BAB II

LANDASAN TEORI

2.1 Jenis Suara Vokal Manusia

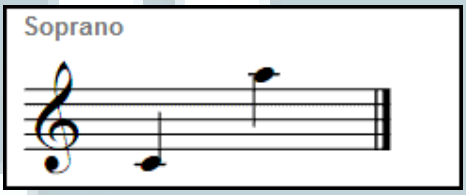
Menurut Kamus Besar Bahasa Indonesia (KBBI) suara adalah suatu bunyi yang dihasilkan dari mulut manusia. Bunyi ini dikeluarkan ketika seseorang melakukan percakapan, menyanyi, tertawa, dan menangis. Suara ini berasal dari pita suara manusia yang bergetar. Karena adanya getaran tersebut, maka terjadilah suatu bunyi/suara.

Paduan suara berasal dari kata suara yang terpadu yang terdiri dari paduan suara besar atau kecil. Dengan demikian paduan suara adalah bernyanyi secara serentak, terpadu dengan keselarasan volume yang baik dan terkontrol, mengikuti keselarasan harmoni dan juga memberikan interprestasi yang sedekat-dekatnya pada kemauan komposer (Harahap, 2005).

Suara manusia, terutama suara vokal terbagi menjadi tiga yaitu suara wanita dewasa, suara pria dewasa dan suara anak-anak. Suara tinggi wanita disebut *sopran,* suara sedang wanita yang disebut *mezzo sopran,* suara rendah wanita yang disebut *alto.* Sedangkan untuk pria, suara pria tinggi disebut *tenor,* suara pria sedang yang disebut *bariton,* dan terakhir suara pria rendah yang disebut *bass*. Pada suara anak-anak dibagi menjadi dua yakni suara tinggi dan rendah (Runtuwene, 2013). Umumnya, pembagian suara ini dilakukan ketika terdapat suatu kelompok paduan suara sehingga ragam suara yang dihasilkan dari masing-masing jenis suara vokal menyatu dan menghasilkan suara yang indah.

2.1.1 Sopran

Sopran (*soprano*) merupakan salah satu dari jenis suara manusia yang berjenis kelamin perempuan. Jenis suara ini merupakan jenis suara wanita yang berada pada nada tinggi, yakni di antara nada C^4 sampai A^5 (pada alat musik *keyboard*) atau berada pada 262Hz – 1047Hz pada frekuensi logaritmik (Case, 2007). Jenis suara ini dapat ditingkatkan lagi tinggi nadanya, tergantung dari seberapa banyak latihan yang dilakukan (Simanungkalit, 2008).

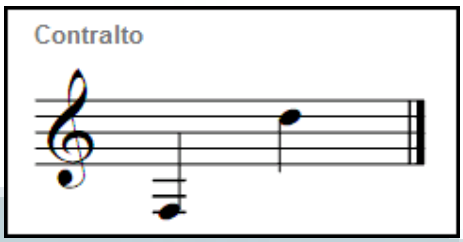


Gambar 2.1 *Vocal Range* untuk Suara *Sopran* (Randel, 1986)

2.1.2 Alto

Alto atau yang juga dikenal dengan istilah *contralto* merupakan salah satu jenis suara manusia berjenis kelamin perempuan. Jenis suara ini memiliki nada rendah, yakni dari F^3 sampai D^5 (pada alat musik keyboard) atau berada pada 165Hz – 659Hz pada frekuensi logaritmik (Case, 2007). Suara *alto* ini memiliki range nada yang hampir sama dengan suara laki-laki dengan tinggi nada tertentu yang dikenal dengan istilah *counter tenor*.

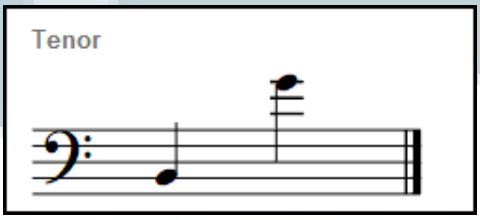
Ciri dari suara *alto* ini adalah harus dinyanyikan dengan nada rendah, berat, dan dalam. Akan tetapi dari beberapa ahli mengatakan bahwa suara *alto* harus dinyanyikan dengan penuh wibawa (Simanungkalit, 2008).



