

PERANCANGAN SISTEM PERBAIKAN NADA SUARA MANUSIA DENGAN MENGGUNAKAN METODE *PHASE VOCODER* TERHADAP NADA REFERENSI MUSIK

Rudi Prasetyo^{*)}, Achmad Hidayatno, and Imam Santoso

Jurusan Teknik Elektro, Universitas Diponegoro Semarang
Jl. Prof. Sudharto, SH, kampus UNDIP Tembalang, Semarang 50275, Indonesia

^{*)} Email : rudiprasetyo92@gmail.com

Abstrak

Audio merupakan sinyal suara yang dapat dideteksi manusia dengan frekuensi terendah 20 Hz dan tertinggi 20 kHz. Suara beraturan dengan frekuensi tunggal tertentu disebut sebagai nada dan sering dikomposisikan dengan beberapa nada lain sehingga terdengar apik. Seorang penyanyi dalam melakukan pengambilan suara terkadang mengalami ketidaktepatan nada. Berdasarkan hal tersebut, maka dirancang sebuah sistem perbaikan nada suara manusia terhadap nada referensi musik dengan menggunakan metode *phase vocoder*. Perancangan ini terbagi menjadi 2 tahapan utama yaitu *pitch detection* dan *pitch correction*. Metode yang digunakan pada deteksi nada adalah FFT dengan ukuran variasi frame sebesar 256, 512, dan 1024 buah sampel serta jarak *overlapping* antara frame sebesar 25%, 50%, dan 75%. Hasil frekuensi suara yang diperoleh kemudian dibandingkan dengan frekuensi referensi musik. Apabila terdapat perbedaan maka dilakukan proses *pitch correction* dengan menggunakan metode *phase vocoder* dengan variasi *overlapping* sebesar 25%, 50%, dan 75%. Hasil yang diperoleh dari sistem menunjukkan bahwa parameter terbaik dalam mendeteksi nada menggunakan ukuran frame 512 dan 1024 buah sampel, serta nilai *overlapping* sebesar 50% dan 75%. Sedangkan parameter *overlapping* terbaik yang dapat digunakan untuk mengkoreksi nada sebesar 50% dan 75%. Sistem ini kemudian diujikan dengan menggunakan data suara dan memperoleh tingkat keberhasilan sebesar 96,2538%.

Kata kunci: Pitch Detection, Pitch Correction

Abstract

Audio is a sound that can be detected by human ears with the lowest frequency is 20 Hz and the highest frequency is 20 kHz. It is produced by the vibration of the object. Sound or the uniform sound with the only one special frequency is called by a tone and always be combined by the others to heard more beautiful. As we known, a singer sometimes wrong to takes a pitch from a tone. Therefore, this research designed a tone rectifying system of human voice to the music reference using *phase vocoder* method. This design will be divided into 2 steps, *pitch detection* and *pitch correction*. Fast Fourier Transform (FFT) is used in *pitch detection* process with 256, 512, and 1024 variation of size frame and 25%, 50%, 75% variation of *overlapping* between two frames. The frequency result from *pitch detection* then compared by the frequency of music reference. If there is a difference between it so the process of *pitch correction* will be done to the system using *phase vocoder* method with 25%, 50%, and 75% variation of *overlapping*. Analytical results from the system show that the best parameters can be used in *pitch detection* is 512 or 1024 frame size and 50% or 75% *overlapping*. While the best parameters can be used in *pitch correction* is 50% or 75% *overlapping*. This system then tested by voice data and get 96,2538% success rate.

Keywords: Pitch Detection, Pitch Correction

1. Pendahuluan

Proses pengolahan sinyal begitu berkembang sejalan dengan perkembangan dunia digital, salah satunya adalah pengolahan sinyal suara berupa audio. Audio merupakan sinyal suara yang dapat dideteksi oleh telinga manusia dengan frekuensi terendah 20 Hz dan frekuensi tertinggi

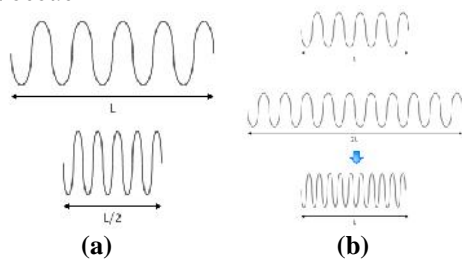
20 kHz yang dihasilkan oleh getaran suatu benda. Bunyi atau suara beraturan dengan frekuensi tunggal tertentu disebut sebagai nada dan sering dikomposisikan dengan beberapa nada lain pada frekuensi tertentu sehingga terdengar apik. Oleh karenanya, proses pengolahan sinyal dapat diterapkan untuk membantu dalam tercapainya nada pada frekuensi tertentu. Seperti yang diketahui bahwa

seorang penyanyi dalam melakukan pengambilan suara terkadang mengalami ketidaktepatan nada atau sumbang. Oleh karena itu, dibutuhkan sebuah metode untuk mengubah nada pada suara manusia agar sesuai terhadap nada referensi atau acuan yang berupa nada tunggal dari audio musik.

