	telematika
14	innN-forr-matika	29	sistematika
15	inn-forrR-maa-tika	30	Matematika

3.2.2 Perancangan Sistem

Berikut adalah penjelasan tentang perancangan sistem untuk mengidentifikasi suara dengan FFT dan backpropagation. Adapun tahapan – tahapan tentang proses identifikasi suara tersebut di tunjukkan pada gambar 3.3



Gambar 3.3 Blok Diagram Sistem Identifikasi Suara.

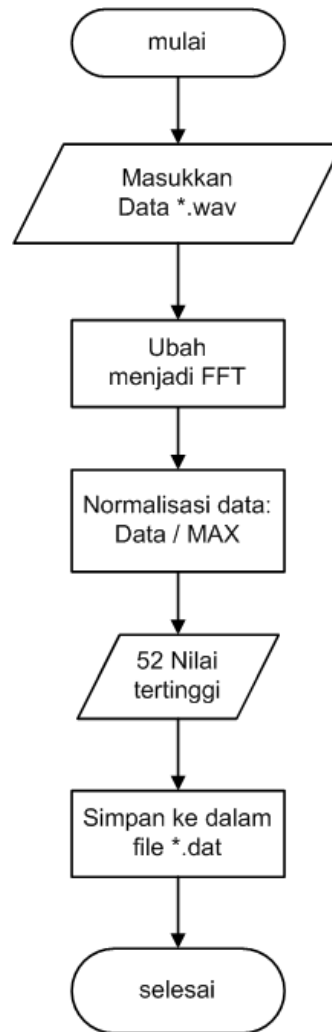
Penjelasan untuk blok diagram tersebut adalah sebagai berikut :

- a. Input = masukan yang diperlukan adalah pola suara yang telah dijadikan matrik untuk dilatih pola suara yang benar dan tidak benar.
- b. Pelatihan data = penentuan nilai normalisasi, jumlah jaringan neuron, fungsi aktifasi dan fungsi identitas untuk keluarannya.
- c. Pengujian data = proses pelatihan masing - masing tiap pola data yang dilakukan tanpa dilatih terlebih dahulu sehingga sistem akan mengetahui apakah hasilnya seperti data yang sudah dilatih dan pola data bisa diidentifikasi dapat dikenali atau tidak.
- d. Benar / dikenali = proses menentukan hasil keputusan apakah data dikenali atau tidak.

Setelah file *.WAV diambil dari proses perekaman suara dengan Matlab, kemudian dilakukan tahap pengkonversian dari sinyal analog menjadi spektrum frekuensi dengan menggunakan FFT. Setelah itu akan ada proses normalisasi data dan menyimpannya dalam bentuk *.DAT, kemudian akan dilakukan tahap pelatihan data menggunakan backpropagation. selanjutnya akan didapat nilai bobot yang terbaik untuk digunakan pada data uji.

3.2.3 Pengolahan Data Awal

Adapun tahap pengolahan data awal ini meliputi beberapa proses, seperti pada gambar 3.4 yaitu:



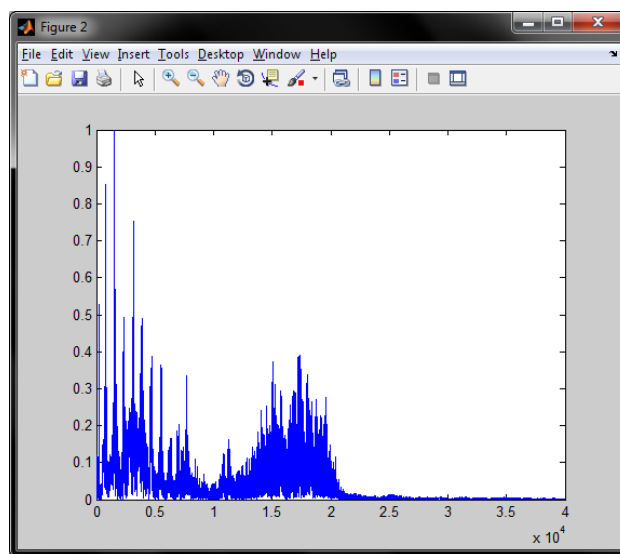
Gambar 3.4 Blok diagram proses data awal

Langkah – langkah proses awal :

