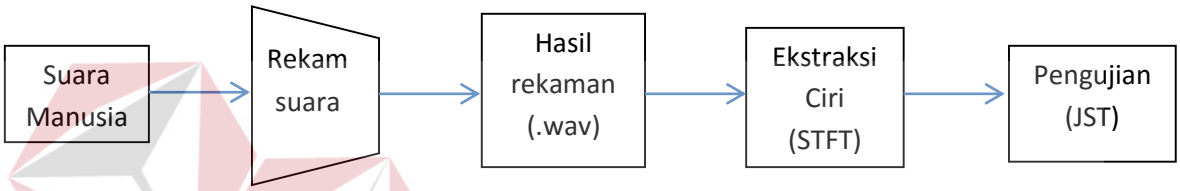
BAB III

METODE PENELITIAN

3.1. Model Penelitian

Penelitian yang dilakukan dapat dijelaskan dengan lebih baik melalui blok diagram seperti yang terliat pada Gambar 3.1.



Gambar 3.1. Blok Diagram Penelitian

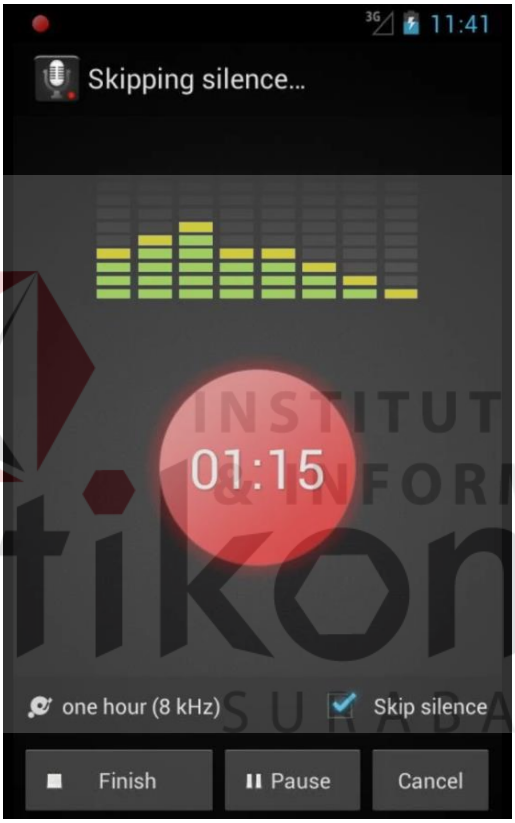
Berikut adalah keterangan setiap blok dari sistem blok diagram pada Gambar 3.1.

3.1.1 Suara manusia.

Pada blok ini, suara manusia yang direkam adalah saat pengucapan huruf vokal (a, e, i, o, u) dan individu tersebut berusia 17-30 tahun. Kondisi pita suara manusia harus dalam keadaan sehat dan tidak serak pada saat pengucapan sehingga suara dapat terdengar dengan jelas. Huruf vokal diucapkan dengan nada datar atau tanpa nada, yaitu nada normal pada saat manusia berbicara, ini untuk memudahkan proses analisa suara tersebut. Begitu juga intonasi dan volume pengucapan huruf harus seragam dari awal perekaman sampai akhir, atau tidak boleh berubah-ubah.

3.1.2 Perekaman Suara

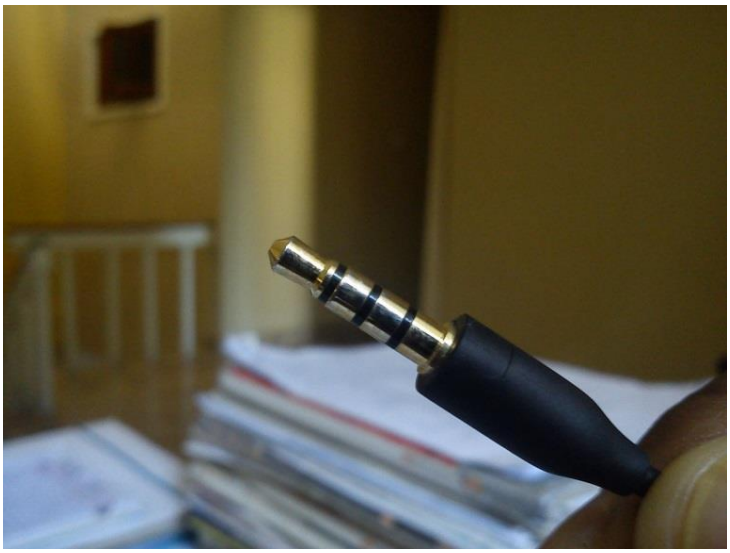
Suara direkam menggunakan *tablet smartphone* berbasis *Android* dengan aplikasi *Smart Voice Recorder* buatan *SmartMob* yang bisa diunduh secara gratis melalui *Google Play Store*. Aplikasi ini memiliki fitur *Skip Silence, microphone gain calibration tool, wave/pcm encoding with adjustable sample rate (8-44 kHz),* dan beberapa fitur lainnya.