Berdasarkan hal tersebut maka pada penelitian ini dilakukan sebuah perancangan mengenai pengolahan sinyal digital pada suara manusia terhadap ketepatannya pada nada referensi dengan menggunakan metode *phase vocoder*. *Phase vocoder* merupakan sebuah metode yang dapat diterapkan untuk menggeser frekuensi sinyal suara penyanyi sehingga didapatkan sinyal suara yang sesuai dengan frekuensi yang telah dijadikan sebagai referensi.

2. Metode

2.1 Pitch Shifting menggunakan metode Phase Vocoder



Gambar 1 Pitch Shifting (a) dengan merubah durasi sinyal (b) tanpa merubah durasi sinyal^[12]

Pitch shifting (menggeser nilai frekuensi) secara sederhana dapat dilakukan dengan mempercepat atau memperlambat waktu putar sinyal suara seperti yang ditunjukkan pada gambar 1(a). Pada gambar tersebut sinyal suara diputar dua kali lebih cepat dari sinyal aslinya sehingga nilai frekuensi yang dimiliki menjadi lebih besar dan menghasilkan *pitch* (nada) yang bernilai satu oktaf lebih tinggi, namun dengan durasi waktu sinyal dua kali lebih pendek.

Cara lain yang lebih baik untuk dapat melakukan *pitch shifting* adalah dengan menggunakan metode *phase vocoder*. Metode ini mampu merubah panjang sinyal suara menjadi lebih panjang atau lebih pendek dari sinyal aslinya tanpa merubah *pitch* (nada). Sinyal yang panjangnya telah berubah kemudian diputar lebih cepat atau lebih lambat sehingga frekuensi dan *pitch* (nada) juga berubah dengan durasi waktu sinyal yang sama seperti sinyal aslinya. Seperti yang ditunjukkan pada gambar 1(b), panjang sinyal dibuat dua kali lebih panjang tanpa adanya perubahan *pitch* (nada) dengan metode *phase vocoder*. Sinyal tersebut kemudian diputar dua kali lebih cepat sehingga nilai frekuensi yang dimiliki menjadi lebih besar dan menghasilkan *pitch* (nada) yang bernilai satu oktaf lebih tinggi tanpa adanya perubahan durasi waktu sinyal.

2.2 Hubungan antara Nada dengan Frekuensi

Nada adalah bunyi yang beraturan dengan frekuensi tunggal tertentu dan memiliki tinggi nada tertentu menurut frekuensinya. Sifat nada ada 4 (empat), yaitu :

- 1) *Pitch* yaitu ketepatan jangkauan nada.
- 2) Durasi yaitu lamanya sebuah nada harus dibunyikan.
- 3) Intensitas nada yaitu keras lembutnya nada yang harus dibunyikan.
- 4) *Timbre* yaitu warna suara yang berbeda pada setiap orang.

Kumpulan dari semua nada dalam musik disebut sebagai Tangga Nada Kromatik. Nada selalu berulang untuk tiap oktaf yang ada, sehingga istilah tangga nada kromatik sering dipakai untuk ke-12 nada dari tiap oktaf.

Perbedaan antara dua buah *pitch* (nada) yang berdekatan disebut sebagai *semitone*. Meskipun ada 12 nada dalam satu oktaf, tapi hanya 7 huruf pertama dari abjad yang dipakai untuk memberi nama pada nada, yaitu dari A sampai G. Kelima nada yang lain dalam tangga nada kromatik diberi nama dengan menempatkan tanda kres (#) atau tanda mol (b) setelah notasi nada.

Tabel 1 Tangga Nada kromatik dalam Musik^[11]

	Oktaf (Hz)					
	1	2	3	4	5	6
C	32,703	65,406	130,81	261,63	523,25	1046,5
C#	34,468	69,296	138,59	277,18	554,37	1108,7
D	36,708	73,416	146,83	293,66	587,33	1174,7
D#	38,891	77,782	155,56	311,13	622,25	1244,5
E	41,204	82,407	164,81	329,63	659,26	1318,5
F	43,654	87,307	174,61	349,23	698,46	1396,9
F#	46,249	92,499	185,00	369,99	739,99	1480,0
G	48,999	97,999	196,00	392,00	783,99	1568,0
G#	51,913	103,83	207,65	415,30	830,61	1661,2
A	55,000	110,00	220,00	440,00	880,00	1760,0
A#	58,270	116,54	233,08	466,16	932,33	1846,7
B	61,735	123,47	246,94	493,88	987,77	1975,5