1. Mulai
2. Masukkan data *.wav yang telah di rekam sebagai pola acuan yang dianggap benar.
3. Ubah file *.wav menjadi nilai frekuensi dengan menggunakan FFT.
4. Lakukan normalisasi data dengan mencari nilai tertinggi sebagai pembagi seluruh data.
5. Menentukan 52 titik nilai yang tertinggi sebagai nilai acuan terhadap semua pola yang akan dijadi kann pelatihan.
6. Simpan kedalam file *.dat
7. selesai

3.2.4 Perubahan sinyal analog ke spektrum frekuensi

Setelah mendapatkan sampel suara berekstensi file *.WAV, proses selanjutnya adalah mengubah dari sinyal analog (*.WAV) menjadikan ke spektrum frekuensi dimana berguna untuk mendapatkan nilai tiap gelombang untuk pola suara yang akan dijadikan pelatihan seperti pada hasil di gambar 3.5. Hal yang pertama dilakukan adalah memanggil file menggunakan Matlab serta mengubahnya menjadi FFT.



Gambar 3.5 plot ucapan ‘informatika’ laki – laki setelah melalui proses FFT.

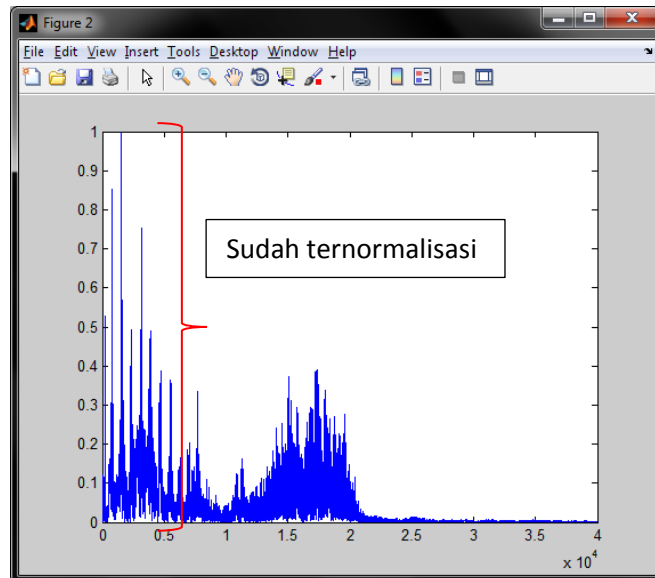
Setelah memperoleh nilai FFT pada setiap data, selanjutnya akan di konversi ke dalam bentuk file data (*.dat).

3.2.5 Normalisasi

Pada tahap ini adalah proses awal untuk mencari nilai normalisasi dari setiap data sebelum menentukan titik – titik tertinggi pada pola suara yang dianggap benar dengan mencari nilai tertinggi terlebih dahulu dan kemudian membaginya pada seluruh data

$$Normalisasi = \frac{Nilai\ maksimum}{Seluruh\ nilai\ setiap\ data} \dots\dots\dots(3.1)$$

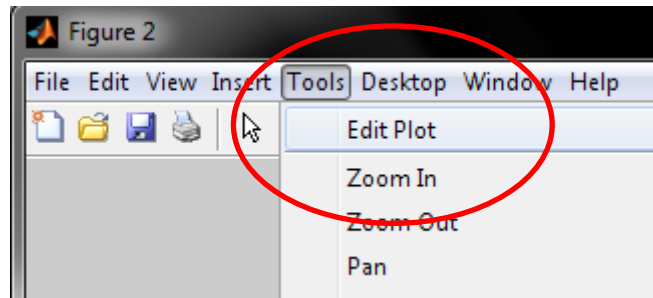
Sehingga seluruh data menunjukkan nilai antara 0 terendah dan 1 tertinggi seperti pada gambar 3.6



Gambar 3.6 Data yang sudah ternormalisasi .