Gambar 2.2 *Vocal Range* untuk Suara *Alto* (Randel, 1986)

2.1.3 Tenor

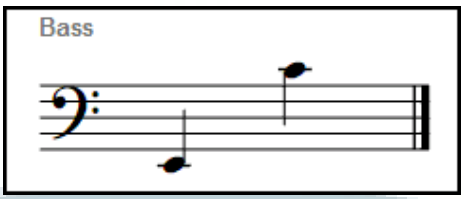
Tenor merupakan salah satu jenis suara manusia yang berjenis kelamin laki-laki. Jenis suara ini memiliki nada yang tinggi untuk laki-laki di mana berada pada range B^2 sampai G^4 (pada alat musik keyboard) atau berada pada 123Hz – 523Hz pada frekuensi logaritmik (Case, 2007). Suara jenis ini memiliki oktaf yang lebih rendah satu oktaf dari pada suara sopran pada jenis suara perempuan (Simanungkalit, 2008).



Gambar 2.3 *Vocal Range* untuk Suara *Tenor* (Randel, 1986)

2.1.4 Bass

Bass merupakan salah satu jenis suara manusia berjenis kelamin laki-laki. Jenis suara ini memiliki nada paling rendah dari suara-suara pria yang ada yakni berada dalam range nada E^2 sampai C^4 (pada alat musik keyboard) (Simanungkalit, 2008) atau berada pada 65Hz – 330Hz pada frekuensi logaritmik (Case, 2007).



Gambar 2.4 *Vocal Range* untuk Suara *Bass* (Randel, 1986)

2.2 Mel Frequency Cepstral Coefficient

Mel Frequency Cepstral Coefficient (MFCC) merupakan salah satu fitur untuk melakukan ekstraksi ciri suara khususnya adalah pada suara manusia. Tujuan utama dari proses MFCC ini adalah untuk menirukan perilaku dari pendengaran manusia. Proses kerja dari MFCC yang sering digunakan adalah *sampling, pre-emphasis, frame blocking, windowing, fast fourier transform, mel-frequency wrapping, logarithmic compression, cepstrum.*

Teknik MFCC ini sering digunakan untuk membuat *fingerprint* pada file suara. MFCC didasarkan pada variasi-variasi yang dikenal oleh batas frekuensi yang dapat dikenal oleh telinga manusia dengan jarak linear filter pada frekuensi rendah dan frekuensi tinggi di mana secara logaritmik digunakan untuk menangkap karakteristik penting dari suatu ucapan.

Karena dalam MFCC proses kerjanya menirukan kondisi dari pendengaran manusia, karakteristik ini dapat digambarkan dalam skala mel-frekuensi di mana terdiri atas frekuensi linear di bawah 1000Hz dan frekuensi logaritmik di atas 1000Hz. Keunggulan dari MFCC ini adalah dapat menangkap karakteristik suara yang sangat penting untuk pengenalan suara, menghasilkan suara seminimal mungkin tanpa menghilangkan informasi-informasi penting,

menirukan proses pendengaran manusia dalam melakukan ekstraksi ciri terhadap sinyal suara yang diberikan.

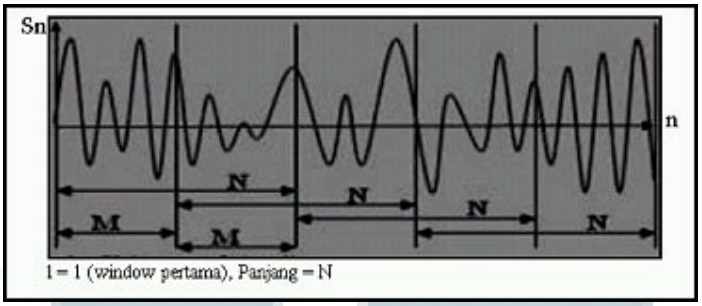
Proses-proses dari MFCC yang akan dilakukan dalam penelitian ini adalah sebagai berikut.

2.2.1 Sampling

Proses ini merupakan proses di mana data-data suara vokal dikumpulkan. Jenis suara yang dikumpulkan adalah suara sopran, tenor, alto, dan bass. Data suara dikumpulkan dengan cara melakukan perekaman terhadap seseorang yang menyanyikan suatu tangga nada di mana hasilnya akan disimpan dalam bentuk file audio berformat *.wav. Data dari hasil sampling ini yang kemudian dipakai untuk melakukan proses ekstraksi ciri suara menggunakan MFCC (Setiawan, 2011).

2.2.2 Frame Blocking

Proses *frame blocking* ini, sinyal ucapan suara yang telah diambil dibagi menjadi beberapa *frame* yang berisi N-Sampel di mana masing-masing dari *frame*nya akan dipisahkan oleh M, di mana M < N. *Frame* pertama berisi sampel N pertama. *Frame* kedua dimulai dari M-Sampel setelah frame pertama sehingga *frame-frame* tersebut overlap. Proses ini dilakukan seterusnya sampai seluruh sinyal suara dimuat dalam *frame*. Masing-masing dari *frame* N dipisah oleh *frame* M (overlap) (Setiawan, 2011).