Hubungan antara *semitone* dan frekuensi dasar adalah sebagai berikut.

$$p = 69 + 12 \log_2 \left(\frac{f}{440} \right) \quad (1)$$

Dengan p merupakan indeks *semitone*, dan f merupakan frekuensi dasar (dalam Hertz). Sementara itu, proses *pitch shifting* dari frekuensi asal menggunakan metode "Twelve-Note Chromatic Scale", sehingga frekuensi hasil didapatkan dari perkalian suatu koefisien berdasarkan metode tersebut dengan frekuensi asalnya. Dengan s adalah jarak antara nada suara dan nada acuan.

$$p_{final} = p_{initial} + s \quad (2)$$

$$f_{final} = 2^{\frac{s}{12}} \times f_{initial} \quad (3)$$

2.3 Deteksi Nada dengan FFT

Transformasi Fourier merupakan bagian penting dalam pengolahan sinyal, yakni mampu merubah sinyal dari

domain waktu menjadi domain frekuensi. Hasil dari transformasi fourier biasanya terdiri dari bilangan kompleks yang merepresentasikan informasi amplitudo serta informasi fasa. Frekuensi didapat dengan mencari letak amplitudo tertinggi dari sebuah *bin* (koefisien) hasil Transformasi Fourier. Audio yang telah direkam dengan bantuan komputer akan menghasilkan amplitudo dalam waktu *discrete*, oleh karena itu *Discrete Fourier Transform* (DFT) diimplimentasikan untuk menganalisis audio yang telah menjadi digital. Secara matematis, transformasi fourier dapat dirumuskan sebagai berikut.

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j\left(\frac{2\pi}{N}\right)nk} \quad (4)$$

Dengan :
 n = indeks dalam domain waktu = 0,1,..., N-1; k = indeks dalam domain frekuensi = 0,1, .. , N-1

Algoritma FFT merupakan bagian dari DFT yang memiliki waktu komputasi lebih singkat, sehingga proses perhitungan dengan menggunakan algoritma FFT akan menjadi lebih cepat.

2.4 Koreksi Nada dengan Phase Vocoder

Phase vocoder merupakan salah satu algoritma pengolahan sinyal digital yang terdiri dari tahap *analysis*, *processing*, dan *synthesis*. Secara matematis, teknik ini merupakan suatu algoritma yang dapat menghasilkan keluaran sinyal yang identik dengan sinyal input atau sinyal keluaran yang berupa versi modifikasi dari sinyal masukannya.

a. Analysis

Pada tahap ini, sinyal dibagi ke dalam bagian yang lebih kecil untuk memudahkan proses *analysis*. Pada sinyal tersebut dilakukan proses *windowing* dengan menggunakan *window* hanning. *Window* hanning dipilih karena memiliki energi sinyal yang lebih terkonsentrasi pada main lobenya.

Setelah itu dilakukan transformasi fourier pada setiap *frame*. Untuk menambah resolusi spektrumnya, digunakan overlap 75%. Jumlah sampel dari dua buah *window* yang berurutan disebut ukuran *hop* (*hop a*), yang setara dengan $N/4$ untuk *overlap* 75%. Tahapan ini dapat dinyatakan dalam persamaan berikut :

$$(X_a[K])_i = \sum_{n=0}^{N-1} x[n + i \times (\text{hop}_a)] w[n] e^{-j\frac{2\pi kn}{N}} \quad (5)$$

Dengan :
 $(X_a[K])_i$ = spektrum diskrit dari *frame* ke- i ; $w[n]$ = *window* hanning; $x[n]$ = sampel dari sinyal; hop_a = ukuran *hop*

b. Processing

Proses transformasi dengan menggunakan transformasi fourier (FFT) N titik menghasilkan sejumlah N frekuensi

bin, mulai dari 0 sampai dengan $(N-1)/N \times fs$ dengan interval (fs/N) Sinyal dengan frekuensi diantara 2 *bin* akan mengalami gangguan dan energinya akan tersebar ke *bin*-*bin* terdekat lainnya. Informasi fasa digunakan untuk memperbaiki akurasi dari estimasi frekuensi pada setiap *bin*. Perbedaan fasa antara 2 fasa kemudian disebut dengan *phase shift* ($\phi_a[k]$) dimana k merupakan indeks *bin*, dan i merupakan indeks *frame*.