Sumber: *Google Play Store*

Gambar 3.2. Tampilan Aplikasi *Smart Voice Recorder*

Supaya kualitas suara lebih baik maka *smartphone* dapat dikoneksikan dengan *earphone*, *headset*, ataupun *microphone* (semua menggunakan 3.5mm 4-*conductor TRRS phone connector).*

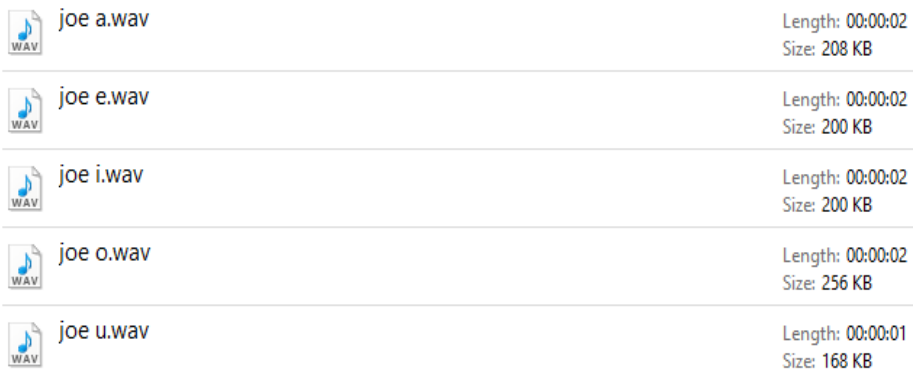


Gambar 3.3. Contoh 3.5mm 4-*conductor TRRS phone connector*

Situasi perekaman adalah di ruangan sepi dan bebas *noise* untuk menghindari adanya suara-suara yang tidak diinginkan. Waktu maksimal untuk merekam suara adalah 3 detik.

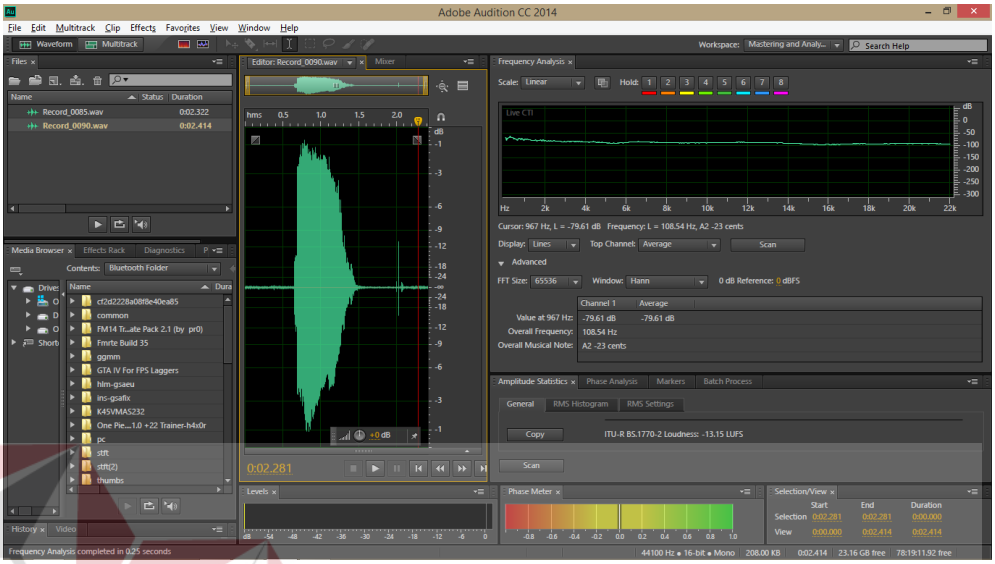
3.1.3 Hasil Rekaman

Suara yang direkam adalah saat pengucapan 5 huruf vokal (a, e, i, o, u) dimana satu huruf adalah satu *file* rekaman. *File* rekaman menggunakan *format* (.wav) dan *sample rate* 44100 Hz. Masing masing huruf direkam dengan waktu maksimal 3 detik.

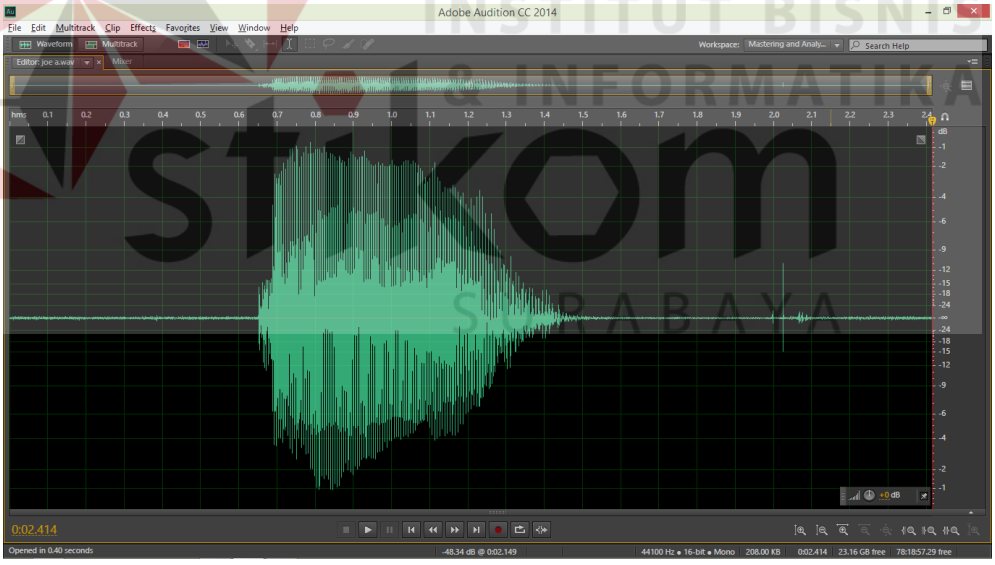


Gambar 3.4. Hasil rekaman berupa *file* ber-*format* (.wav)

File tersebut kemudian diubah atau dipotong secara manual menggunakan perangkat lunak *Adobe Audition* untuk menghilangkan rekaman kosong.