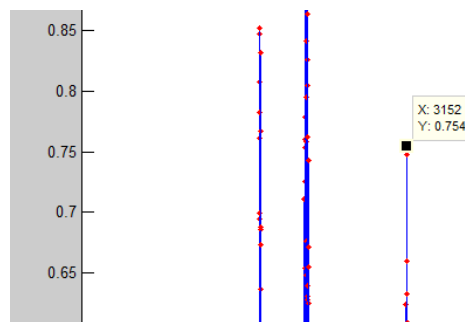
3.2.6 Penentuan titik pola acuan

Setelah mendapatkan nilai normalisasi , proses selanjutnya adalah penentuan titik- titik yang akan digunakan sebagai acuan pola suara yang dianggap benar. Hal ini dapat dilakukan dengan menggunakan tool dari Matlab itu sendiri yaitu dengan menggunakan edit plot yang sudah melalui proses konversi kedalam bentuk sinyal frekuensi atau FFT. Karena sampel data menggunakan frekuensi cuplikan 16000. Maka dalam waktu 5 detik saja akan mendapatkan 80000 nilai pada setiap data. Kelebihan dari FFT tersebut adalah bersifat pencerminan dengan kata lain setengah dari data merupakan pencerminan atau sama dengan setengah dari data tersebut. Jadi akan lebih cepat untuk langsung mengabaikan setengah dari setiap data yang sudah diolah menjadi FFT tersebut sehingga menjadi 40000 nilai dan diambil nilai yang sering muncul. Dengan menggunakan edit plot seperti pada gambar 3.7



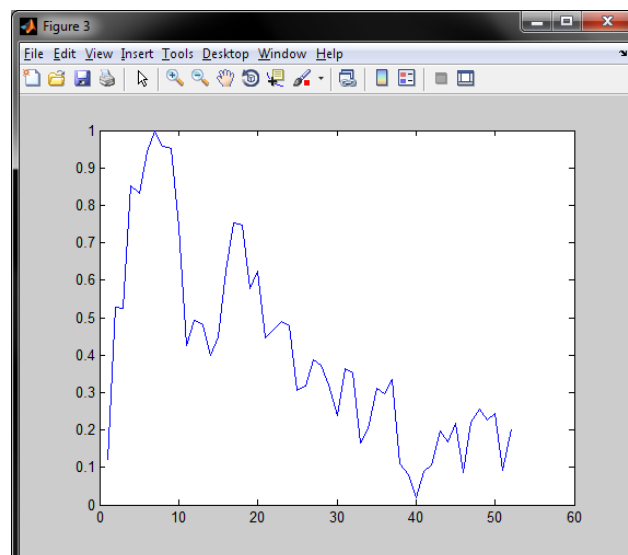
Gambar 3.7. Mencari nilai yang sering muncul menggunakan edit plot

Kemudian menampilkan nilai dari setiap titik yang dianggap nilai paling sering muncul seperti pada gambar 3. 8 Dengan menggunakan tool data cursor



Gambar 3.8 Menampilkan nilai data menggunakan tool data cursor.

Sehingga akan didapatkan pola yang telah memiliki 52 nilai dari titik – titik yang ditentukan seperti gambar 3.9.



Gambar 3.9 Pola 52 titik dari ucapan ‘informatika’ laki - laki.

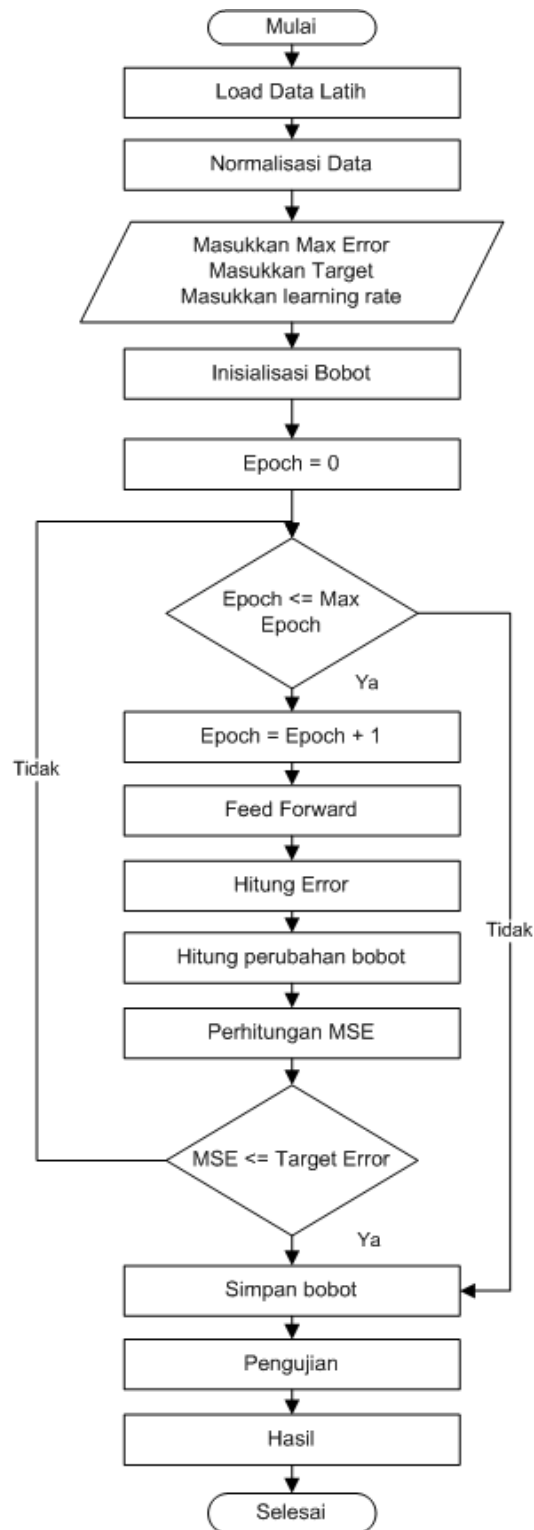
Setelah mendapatkan pola dari setiap data, maka bisa dilakukan proses pembelajaran atau pelatihan .