Gambar 2.5 Contoh Pembagian *Frame Blocking* (M < N) (Setiawan, 2011)

Keterangan :

Sn = nilai sampel yang dihasilkan

n = urutan sampel yang akan diproses

2.2.3 Windowing

Pada langkah ini, dilakukan *windowing* pada setiap *frame* untuk meminimalisir terjadinya diskontinuitas sinyal pada permulaan dan akhir dari setiap *frame*. Konsep dari proses ini adalah membuat sinyal mendekati angka nol pada permulaan dan akhir dari *frame*. Bila seandainya *window* didefinisikan sebagai w(n), di mana 0 <= n <= N-1, dengan N adalah jumlah sampel dalam tiap *frame*, maka hasil dari proses ini adalah sinyal :

$$y(n) = x(n)w(n), \qquad 0 \leq n \leq N-1 \qquad \text{... Rumus 2.1}$$

Dengan

y(n) = sinyal hasil *windowing* sampel ke-n

x(n) = nilai sampel ke-n

w(n) = nilai window ke-n

N = jumlah sampel dalam *frame*

Juga menggunakan *hamming window* dengan bentuk sebagai berikut (Setiawan, 2011).

$$w(n) = 0{,}54 + 0{,}46 \cos\left(\frac{2\pi n}{N-1}\right), 0 \leq n \leq N-1 \quad \text{... Rumus 2.2}$$

2.2.4 Fast Fourier Transform

Proses *Fast Fourier Transform* (FFT) melakukan konversi tiap-tiap frame yang berisi N-Sampel dari format waktu ke format frekuensi. FFT merupakan suatu *fast-algorithm* untuk implementasi *Discrete Fourier Transform* yang dioperasikan pada sebuah sinyal waktu diskrit yang terdiri dari N-Sampel sebagai berikut (Setiawan, 2011).

$$f(n) = \sum_{K=0}^{N-1} y_k \, e^{-2\pi jkn/N}, n = 0{,}1{,}2, \ldots, N-1 \quad \text{... Rumus 2.3}$$

2.2.5 Mel-Frequency Wrapping

Terdapat studi psikofisik yang menunjukkan bahwa adanya persepsi manusia tentang frekuensi suara untuk sinyal ucapan yang tidak mengikuti dari skala linear. Sehingga untuk nada-nada yang memiliki frekuensi sebenarnya (f) dalam satuan Hz, sebuah pola akan diukur dalam sebuah skala '*mel*'. Skala '*mel-frequency*' ini adalah skala frekuensi linear yang berada di bawah 1000 Hz dan skala logaritmik yang berada di atas 1000 Hz. Skala *mel-frequency* ini didefinisikan oleh Stanley Smith, John Volkman, dan Edwin Newman sebagai berikut (Setiawan, 2011).

$$mel(f) = 2595 * log_{10}\left(1 + \frac{f}{700}\right) \quad \text{... Rumus 2.4}$$

2.2.6 Cepstrum

Cepstrum ini biasa digunakan untuk memperoleh informasi dari suatu sinyal suara yang diucapkan oleh manusia. Pada langkah ini, skala mel yang dihasilkan pada proses sebelumnya diubah menjadi *cepstrum* menggunakan *Discrete Cosine Transform* (DCT). Hasil dari proses di sini dinamakan MFCC.

Hasil proses MFCC merupakan keluaran dari alih ragam cosinus dari logaritma *short-term power spectrum* yang dinyatakan dalam skala *mel-frequency*. Jika mel *power spectrum coefficients* dinotasikan sebagai S_k, k = 1,2,….,K, Minh N.Do mendefinisikan koefisien dari MFCC (C_n) sebagai (Mustofa, 2007).

$$cn = \sum_{k=1}^{K} (\log Sk)\, cos\left[n\left(k-\frac{1}{2}\right)\frac{\pi}{k}\right], n = 1,2,\dots,K \qquad \text{... Rumus 2.5}$$