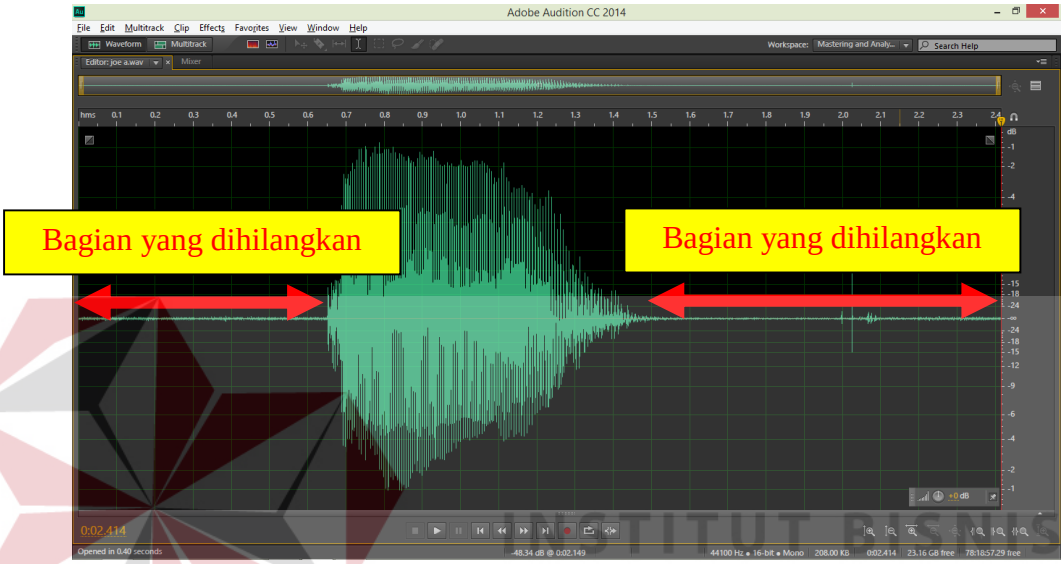
Gambar 3.5. Tampilan *software Adobe Audition* 2014



Gambar 3.6. Contoh tampilan sinyal suara pria “a” dalam domain waktu.

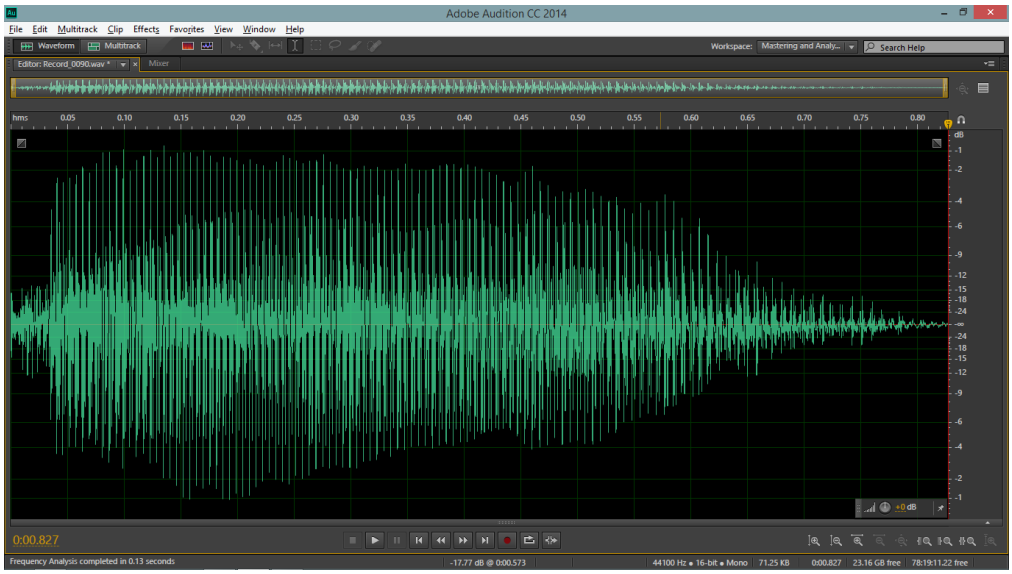
Gambar 3.6. menunjukkan hasil rekaman suara laki-laki mengucapkan “a” dalam bentuk sinyal dalam *domain* waktu. Suara direkam dalam waktu 2.414 sekon. Terkadang pada saat rekaman, seseorang baru mulai berbicara pada

waktu tertentu sehingga terdapat waktu jeda. Untuk mendapatkan sinyal yang diinginkan, maka perlu dilakukan *cropping* atau menghilangkan waktu jeda tersebut atau bagian sinyal yang tidak perlu (dalam hal ini suara hembusan nafas maupun *noise* lainnya). Seperti terlihat pada gambar 3.7.