Dari hasil transformasi fourier sebelumnya, informasi fasanya masih merupakan *wrapped phase*, sehingga ($\phi_a[k]$) masih ada pada rentang $-\pi$ sampai π . Untuk mendapatkan nilai frekuensi sebenarnya, maka perhitungannya akan lebih rumit dibandingkan dengan informasi fasa yang bukan merupakan *unwrapped phase*. Deviasi frekuensi dari *bin* harus dihitung terlebih dahulu dengan persamaan berikut

$$(\Delta\omega[K])_i = \frac{(\phi_a[k])_i - (\phi_a[k])_{i-1}}{\Delta t_a} - \omega_{bin}[k] \quad (6)$$

Nilai deviasi frekuensi tersebut kemudian dikonversikan ke dalam bentuk *wrapped*.

$$(\Delta_{wrapped}[K])_i = \text{mod}[(\Delta\omega[k])_i + \pi), 2\pi] - \pi \quad (7)$$

Nilai ini kemudian dijumlahkan dengan frekuensi *bin* untuk mendapatkan frekuensi sebenarnya pada komponen dalam *frame*.

$$(\omega_{true}[K])_i = \omega_{bin}[k] + (\Delta_{wrapped}[k])_i \quad (8)$$

Dengan :
 $(\phi_a[k])_i$ = fasa dari *frame* ke- i ; $(\phi_a[k])_{i-1}$ = fasa dari *frame* sebelumnya; $\omega_{bin}[k]$ = frekuensi *bin*; $(\Delta\omega[k])_i$ = deviasi frekuensi (bukan dalam bentuk *wrapped*)

Nilai fasa baru dari setiap *bin* kemudian dihitung dengan menjumlahkan *phase shift* yang diinginkan untuk menghindari diskontinuitas. Hal ini dilakukan dengan mengalikan frekuensi sebenarnya dengan interval waktu dari tahap *synthesis* yang dilakukan secara rekursif.

$$\phi_s[k]_i = (\phi_s[k])_{i-1} + \Delta t_s \times (\omega_{true}[k]_i) \quad (9)$$

Kemudian, akan dihasilkan spektrum baru dengan amplitudo dan fasa seperti ditunjukkan pada persamaan berikut:

$$|(x_s[k]_i)| = |(x_a[k]_i)| \quad \angle(x_s[k]_i) = \phi_s[k]_i \quad (10)$$

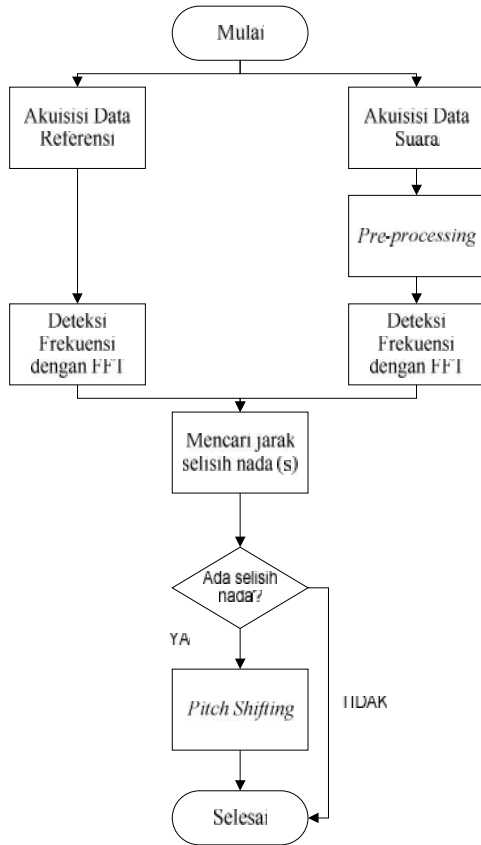
c. Synthesis

Untuk setiap *frame* dilakukan *invers* transformasi fourier untuk mengembalikan representasi sinyal ke dalam domain waktu. *Invers* transformasi fourier ini dilakukan pada spektrum di setiap *frame*. Pada hasilnya kemudian dilakukan proses *windowing* dengan *window* hanning untuk memperhalus sinyal.

$$y[n] = \left(\frac{1}{N}\right) \sum_{k=0}^{N-1} (X_s[k]_i) e^{j2\pi kn/N} w(n) \quad (11)$$

2.5 Perancangan Sistem

Pada penelitian kali ini, perancangan dilakukan secara bertahap. Studi literatur dilakukan pertama kali agar perancangan yang dilakukan dapat dikerjakan secara teratur dan terarah. Gambar diagram alir pengerjaan sistem dapat dilihat pada gambar 3.



Gambar 2 Diagram alir pengerjaan sistem

Dalam diagram alir pengerjaan sistem terdapat 5 langkah utama yang harus dilakukan yaitu akuisisi data baik itu data suara maupun data referensi, *pre-processing* terhadap data suara yang telah diambil, deteksi frekuensi pada masing-masing data, setelah deteksi frekuensi dilakukan maka mencari jarak selisih nada (s) antara nada suara dan data referensi, kemudian *pitch shifting* dapat dilakukan pada nada suara yang belum sesuai dengan nada referensi.