3.3 Backpropagation

Setelah mendapatkan semua pola dari data pelatihan, maka selanjutnya bisa di jadikan sebagai input data pelatihan backpropagation. dalam penelitian ini akan mencoba menggunakan beberapa Arsitektur pelatihan backpropagation yang terdiri dari

- 60 Neuron *input layer*
- 1 Neuron *Output Layer*
- 2 - 5 *Hidden layer* maksimal 250 Neuron dalam 1 hidden layer
- Fungsi aktivasi menggunakan fungsi sigmoig bipolar (logsig) dalam *hidden layer*
- Metode training menggunakan Backpropagation dengan Metode Penurunan Gradien serta variasi Learning rate dan momentum.
- Batas MSE yang digunakan 0,00001
- Besar learning rate antara 0,01 – 0,5 dan besar momentum antara 0,01 – 0,9
- Weight (bobot) diinisialisasikan secara acak antara -0,5 sampai 0,5

Tujuan variasi ini adalah menentukan waktu tercepat dan hasil terbaik dalam proses pelatihan. Penentuan jumlah neuron pada lapisan hidden dilakukan secara eksperimen karena belum ada metode yang baku dalam menentukan jumlah neuron pada lapisan hidden.



Gambar 3.10 Diagram Alir proses Backpropagation.

Langkah-langkah propagasi balik untuk membangun jaringan pada penelitian ini sebagai berikut:

1. Load Data

Proses pembagian data yang akan di gunakan pada penelitian ini 70 % data pelatihan dan 30 % sebagai data pengujian. Aspek pembagian data ditekankan agar mendapat data pelatihan yang cukup.

2. Normalisasi data

Sebelum pelatihan maka akan dilakukan normalisasi agar data input masuk dalam range preprocessing.

Matlab menyediakan fungsi `prestd` untuk melakukan normalisasi, dengan syntax : `[pn,meanp,stdp,tn,meant,stdt] = prestd (p,t)`

Dengan p adalah matriks input dan t adalah matriks target .

pn : matriks input yang ternormalisasi

tn : matriks target yang ternormalisasi

meanp : mean pada matriks input asli (p)

stdp : deviasi standar pada matriks input asli (p)

meant : mean pada matriks target asli (t)

stdt :deviasi standar pada matriks target asli (t)

3. Perancangan Struktur jaringan yang optimum

Penentuan jumlah lapisan masukan (input), lapisan tersembunyi dan jumlah lapisan keluaran yang akan digunakan dalam jaringan.

4. Pemilihan Koefisien pemahaman (learning rate) dan Momentum

Pemilihan learning rate dan momentum memiliki peranan penting. Hasil keputusan yang kurang memuaskan dapat diperbaiki dengan penggunaan koefisien pemahaman dan momentum secara trial and error untuk dapat nilai bobot yang paling optimum agar MSE jaringan dapat diperbaiki.

5. Postprocessing / Denormalisasi

Setelah proses pelatihan, output jaringan harus di kembalikan (denormalisasi) ke nilai asli pada range sebenarnya. Matlab menyediakan fungsi `poststd` untuk melakukan denormalisasi dengan syntax :

`[p,t] = poststd (pn,meanp,stdp,tn,meant,stdt)` dengan p dan t adalah matriks yang telah di denormalisasi.

Selanjutnya akan terjadi proses pengujian dari backpropagation sampai menemukan bobot yang bagus sehingga dapat digunakan untuk pengujian data uji.

