pattern layer dapat ditunjukkan pada persamaan 8.

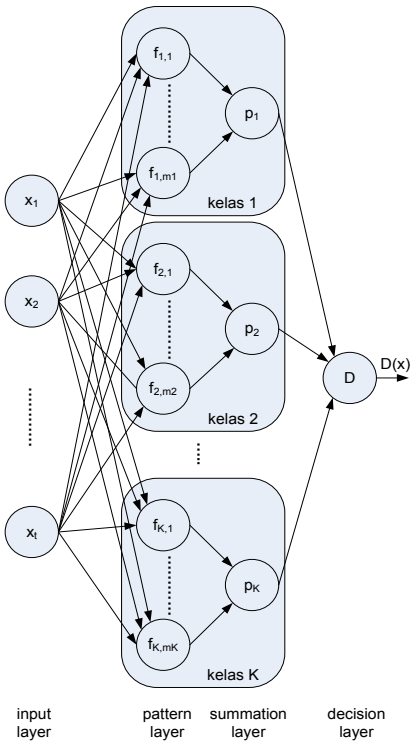
$$f_i(x) = \prod_{j=1}^{d} k\left(\frac{x_j - x_{ij}}{h_j}\right) \qquad (8)$$

Keterangan:
k(z) = $e^{-0.5 \times z^2}$
d = banyaknya data pada satu *pattern layer*
x_j = *input* data uji ke-*j*
x_{ij} = *pattern* ke-*i* data ke-*j*
h_j = *smoothing parameter* (α x simpangan baku ke-*j* x $n^{-1/5}$)
i = 1, 2 sampai n
j = 1, 2 sampai d
n = banyaknya *pattern* pada satu kelas

3. *Summation Layer*. *Layer* ini menghasilkan peluang untuk satu kelas. Peluang tersebut didapat dari penjumlahan *pattern layer* pada kelas tersebut dan hasilnya dibagi dengan $(2\pi)^{d/2} h_i h_2 ... h_d n$. Nilai $h_i h_2 ... h_d$ adalah nilai *smoothing* dari kelas tersebut. Persamaan untuk menghitung peluang tersebut adalah :

$$P(x) = \frac{1}{(2\pi)^{\frac{d}{2}} h_1 h_2 ... h_d n} \sum_{i=1}^{n} \left(f_i(x)\right) \qquad (9)$$

4. *Decision Layer*. *Layer* ini membandingkan hasil peluang pada setiap kelas. Selanjutnya, *input* data dimasukkan dalam kelas yang memiliki nilai peluang terbesar.



Gambar 5 Bagan Model PNN (Ganchev 2005).

Fonem

Fonem merupakan satuan bunyi terkecil yang mampu menunjukkan kontras makna (Depdikbud 2003). Fonem dibagi menjadi dua yaitu:

1. Fonem vokal merupakan bunyi ujaran akibat adanya udara yang keluar dari paru-paru tidak terkena hambatan atau halangan. Jumlah fonem vokal ada lima yaitu a, i, u, e, dan o.
2. Fonem konsonan merupakan bunyi ujaran akibat adanya udara yang keluar dari paru-paru mendapatkan hambatan atau halangan. Jumlah fonem konsonan ada 21 buah yaitu b, c, d, f, g, h, j, k, l, m, n, p, q, r, s, t, v, w, x, y, dan z.

METODE PENELITIAN

Kerangka Pemikiran

Penelitian ini dilakukan dengan mengambil data suara dari satu orang dengan mengucapkan satu kata sebanyak 16 kali. Bagian *silence* pada data suara akan dihapus. Proses selanjutnya yaitu normalisasi dan segmentasi. Data dibagi menjadi dua yaitu data latih dan data uji. Kemudian data akan diolah dengan proses MFCC. Hasil MFCC dirata-ratakan pada setiap data suara.

Data Suara

Penelitian ini akan menggunakan data yang telah didigitalisasi dan direkam dari satu orang pembicara yang mengucapkan satu kata sebanyak 16 kali. Setiap suara direkam dengan rentang waktu satu detik dengan *sampling rate* 12000 Hz. Kata yang digunakan dalam penelitian ini dapat dilihat pada Tabel 1.

Tabel 1 Kata dalam penelitian.