Gambar 3.7. Bagian-bagian sinyal yang akan dipotong.

Contohnya pada hasil rekaman ini, sinyal suara manusia baru terbentuk saat 0.653 s dan berakhir pada 1.480 s, sehingga sinyal suara yang diambil adalah sinyal pada rentang 0.653 s dan 1.480 s, seperti ditunjukkan pada gambar 3.8. Dengan demikian waktu keseluruhan sinyal suara adalah 0.827 s seperti terlihat pada gambar.

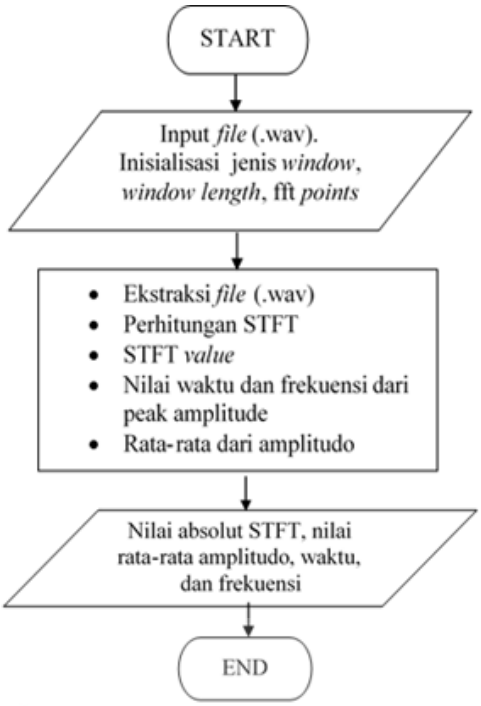


Gambar 3.8. Contoh bentuk sinyal suara setelah *cropping*

Setelah proses diatas, maka sinyal hasil *crop* tersebut disimpan menjadi *file* berformat (.wav) dengan *sample type* 44100 Hz Mono, 16-bit. Panjang waktu rekaman setelah *crop* hanya dibatasi sampai 1 detik. Sehingga untuk satu orang memiliki 5 *file* berformat (.wav).

3.1.4 Ekstraksi Ciri

Pada blok ini, *file* suara yang sudah diubah akan dianalisa menggunakan perangkat lunak *Matlab* untuk mendapatkan nilai STFT. Adapun alur proses digambarkan seperti pada *flowchart* sederhana dibawah.



Gambar 3.9. Diagram alir proses ekstraksi ciri STFT

File suara yang telah di-*crop* berformat (.wav) akan di masukkan ke dalam matlab untuk diekstraksi dengan perintah `[x,fs]=audioread('file_location')`, sehingga dari *file* tersebut dapat diperoleh nilai waktu (`x`), dan *sample rate* (`fs`). Dari nilai x tersebut kemudian diketahui panjang dari sinyal rekaman tersebut (`xlen = length(x)`). Langkah selanjutnya adalah menentukan nilai parameter untuk *input* fungsi STFT, yaitu *window length*, *hop size*, dan *fft points*.

`wlen = 2048;` Panjang *window*

`h = wlen/4;` Jumlah sampel antara masing-masing *window*

`nfft = 2*wlen;` Banyaknya poin FFT.

Adapun nilai `wlen` adalah bilangan kelipatan 2 berpangkat (power of 2) untuk memudahkan perhitungan. Dalam hal ini, *window length* ditentukan sebesar 2048. Nilai tersebut dipilih untuk mendapatkan resolusi frekuensi dan

resolusi waktu yang bagus. Resolusi frekuensi yang bagus adalah resolusi dengan jarak antar *window* yang lebar, artinya *range* frekuensi semakin kecil dan memberikan nilai yang semakin akurat. Namun dengan lebarnya *window*, maka resolusi waktu pun menjadi tidak bagus. Untuk lebih jelasnya dapat dilihat pada pembahasan di bab sebelumnya. Oleh karena itu diperlukan nilai dari *window length* yang dapat mengoptimalkan *range* dari kedua resolusi.

Adapun jenis *window* yang digunakan pada penelitian ini adalah *Hann* atau *Hanning window* dimana nilai koefisiennya bisa didapatkan dengan rumus berikut.