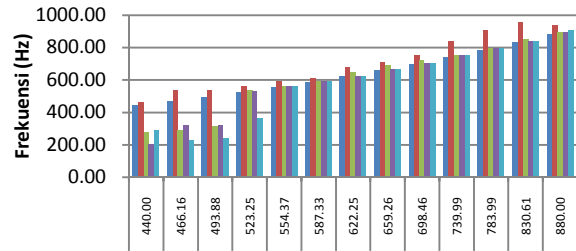
3. Hasil dan Analisa

3.1 Pengaruh Tingkat *Downsampling* pada proses *Harmonic Product Spectrum*

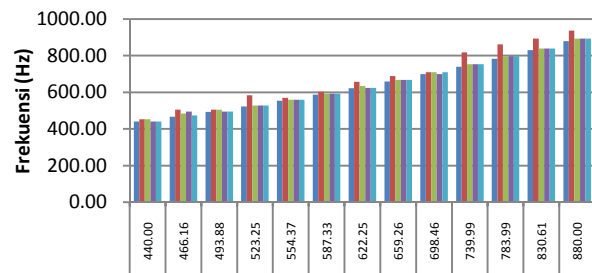
Pada pendeteksian frekuensi dengan menggunakan metode transformasi fourier terdapat sebuah proses yang disebut sebagai *harmonic product spectrum*. Proses ini bertujuan untuk mendapatkan amplitudo yang paling dominan dengan mengurangi jumlah sampel

(*downsampling*) kemudian mengalikannya dengan sinyal asli hasil FFT.

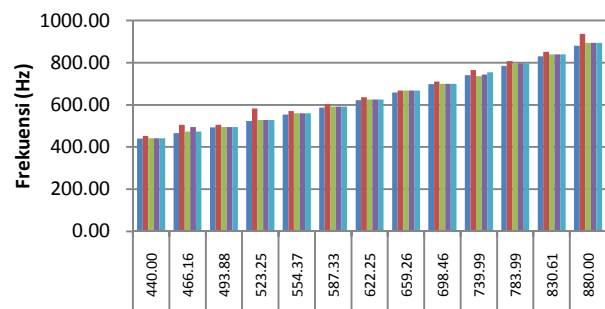
Tingkat *downsampling* terbaik pada perancangan ini dicari melalui proses perhitungan dengan menggunakan *relative percentage error* dan *mean relative percentage error* pada masing-masing *frame* kemudian mengakumulasi secara keseluruhan. Berikut hasil akumulasi perhitungan dari keseluruhan *frame*.



Gambar 3 Grafik perbandingan hasil deteksi frekuensi terhadap masing-masing nada dengan variasi *downsampling* pada ukuran *frame* 256



Gambar 4 Grafik perbandingan hasil deteksi frekuensi terhadap masing-masing nada dengan variasi *downsampling* pada ukuran *frame* 512



Gambar 5 Grafik perbandingan hasil deteksi frekuensi terhadap masing-masing nada dengan variasi *downsampling* pada ukuran *frame* 512

Keterangan mengenai warna gambar 3 – 5 ditunjukkan berikut ini.

■ referensi ■ 1x *downsampling* ■ tanpa *downsampling* ■ 2x *downsampling* ■ 3x *downsampling*

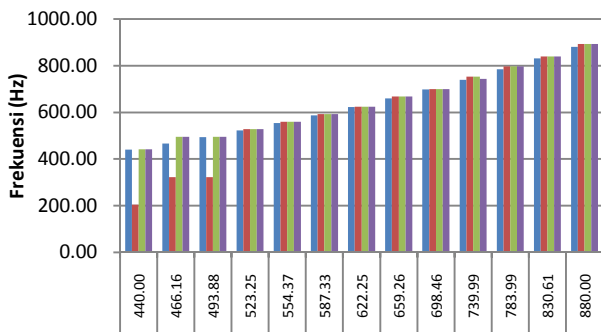
Tabel 2 Akumulasi *mean relative percentage error* dari seluruh *frame*

Ukuran <i>Frame</i>	<i>Mean relative percentage error (%)</i>			
	Tanpa <i>downsampling</i>	1x <i>downsampling</i>	2x <i>downsampling</i>	3x <i>downsampling</i>
256	9.488783841	10.34874064	9.953049445	13.73348026
512	5.933269365	1.769874067	1.340273025	1.1032253
1024	3.932113431	0.887501092	1.228061031	0.984633382
MRPE	6.451388879	4.335371934	4.173794501	5.273779648

Keterangan :MRPE (*Mean relative percentage error*)

Dari hasil akumulasi keseluruhan *frame* maka didapat tingkat *downsampling* paling baik dengan nilai *error* sebesar 4.173794501 % pada saat sistem menggunakan 2 tingkat *downsampling*.