2.3 K-Nearest Neighbor

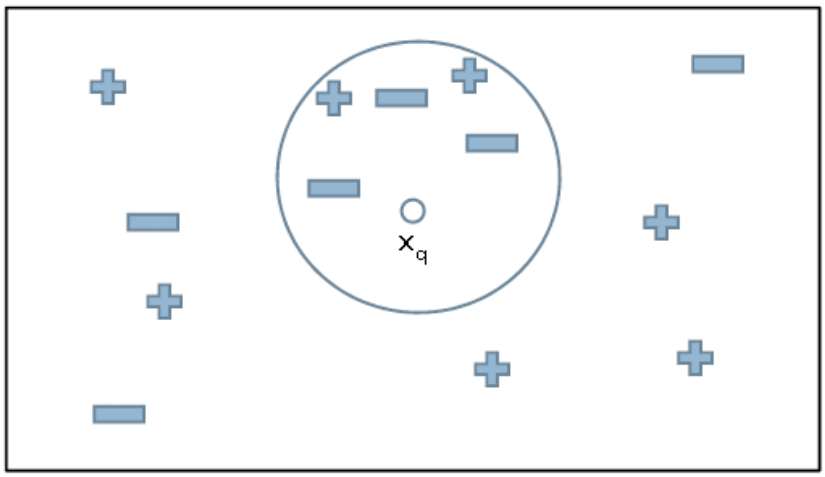
K-Nearest Neighbor merupakan suatu metode yang digunakan untuk melakukan klasifikasi atau pengelompokan suatu objek dengan melihat jarak terdekat antara data pembelajaran dengan objek tersebut. *K-Nearest Neighbor* juga sering kali disebut sebagai "*Lazy Algorithm*" di mana menunda suatu hasil untuk menggeneralisasi contoh pelatihan sampai adanya permintaan baru ditemukan. Konsepnya adalah kapanpun poin baru yang akan diklasifikasi ditemukan, maka nilai dari *K-Nearest Neighbor* (K-NN) ditemukan berdasarkan training data yang sudah ada.

Umumnya K-NN digunakan untuk melakukan kategorisasi pada suatu naskah dan beberapa digunakan untuk melakukan klasifikasi terhadap suatu gambar. K-NN berdasarkan pada konsep "*learning by analogy*" di mana data latih dideskripsikan dengan atribut numeric berupa *n*-dimensi. Pada setiap data latih akan mewakili sebuah titik yang ditandai dengan *c* dalam ruang dimensi *n*-dimensi. Jika terdapat sebuah data baru di mana labelnya belum diketahui diinputkan, maka K-NN akan mencari *k* buah data learning yang jaraknya merupakan jarak terdekat dengan data baru tersebut (Sukma, 2014).

Dalam pembelajaran mesin, algoritma K-NN ini juga dapat digunakan untuk *speech recognition, autonomous vehicle* (mesin mandiri), melakukan klasifikasi terhadap benda astronomi, bermain *backgammon*, meramalkan *heart attack*, meramalkan harga saham, melakukan *filter* terhadap spam, dan lainnya. Proses penghitungan jarak pada algoritma ini dapat menggunakan salah satu dari proses ini, yaitu : *Euclidean Distance, Minkowski Distance,* dan *Mahalanobis Distance* (Zhu, 2006).

Berikut ini merupakan *simple* K-NN *algorithm*:

- pada setiap contoh *training* [x, f(x)], tambahkan contohnya ke dalam suatu daftar, mis : contoh_latih,
- ketika sebuah data baru x_q akan diklasifikasi, $x_1,x_2,\ldots,x_k$ menandakan suatu instansi *k* dari contoh_latih yang paling dekat dengan x_q, setelah itu kembalikan *class* yang merepresentasikan data maksimal pada instansi *k* (Siddharth, 2009).



If K = 5, then in this case query instance xq will be classified as negative since three of its nearest neighbors are classified as negative.

Gambar 2.6 Contoh dari Algoritma K-NN (Deokar, 2009)

Perhitungan jarak dapat dihitung menggunakan beberapa fungsi jarak $d(x,y)$, di mana x dan y merupakan skenario dari susunan fitur N, $x = \{x_1,\ldots,x_N\}$, $y = \{y_1,\ldots, y_N\}$. dua fungsi jarak tersebut adalah sebagai berikut (Lammertsma, 2004).

- *Absolute distance measuring*

- *Euclidean distance measuring*

K-NN memiliki beberapa kelebihan yakni, dapat melakukan *training* data yang memiliki banyak *noise* dan efektif ketika *training* data-nya memiliki nilai yang besar. Sedangkan kelemahannya adalah K-NN harus menentukan nilai

dari parameter *k* (jumlah tetangga terdekat), melakukan pelatihan berdasarkan jarak yang belum diketahui dan jenis jarak apa dan atribut mana yang digunakan untuk mendapatkan hasil yang terbaik, dan ketika melakukan perhitungan memerlukan besar memori yang cukup besar karena diperlukan perhitungan jarak dari tiap instansi input pada keseluruhan contoh sampel (Sukma, 2014).