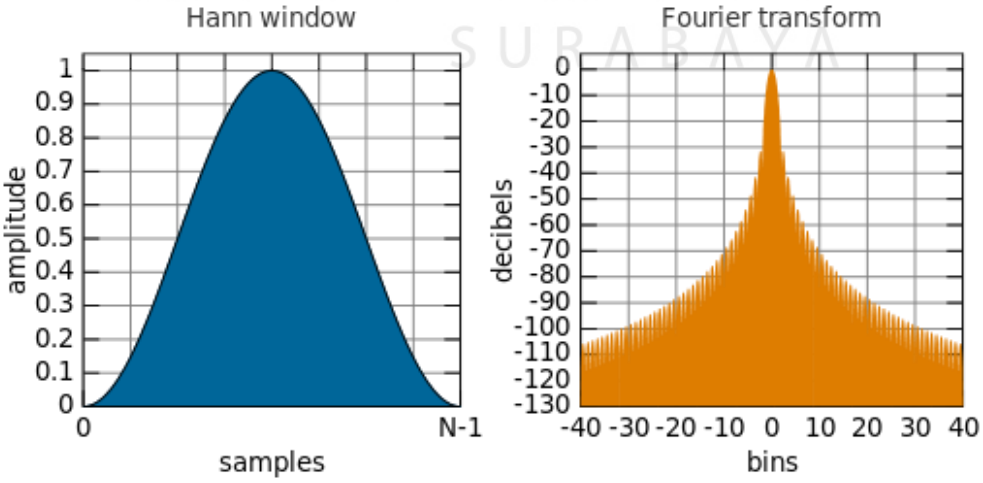
(1).

$$w(n) = \frac{1}{2}\left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right)$$

atau (2).

$$w(n) = sin^2\left(\frac{\pi n}{N-1}\right)$$

Dimana n = jumlah sampel pada sinyal, dan N adalah panjang *window*.

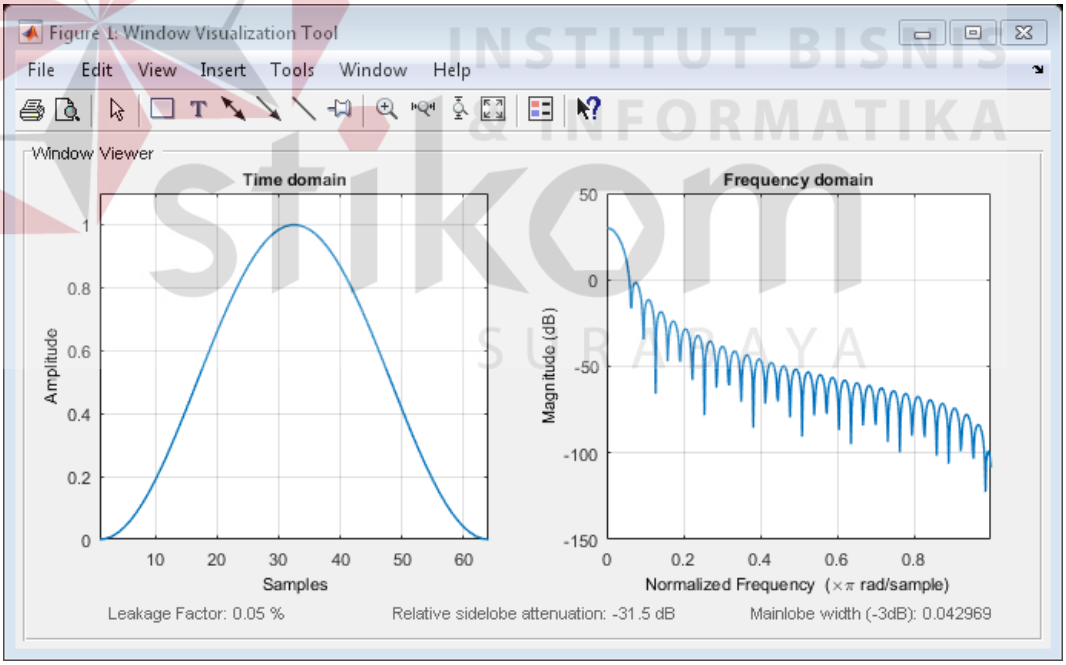


Gambar 3.10. Fungsi *Hann* (kiri), dan respon frekuensinya (kanan)

Hann window atau juga disebut jendela kosinus yang ditinggikan biasanya dipakai sebagai fungsi *window* dalam pemrosesan sinyal digital untuk menjalankan transformasi *fourier* dimana ujung dari kosinus menyentuh nilai nol, sehingga *side-lobe* berada pada 18 dB per oktaf. Adapun keunggulan dari *hann window* adalah sangat rendahnya artifak distorsi atau *aliasing* dan lebarnya *main-lobe* (*lobe* dimana di terdapat energi maksimal).

Pada aplikasi *Matlab*, fungsi *hann window* dapat ditulis dengan sintaks `w = hann(L)`, dimana L adalah panjang dari *window* tersebut.

Berikut adalah contoh visualisasi 64-point *hann window* pada *Matlab* dengan menggunakan *Window Visualisation Tool*. Dengan sintaks sebagai berikut: `L = 64; wvtool(hann(L))`