Kata	Fonem	Kata	Fonem
coba	/a/,/b/,/c/,/o/	quran	/a/,/n/,/q/,/r/,/u/
fana	/a/,/f/,/n/	tip-x	/i/,/p/,/t/,/x/
gajah	/a/,/g/,/h/,/j/	visa	/a/,/i/,/s/,/v/
jaya	/a/,/j/,/y/	weda	/a/,/d/,/e/,/w/
malu	/a/,/l/,/m/,/u/	zakat	/a/,/k/,/t/,/z/
pacu	/a/,/c/,/p/,/u/		

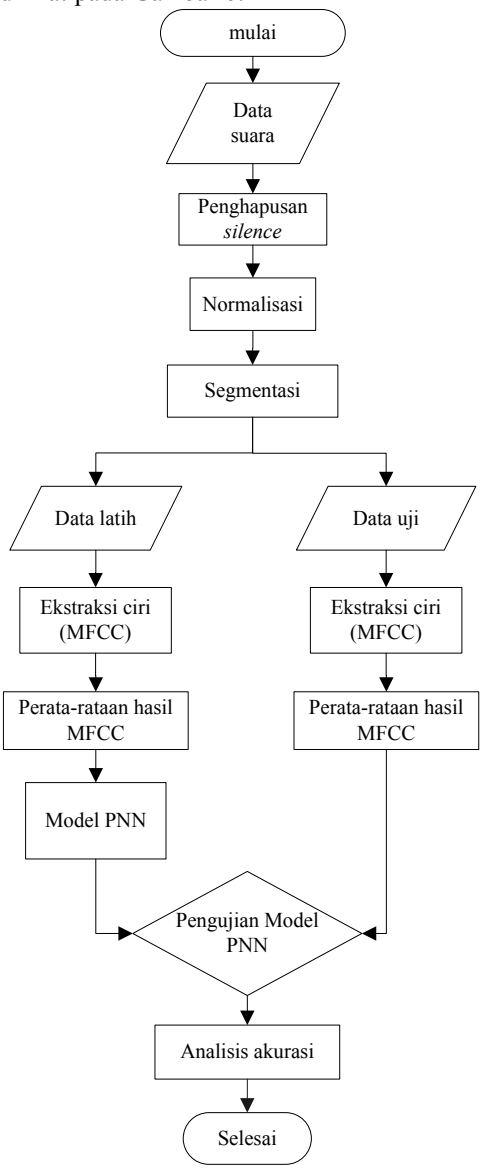
Penghapusan S*ilence*

Tahap ini sinyal yang memiliki bagian yang *silence* akan dihapus baik di depan atau di belakang menggunakan *Audacity 1.2.6* pada data latih dan data uji.

Normalisasi

Normalisasi dilakukan dengan mengabsolutkan nilai-nilai data suara dan mencari nilai maksimumnya. Selanjutnya, setiap nilai data tersebut dibagi dengan nilai maksimumnya. Hal ini dilakukan agar menormalkan suara sehingga memiliki amplitudo maksimum satu dan minimum minus satu.

Untuk lebih jelas metode penelitian ini dapat dilihat pada Gambar 6.



Gambar 6 Diagram Alur Penelitian.

Segmentasi Sinyal

Tahap segmentasi sinyal merupakan tahap dimana setiap fonem dari kata-kata yang ada akan dipisahkan secara manual menggunakan *Audacity*. Hasil dari banyaknya segmentasi dari kata yang digunakan sebagaimana dijelaskan pada Tabel 1 maka jumlah data dari tiap fonem dijelaskan dalam Tabel 2.

Tabel 2 Jumlah data tiap fonem.

Fonem	Jumlah	Fonem	Jumlah
a	224	n	32
b	16	o	16
c	32	p	32
d	16	q	16
e	16	r	16
f	16	s	16
g	16	t	32
h	16	u	48
i	32	v	16
j	32	w	16
k	16	x	16
l	16	y	16
m	16	z	16

Data Latih dan Data Uji

Data dibagi menjadi data latih dan data uji. Proporsi data latih dan data uji yaitu 75%:25%. Data uji yang digunakan yaitu tanpa *noise* dan data uji yang ditambah *noise* 30 dB, 20 dB, dan 10 dB. *Noise* yang ditambahkan pada sinyal perekaman suara dalam penelitian ini menggunakan *Gaussian white noise* dengan maksud untuk mengetahui sinyal mana yang memberikan generalisasi yang lebih baik pada testing identifikasi yang dibuat terhadap sinyal yang tanpa ditambahkan *noise*. Lebih detail banyaknya data latih dan data uji untuk masing-masing fonem dapat dilihat pada Tabel 3.

Tabel 3 Jumlah data latih dan data uji.

Fonem	Data latih	Data uji	Fonem	Data latih	Data uji
a	168	56	n	24	8
b	12	4	o	12	4
c	24	8	p	24	8
d	12	4	q	12	4
e	12	4	r	12	4
f	12	4	s	12	4
g	12	4	t	24	8
h	12	4	u	36	12
i	24	8	v	12	4
j	24	8	w	12	4
k	12	4	x	12	4
l	12	4	y	12	4
m	12	4	z	12	4

Ekstraksi Ciri (MFCC)

Tahap ekstraksi ciri merupakan tahap untuk menentukan vektor penciri dan biasanya menggunakan koefisien *cepstral*. Proses yang dilakukan pada tahap ini adalah *Framing*, *windowing*, *Fast Fourier Transform*, *Mel-Frequency Wrapping*, dan *Cepstrum*. Proses MFCC dilakukan dengan menggunakan *toolbox* yang tersedia yaitu *Auditory Toolbox* yang dikembangkan oleh *Slaney* (1998) dimana membutuhkan lima parameter yaitu :