3.4 Alat Ukur Evaluasi

Dalam penelitian ini menggunakan alat ukur Confusion Matrix dengan tujuan untuk mempermudah dalam menganalisis performa algoritma karena confusion matrix memberikan informasi dalam bentuk angka sehingga dapat dihitung rasio keberhasilan klasifikasi. Confusion matrix adalah salah satu alat ukur yang berbentuk matrik 2 x 2 yang digunakan untuk mendapatkan jumlah ketepatan klasifikasi. Dalam penelitian ini memiliki keluaran apabila mendekati 1 bernilai benar sedangkan salah apabila bernilai 0. Tiap kelas yang di prediksi memiliki empat kemungkinan keluaran yang berbeda, yaitu true positives (TP) dan true negatives (TN) menunjukkan ketepatan klasifikasi. Jika prediksi keluaran positif sedangkan nilai aslinya adalah negatif maka disebut false positive (FP) dan jika prediksi keluaran bernilai negatif sedangkan aslinya positif maka disebut dengan false negative (FN). Berikut ini tabel 3.2 disajikan bentuk confusion matrix

Tabel 3.2 Confusion Matrix

Kelas Sebenarnya	Kelas Prediksi	
	Benar	Salah
Benar	D	C
Salah	B	A

Rumus ini melakukan perhitungan dengan 4 keluaran, yaitu: recall, precision, accuracy dan error rate.

- Recall adalah proporsi kasus positif yang diidentifikasi dengan benar.

$$\text{Rumus dari Recall} = \frac{D}{(C+D)}$$

- Precision adalah proporsi kasus dengan hasil positif yang benar.

$$\text{Rumus dari Precision} = \frac{D}{(B+D)}$$

- Accuracy adalah perbandingan kasus yang diidentifikasi benar dengan jumlah semua kasus.

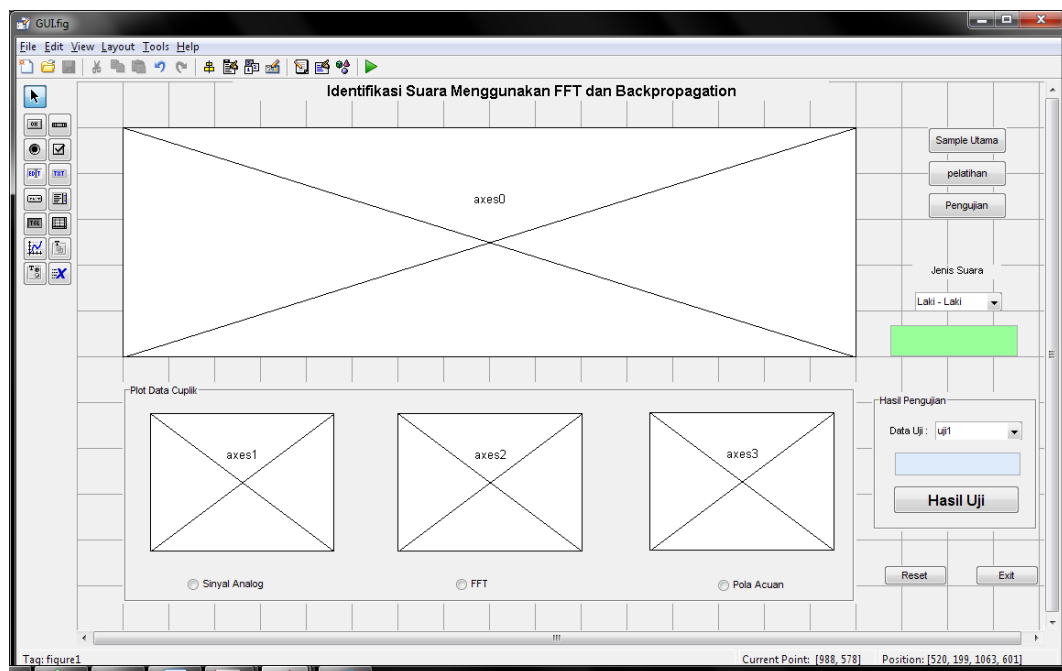
$$\text{Rumus dari Accuracy} = \frac{(A+D)}{(A+B+ C+D)}$$

- Error Rate adalah kasus yang diidentifikasi salah dengan sejumlah semua kasus

$$\text{Rumus dari Error Rate} = \frac{(B+C)}{(A+B+ C+D)}$$

3.5 Desain Antar muka

Gambaran untuk desain antar muka aplikasi Identifikasi Suara menggunakan FFT dan backpropagation dapat dilihat pada gambar 3.11.



Gambar 3.11 Desain Antar Muka