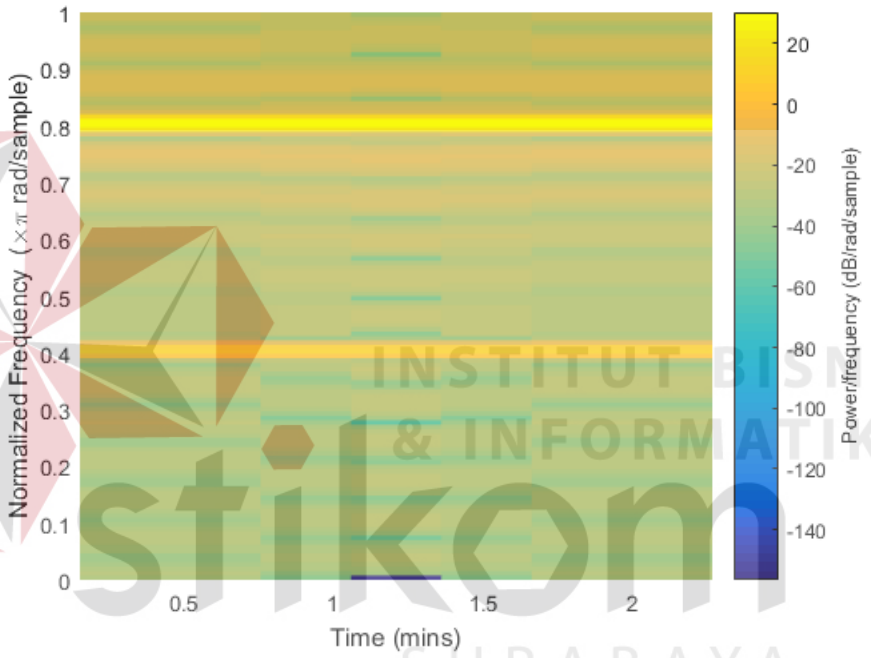


Gambar 3.11. Visualisasi 64-point *hann window* pada *Matlab*

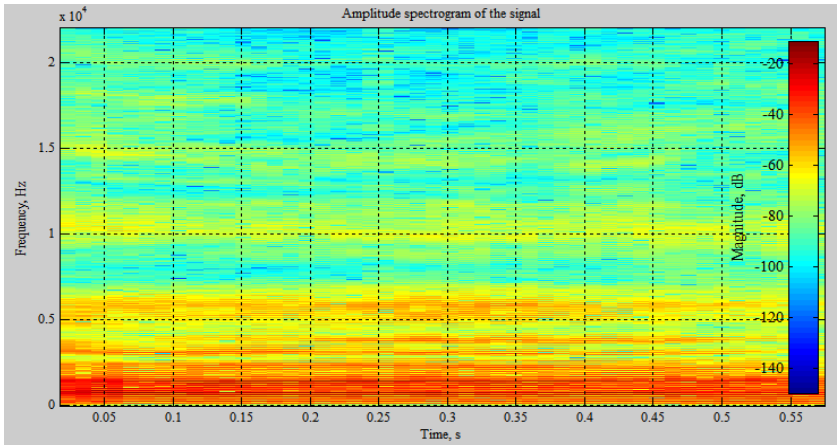
Kemudian setelah menentukan *window*, maka dapat dilakukan perhitungan STFT, dimana matriks STFT akan merepresentasikan waktu pada kolom (sumbu x), dan frekuensi pada baris (sumbu y) dalam sebuah spektrogram atau

sebuah grafik yang memberikan informasi tentang perubahan gelombang dalam rentang waktu, frekuensi, dan intensitas amplitudo. Intensitas amplitudo pada suatu frekuensi dan pada suatu waktu (waktu, frekuensi) di dalam spektrogram dinyatakan dengan nilai warna tertentu (grayscale atau RGB).

Spektrogram pada *Matlab* pada umumnya dapat dibuat dengan sintaks `s=spectrogram(x,window,noverlap,nfft).`



Gambar 3.12. Contoh grafik spektrogram pada *Matlab*



Gambar 3.13. Contoh spektrogram pada penelitian ini.

Gambar 3.12. dan gambar 3.13. mencontohkan bentuk spektrogram pada umumnya dimana sumbu x merepresentasikan waktu, sumbu y merepresentasikan frekuensi, dan intensitas amplitudo atau energi (umumnya dalam dB) dinyatakan dalam spektrum warna tertentu. Pada spektrogram, panjang *window* dapat mempengaruhi intensitas warna dan resolusi dari waktu dan frekuensi.

Setelah menghasilkan spektrogram, berikutnya akan diambil nilai STFT, nilai frekuensi dan waktu pada saat energi terbesar, dan nilai rata-rata energi dari sebuah sinyal suara untuk kemudian akan dilatih dan diuji menggunakan jaringan saraf tiruan *backpropagation*. Keempat nilai tersebut digunakan karena metode STFT menghasilkan suatu nilai waktu dan frekuensi, dan juga amplitude. Nilai STFT adalah suatu bilangan kompleks yang terdiri dari bilangan *real* dan *imaginer* yang merepresentasikan suatu nilai dalam domain waktu dan frekuensi. Agar nilai kompleks tersebut dapat dihitung, maka nilai tersebut diabsolutkan dengan rumus $absolut = \sqrt{real^2 + imaginer^2}$.