3.2 Pengaruh Ukuran *Frame* terhadap Deteksi Frekuensi



Keterangan :

■ referensi ■ ukuran *frame* 256 ■ ukuran *frame* 512 ■ ukuran *frame* 1024

Gambar 6 Grafik perbandingan hasil deteksi frekuensi terhadap masing-masing nada dengan variasi *frame* tanpa ada *overlapping*

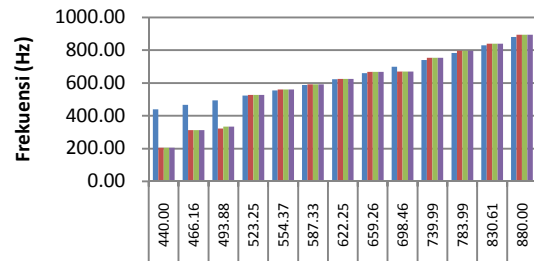
Berdasarkan dari pengujian dan analisis sub bab sebelumnya tingkat *downsampling* yang tepat dengan kesalahan minimum didapat ketika sistem memiliki 2 tingkat *downsampling*. Oleh karena itu, pada pengujian dan analisis ukuran *frame* digunakan 2 tingkat *downsampling* tanpa ada *overlapping* untuk mendapatkan ukuran *frame* terbaik. Berdasarkan hasil perhitungan dengan melihat data pada gambar 6 didapat parameter terbaik dengan ukuran *frame* 1024 buah sampel dengan *error* sebesar 1.228061 %. Berikut hasil perhitungan keseluruhan.

Tabel 3 Data hasil perhitungan *relative percentage error* pada masing-masing *frame*

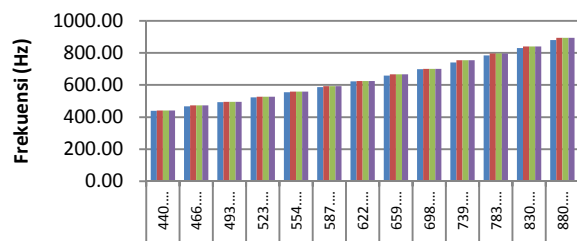
Nada	Frekuensi (Hz)	<i>relative percentage error (%)</i>		
		<i>Frame</i> 256	<i>Frame</i> 512	<i>Frame</i> 1024
A4	440.00	53.50795	0.325227	0.325227
A4#	466.16	30.71091	6.238201	6.24335
B4	493.88	34.5999	0.28023	0.28023
C5	523.25	0.824271	0.824271	0.824271
C5#	554.37	0.990854	0.990854	0.990854
D5	587.33	0.822876	0.822876	0.822876
D5#	622.25	0.355645	0.355645	0.355645
E5	659.26	1.254285	1.254285	1.254285
F5	698.46	0.196003	0.205022	0.196003
F5#	739.99	1.847593	1.847593	0.392708
G	783.99	1.624893	1.624893	1.624893
G5#	830.61	1.105814	1.105814	1.105814
A5	880.00	1.548636	1.548636	1.548636
MRPE		9.953049	1.340273	1.228061

Keterangan :MRPE (*Mean relative percentage error*)

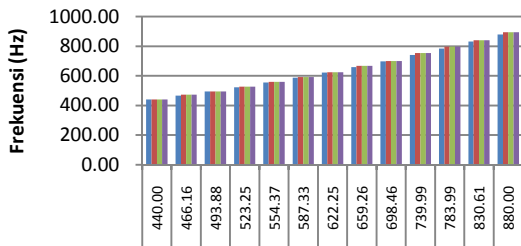
3.3 Pengaruh Besar *Overlapping* terhadap Deteksi Frekuensi



Gambar7 Grafik perbandingan hasil deteksi frekuensi terhadap masing-masing nada dengan variasi *overlapping* pada ukuran *frame* 256



Gambar 8 Grafik perbandingan hasil deteksi frekuensi terhadap masing-masing nada dengan variasi *overlapping* pada ukuran *frame* 512



Gambar 9 Grafik perbandingan hasil deteksi frekuensi terhadap masing-masing nada dengan variasi overlapping pada ukuran frame 1024

Keterangan mengenai warna pada gambar 7 – 9 ditunjukkan berikut ini.
 ■ Asli ■ Overlapping 25% ■ Overlapping 50% ■ Overlapping 75%