1. *Input* yaitu suara yang merupakan masukan dari setiap pembicara.
2. *Sampling rate* yaitu banyaknya nilai yang diambil dari setiap detik. Penelitian ini menggunakan *sampling rate* sebesar 12000 Hz.
3. *Time frame* yaitu waktu yang digunakan untuk satu *frame* (dalam milidetik). *Time frame* yang digunakan adalah 30 ms.
4. *Lap* yaitu *overlaping* yang diinginkan (harus kurang dari 100%). *Lap* yang digunakan pada penelitian ini adalah 0%, 25%, dan 50%.
5. *Cepstral coefficient* yaitu jumlah *cepstrum* yang diinginkan sebagai *output*. *Cepstral coefficient* yang digunakan sebanyak 13, 20, dan 26.

Setiap data suara dilakukan proses *framing* dimana masing-masing *frame* berukuran 30 ms dengan *overlap* 0%, 25%, dan 50% tanpa *noise*. Penelitian ini menggunakan 13, 20, dan 26 koefisien *mel cepstrum* untuk masing-masing *frame*. Hasil matriks ini yang merupakan masukan untuk *Probabilistic Neural Network* (PNN).

Perata-rataan Hasil MFCC

MFCC memiliki hasil berupa matriks ciri $n \times k$, n adalah koefisien dan k adalah jumlah *frame*. Agar ukuran matriks sama untuk setiap suara yaitu berbentuk $n \times 1$ untuk setiap suara, maka dilakukan proses perata-rataan koefisien pada setiap baris.

Pemodelan PNN

Data uji digunakan sebagai *input* data. *Input* data tersebut diidentifikasikan dengan *pattern layer* pada Persamaan 8. Parameter h pada Persamaan 8 digunakan nilai 1,14 × (simpangan baku) × $n^{-1/5}$. Nilai $f_i(x)$ ialah nilai hasil *pattern layer* ke i, dimana i=1, 2 sampai banyaknya observasi pada satu kelas. Setelah memperoleh selisih jarak antara nilai data input dengan data pada *pattern* layer, maka nilai tersebut dibagi dengan nilai *smoothing parameter*. Nilai *smoothing* h_j didapat dari simpangan baku data setiap *pattern* ke j=1, 2 sampai jumlah koefisien yang digunakan.

Pengujian Model PNN

Setiap data uji (matriks $n \times 1$) dimasukkan ke dalam setiap kelas pada model PNN. Perhitungan pada pengujian setiap kelas menggunakan Persamaan 9, sehingga nilai peluang p(x) diperoleh dari setiap kelas pada pengujian model PNN. Nilai p(x) terbesar pada satu kelas merupakan pemenang, sehingga *input* data dikenali sebagai kelas tersebut.

Perhitungan Nilai Akurasi

Perhitungan dilakukan dengan membandingkan banyaknya hasil kata yang benar dengan kata yang diuji. Persentase tingkat akurasi dihitung dengan fungsi berikut:

$$\text{Hasil} = \frac{\sum data\ yang\ benar}{\sum data\ yang\ diuji}\ x\ 100\% \quad (10)$$

Lingkungan Pengembangan

Sistem ini diimplementasikan dengan MATLAB 7.0 yang dijalankan pada sistem operasi Windows 7, sedangkan perangkat keras yang digunakan adalah Intel Atom M 1.66 GHz, 1 GB RAM.s.

HASIL DAN PEMBAHASAN

Kata yang digunakan sebanyak sebelas yang masing-masing direkam sebanyak 16 kali. Data tersebut masih berupa data suara kotor karena masih terdapat *silent*, sehingga perlu dibersihkan dengan menghilangkan *silent* setelah itu dilakukan normalisasi. Proses segmentasi dilakukan secara manual sehingga membutuhkan waktu yang cukup lama. Segmentasi secara manual menghasilkan data suara berjumlah 752 yang meliputi 26 fonem. Data fonem yang dihasilkan dari segmentasi kemudian ditetapkan 75% sebagai data latih dan 25% sebagai data uji sehingga penelitian ini menggunakan data sebanyak 564 untuk data latih dan 188 untuk data uji. Kemudian data diekstraksi menggunakan MFCC yang dimplementasi menggunakan fungsi yang sudah tersedia yang dikembangkan oleh *Slaney* pada tahun 1998. Seperti yang telah dijelaskan sebelumnya, *frame* yang digunakan sebesar 30 ms, dimana terjadi *overlap* antar *frame* sebesar 0%, 25%, dan 50%, serta *cepstral coefficient* yang digunakan sebesar 13, 20, dan 26 untuk setiap *frame*. Data pelatihan yang telah diolah dan dilakukan praproses digunakan untuk membangun model pengenalan kata dengan