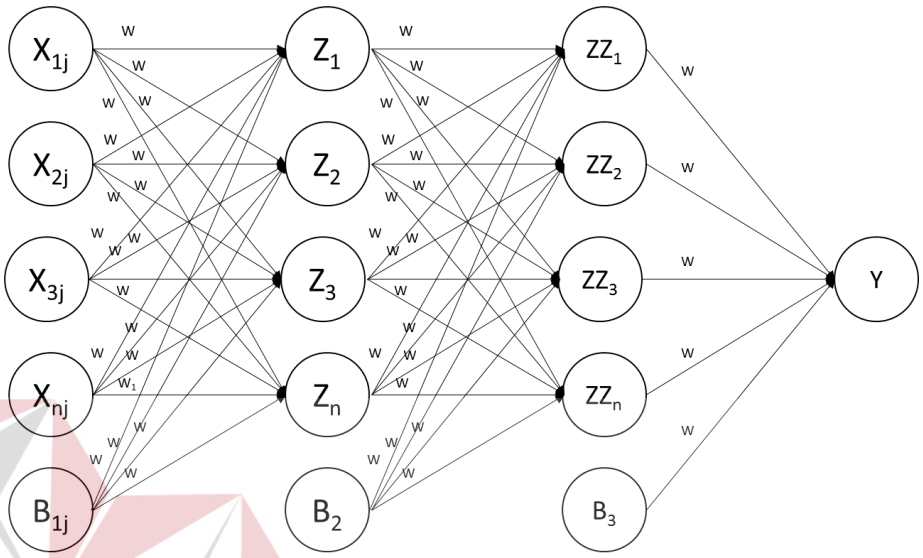
Dan nilai amplitudo/magnitude adalah nilai yang menunjukkan kekuatan sinyal dalam desibel (dB). Nilai awal dari amplitudo adalah berupa tegangan (volt), dan untuk mengubahnya kedalam bentuk desibel digunakan rumus 20*log10(amplitudo).

3.1.5 Pengujian

Pengujian yang dilakukan adalah menggunakan metode jaringan saraf tiruan *backpropagation* dengan input nilai STFT, nilai frekuensi dan waktu saat amplitude tertinggi, dan nilai rata-rata amplitudo pada sebuah sinyal suara. *Neuron input* berjumlah sebanyak 20 untuk satu data pelatihan, dan banyaknya

responden adalah sebanyak 30 (laki-laki 15, perempuan 15) seperti terlihat pada gambar 3.14.

Arsitektur *backpropagation* yang akan dibuat adalah sebagai berikut:



Gambar 3.14. Arsitektur *backpropagation* pada penelitian ini

Pada gambar 3.14. Nilai X1, X2, X3, X4, adalah nilai untuk fitur suara "a" pada data responden pertama, kemudian X5, X6, X7, X8, adalah nilai fitur suara "e" pada data responden pertama demikian seterusnya untuk huruf vokal i, o, dan u. Variabel j adalah banyaknya data pelatihan pada *backpropagation* dimana pada tabel 3.1. menunjukkan data pelatihan pada *backpropagation* dan nilai target yang ingin dicapai pada pelatihan ini.

Tabel 3.1. Data pelatihan *backpropagation* dan nilai target yang ingin dicapai

Data pelatihan (j)	Keterangan	Nilai target
1	Data pelatihan responden perempuan 1	1
2	Data pelatihan responden perempuan 2	1
3	Data pelatihan responden perempuan 3	1
4	Data pelatihan responden perempuan 4	1
5	Data pelatihan responden perempuan 5	1

6	Data pelatihan responden perempuan 6	1
7	Data pelatihan responden perempuan 7	1
8	Data pelatihan responden perempuan 8	1
9	Data pelatihan responden perempuan 9	1
10	Data pelatihan responden perempuan 10	1
11	Data pelatihan responden perempuan 11	1
12	Data pelatihan responden perempuan 12	1
13	Data pelatihan responden perempuan 13	1
14	Data pelatihan responden perempuan 14	1
15	Data pelatihan responden perempuan 15	1
16	Data pelatihan responden laki-laki 1	0
17	Data pelatihan responden laki-laki 2	0
18	Data pelatihan responden laki-laki 3	0
19	Data pelatihan responden laki-laki 4	0
20	Data pelatihan responden laki-laki 5	0
21	Data pelatihan responden laki-laki 6	0
22	Data pelatihan responden laki-laki 7	0
23	Data pelatihan responden laki-laki 8	0
24	Data pelatihan responden laki-laki 9	0
25	Data pelatihan responden laki-laki 10	0
26	Data pelatihan responden laki-laki 11	0
27	Data pelatihan responden laki-laki 12	0
28	Data pelatihan responden laki-laki 13	0
29	Data pelatihan responden laki-laki 14	0
30	Data pelatihan responden laki-laki 15	0

Pada tabel 3.1., nilai 1 dalam penelitian ini adalah nilai permisalan yang menandakan data perempuan dan nilai 0 adalah data laki-laki.

Masing-masing data responden mempunyai 20 fitur suara yaitu 4 fitur suara pada masing masing huruf vokal, seperti pada tabel 3.2. yang menunjukkan nilai *neuron input* dari *backpropagation* pada masing-masing responden.

Tabel 3.2. Keterangan dari nilai *neuron input backpropagation*

Neuron Input	Keterangan
X1	STFT_value "a"
X2	Rata-rata Amplitudo "a"
X3	Waktu "a"
X4	Frekuensi "a"
X5	STFT_value “e”
X6	Rata-rata Amplitudo "e"
X7	Waktu "e"
X8	Frekuensi "e"
X9	STFT_value "i"
X10	Rata-rata Amplitudo "i"
X11	Waktu "i"
X12	Frekuensi "i"
X13	STFT_value "o"
X14	Rata-rata Amplitudo "o"
X15	Waktu "o"
X16	Frekuensi "o"
X17	STFT_value "u"
X18	Rata-rata Amplitudo "u"
X19	Waktu "u"
X20	Frekuensi "u"

Tabel 3.2. menunjukkan *neuron input* pada *backpropagation* untuk masing-masing responden. Untuk responden berikutnya atau data pelatihan yang kedua seperti pada tabel 3.1, maka STFT_value, Rata-rata Amplitudo, Waktu, dan Frekuensi pada tabel 3.2. adalah untuk responden kedua. Begitu juga untuk responden ketiga dan seterusnya, nilai X1 – X20 adalah nilai dari responden yang dilatih saat itu.