Tabel 4 Akumulasi mean relative percentage error dari seluruh frame

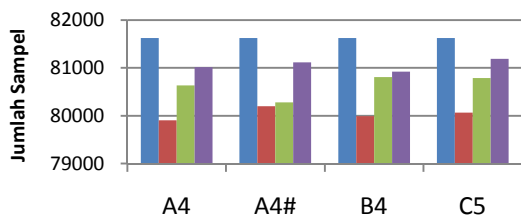
Frame	Mean relative percentage error (%)		
	25%	50%	75%
256	9.68047	9.530182	9.530182
512	0.984633	0.984633	0.984633
1024	0.984633	0.984633	0.984633
MRPE	3.883245	3.833149	3.833149

Keterangan :MRPE (Mean relative percentage error)

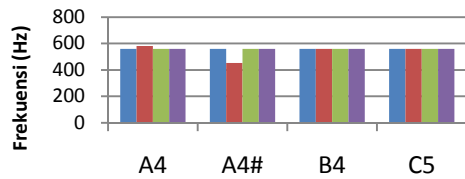
Dari hasil akumulasi keseluruhan frame pada table 4 dapat terlihat bahwa semakin besar overlapping akan semakin bagus hasil yang diperoleh. Apabila diamati pada masing-masing frame, nilai kesalahan paling kecil sama-sama ditunjukkan pada ukuran frame 512 dan 1024 dengan besar overlapping yang sama yaitu 25%, 50%, 75%.

3.4 Pengaruh Besar Overlapping terhadap Koreksi Nada

Berdasarkan hasil analisis pengaruh overlapping terhadap koreksi frekuensi hasil terbaik didapat ketika besar overlapping senilai 50% dan 75%. Hal ini diamati berdasarkan frekuensi dan jumlah sampel yang dihasilkan dari proses koreksi nada (pitch correctin). Dalam mengamati perubahan frekuensi menjadi lebih tinggi digunakan data musik dengan nada A₄, A₄#, B₄, dan C₅ dengan nada referensi C₅#. Hasilnya sebagai berikut.

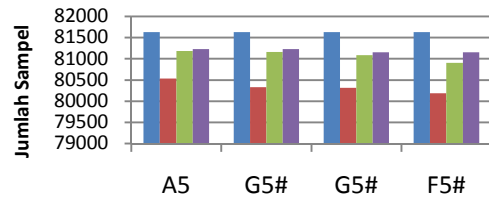


Gambar 10 Grafik perbandingan jumlah sampel pada nada A₄, A₄#, B₄, C₅ setelah mengalami proses pitch correction dengan nada referensi C₅#

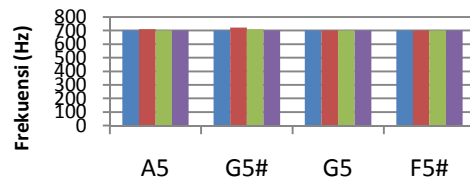


Gambar 11 Grafik perbandingan nilai frekuensi pada nada A₄, A₄#, B₄, C₅ setelah mengalami proses pitch correction dengan nada referensi C₅#

Untuk dapat mengamati pengaruh perubahan frekuensi menjadi lebih rendah atau penurunan nada digunakan data musik A₅, G₅#, G₅, F₅# dengan nada referensi F₅. Berikut hasil yang diperoleh.



Gambar 12 Grafik perbandingan jumlah sampel pada nada A₅, G₅#, G₅, F₅# setelah mengalami proses pitch correction dengan nada referensi F₅



Gambar 13 Grafik perbandingan nilai frekuensi pada nada A₅, G₅#, G₅, F₅# setelah mengalami proses pitch correction dengan nada referensi F₅

Keterangan warna mengenai warna pada gambar 10 – 13 ditunjukkan berikut ini.

■ Asli ■ Overlapping 25% ■ Overlapping 50% ■ Overlapping 75%

3.5 Pengujian dengan Data Suara

Tabel 5 Hasil Pengujian Data Suara pada sampel 1

Nada	Frekuensi Referensi (Hz)	Deteksi Nada (Pitch Detection)		Koreksi Nada	Koreksi Nada (Pitch Correction)	
		Frekuensi (Hz)	Keterangan		Frekuensi (Hz)	Keterangan
A4	441.431	441.431	T	TIDAK	-	-
A4#	473.73	473.73	T	TIDAK	-	-
B4	495.264	506.03	S	YA	495.264	T
C5	527.563	538.33	S	YA	527.563	T
C5#	559.863	570.63	S	YA	559.863	T
D5	582.163	582.163	T	TIDAK	-	-
D5#	624.463	635.229	S	YA	624.463	T
E5	667.529	678.296	S	YA	678.296	S
F5	699.829	710.596	S	YA	699.829	T
F5#	753.662	699.829	S	YA	753.662	T
G	796.729	796.729	T	TIDAK	-	-
G5#	839.795	861.328	S	YA	839.795	T
A5	883.628	904.395	S	YA	883.628	T

Tabel 6 Hasil Pengujian Data Suara pada sampel 2

Nada	Frekuensi Referensi (Hz)	Deteksi Nada (<i>Pitch Detection</i>)		Koreksi Nada	Koreksi Nada (<i>Pitch Correction</i>)	
		Frekuensi (Hz)	Keterangan		Frekuensi (Hz)	Keterangan
A4	441.431	441.431	T	TIDAK	-	-
A4#	473.73	484.497	S	YA	473.73	T
B4	495.264	506.03	S	YA	495.264	T
C5	527.563	538.33	S	YA	527.563	T
C5#	559.863	559.863	T	TIDAK	-	-
D5	592.163	592.163	T	TIDAK	-	-
D5#	624.463	635.229	S	YA	624.463	T
E5	667.529	667.529	T	TIDAK	-	-
F5	699.829	699.829	T	TIDAK	-	-
F5#	753.662	699.829	S	YA	753.662	T
G	796.729	775.195	S	YA	796.729	T
G5#	839.795	839.795	T	TIDAK	-	-
A5	893.628	893.628	T	TIDAK	-	-

KETERANGAN T = TEPAT ;S = SUMBANG

Masing-masing sampel kemudian dihitung keberhasilan rata-rata dan diperoleh hasil 96,2538 %.

4. Kesimpulan

Dari hasil pengujian dan analisis yang telah dilakukan pada bab sebelumnya maka dapat diambil kesimpulan bahwa algoritma FFT mampu dikatakan baik sebagai metode pendeteksi frekuensi dengan tingkat kesalahan 0.984633% pada saat menggunakan ukuran *frame* 512 atau 1024 buah sampel dan jarak antara *frame* sebesar 50% atau 75%. Selain itu sistem dikatakan baik dalam melakukan koreksi nada menggunakan metode *phase vocoder* dengan tingkat keberhasilan sebesar 96,2538 % yakni semakin besar *overlapping* maka hasil yang diperoleh juga semakin baik.

Adapun saran yang dapat diberikan sehubungan dengan perancangan ini adalah perlunya diperhatikan kualitas *hardware*, *software*, jarak antara mulut dengan *microphone*, serta kondisi lingkungan saat akuisisi data sehingga data yang didapat memiliki kualitas yang baik.

Referensi

- [1]. Gunawan, Dadang dan Filbert Hilman Juwono. *Pengolahan Sinyal Digital Dengan Pemrograman MATLAB*. Jakarta: Graha Ilmu, 2012.
- [2]. Sugiarto, Aris. *Pemrograman GUI dengan Matlab*. Yogyakarta : Andi, 2006.
- [3]. Zolzer, Udo. *DAFX – Digital Audio Effects*. England: John Wiley & Sons, Ltd, 2007.
- [4]. Peimani, Michael. *Pitch Correction for the Human Voice*, Thesis of California University Santa Cruz, 2009.
- [5]. Utomo, Aryo Baskoro. *Analisa Karakteristik Suara Manusia Berdasarkan Frekuensi Fundamental Dan Tingkat Usia*. Makalah Penelitian Teknik Elektro Universitas Diponegoro Semarang.
- [6]. Eva, Kartika Kurniawati Susantyo. *Simulasi dan Analisis Perbaikan Ketepatan Nada Suara Manusia Menggunakan Phase Vocoder*. Makalah Penelitian Fakultas Elektro dan Komunikasi Institut Teknologi Telkom, 2013.
- [7]. Gotzen, Amalia De, Nicola Bernardini, Daniel Arfib. *Traditional (?) Implementations of Phase Vocoder : The Tricks of The Trade*. Italia, 2000.
- [8]. Nielsen, Andreas Brinch. *Pitch Based Sound Classification*. Thesis of Technical Denmark University, 2005.
- [9]. Pedersen, Magnus Eric Hvass. *The Phase Vocoder and its Realization*. University of Aarhus, 2003.
- [10]. Nurhayati, Okky Dwi. *Pengolahan Audio*. http://eprints.undip.ac.id/20105/1/Multi_pert2.pdf (diakses tanggal 24 Mei 2014)
- [11]. Blood, Brian. *Music Theory Online*. <http://www.dolmetsch.com/musictheory27.htm> (diakses tanggal 25 Mei 2014).
- [12]. Grondi, Francois. *Guitar Pitch Shifter*. <http://www.guitarpitchshifter.com/> (diakses tanggal 25 Mei 2014)
- [13]. Fletwood, Dan. Rhea. https://www.projectrhea.org/rhea/index.php/Student_summary_speech (diakses tanggal 31 Juli 2014)