Nilai awal bobot B1, B2, B3, dan nilai W pada masing masing *layer* adalah nilai random dari -1 sampai 1. Nilai target adalah nilai 1 sebanyak 15 (untuk input perempuan), dan nilai 0 sebanyak 15 (untuk input laki-laki).

Output dari gambar 3.14. adalah sebuah nilai antara 0 sampai 1. Nilai tersebut kemudian di *threshold* untuk membatasi data pelatihan suara perempuan atau data pelatihan suara laki-laki. Nilai yang lebih besar atau sama dengan 0.5 (>=0.5) akan dianggap sebagai perempuan, dan nilai yang lebih kecil dari 0.5 akan dianggap sebagai laki-laki (<0.5).

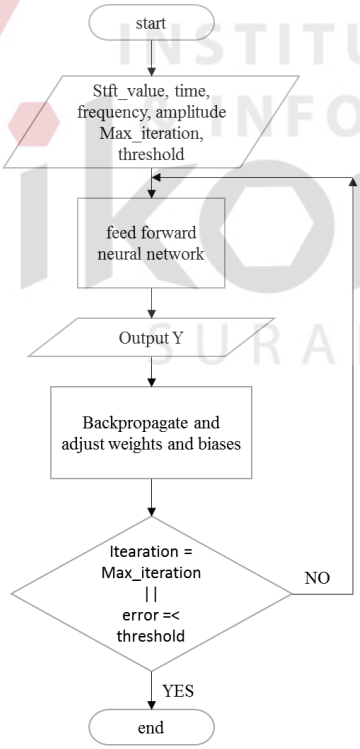
Pada gambar 3.14., setiap *input* Xnj dan bias B1 akan dikalikan dengan Wi pada *layer input*, dan kemudian hasilnya akan menjadi *input* bagi masing-masing *neuron* Zi, begitu juga dengan *bias*; dan diulang sebanyak jumlah data pelatihan (30). Kemudian hasil inputan pada Zi akan diaktivasi dengan fungsi *sigmoid* biner untuk menghasilkan nilai antara 0 sampai 1.

Setelah masing-masing *neuron* diaktivasi, maka *output* dari Zi akan menjadi *input* ZZi. Untuk melanjutkan perhitungan ke *hidden layer 2*, dan *output*, maka *output* dari Zi dan juga *bias* B2 dikalikan pada W pada *hidden layer* 1 untuk dimasukkan pada *hidden layer 2* (ZZi). Masukkan dari *Hidden layer* 1 (Zi) nantinya akan diaktivasi dahulu sebelum menjadi *input* ZZi. *Input* pada ZZi nantinya akan digunakan untuk menghitung nilai keluaran pada alur maju dengan cara mengalikan ZZi dan bias B3 dengan bobot Wi pada *hidden layer 2.* Hasil dari perkalian ini akan menghasilkan nilai output Y yang nilainya akan diaktivasi untuk perhitungan alur mundur.

Setelah menghasilkan keluaran Y yang telah diaktivasi, maka akan dicari selisih (error) dari target awal dengan Y, dan kemudian menghitung koreksi

bobot dan bobot bias dan mengubah bobot garis yang berhubungan langsung dengan unit keluaran. Dengan cara yang sama, dihitung faktor *error* di setiap unit di *hidden layer* sebagai dasar perubahan bobot semua garis yang berasal dari unit tersembunyi di *layer* di bawahnya. Demikian seterusnya hingga semua faktor *error* di unit *hidden* yang berhubungan langsung dengan unit masukan dihitung. Umumnya kondisi penghentian yang dipakai adalah jumlah iterasi atau kesalahan. Toleransi kesalahan yang digunakan di penelitian ini adalah 0.01.

Nilai learning rate (α) adalah adalah laju pembelajaran, semakin besar learning rate akan berpengaruh pada semakin besarnya langkah pembelajaran. Sehingga dalam penelitian ini, learning rate yang digunakan adalah 0.1



Gambar 3.15. Diagram alir proses *backpropagation*

Gambar 3.15. menjelaskan alur kerja dari *backpropagation* pada penelitian ini. Yang menjadi input adalah nilai X1 – X20 yang merupakan nilai

ekstraksi ciri, seperti pada tabel 3.2. Langkah yang dijalankan adalah alur maju (feed forward) yaitu untuk mencari nilai *error* selisih antara nilai target awal dengan nilai keluaran (t–y). Selisih error tersebut akan digunakan untuk alur mundur (backpropagate) untuk mengubah nilai bobot pada *neuron hidden* dan juga *bias*. Proses akan berlangsung selama jumlah iterasi belum terpenuhi atau nilai *error* masih lebih besar dari nilai toleransi *error*.