

METODE KOMPRESI AUDIO DAN VIDEO UNTUK PENGIRIMAN DATA TELEKONFERENSI

Arief Kurniawan

Jurusan Teknik Elektro FTI – ITS

Email : arifku@ee.its.ac.id

ABSTRAK

Program-program yang digunakan telekonferensi sudah banyak tersedia baik yang komersial atau yang gratis. Tetapi kebanyakan program tersebut dapat berjalan dengan baik jika mempunyai bandwidth yang lebar. Karena lebar bandwidth yang mencukupi untuk melakukan telekonferensi sangat mahal, maka diperlukan suatu rekayasa agar pengiriman data telekonferensi (data audio dan video) menjadi lebih kecil, agar telekonferensi dapat dilakukan walaupun dengan bandwidth terbatas. Kompresi data Audio dapat dilakukan dengan menggunakan metode kompresi Transformasi Fourier, sedangkan untuk data Video setelah dipecah-pecah dikompresi dengan menggunakan kompresi Super Resolusi Iterative Back Projection (IBP) yang dimulai dengan sebuah initial guess sebagai gambar resolusi yang lebih tinggi dengan melakukan iterasi.

Kata Kunci : *Transformasi Fourier, Super Resolusi Iterative back Projection.*

1. LATAR BELAKANG

Dalam menyelesaikan suatu masalah, seringkali manusia membutuhkan komunikasi untuk menyampaikan ide dan pendapatnya. Bentuk komunikasi paling sederhana adalah dengan berbicara langsung, memberikan informasi langsung pada lawan bicara, kemudian komunikasi berkembang dari penyampain langsung menjadi jarak jauh. Komunikasi dapat terjalin juga tidak hanya dua pihak tapi bisa lebih dari dua pihak sehingga seluruhnya bisa mendapatkan atau menyampaikan informasi masing-masing

Untuk komunikasi jarak jauh oleh lebih dari satu pihak sekarang ini dapat dilakukan melalui telekonferensi sebagai pengembangan dari teknologi telepon. Telekonferensi sendiri hanya pada awalnya hanya menyampaikan informasi suara saja seperti halnya telepon. Dan akhirnya telekonferensi semakin menarik dengan tambahan penyampaian informasi visual sehingga tampak seolah bertemu langsung dengan lawan bicara. Telekonferensi audio visual lebih banyak diterapkan pada sistem berbasis komputer dan melauai jaringan komputer.

Program-program yang digunakan telekonferensi sudah banyak tersedia baik yang komersial atau yang gratis. Tetapi kebanyakan program tersebut dapat berjalan dengan baik jika mempunyai bandwidth yang lebar. Jika dipaksakan dengan bandwidth yang kecil maka program telekonferensi akan berjalan dengan putus-putus atau tidak lancar.

2. TRANSFORMASI FOURIER

Transformasi *Fourier* dapat digunakan untuk mengubah bunyi dalam domain waktu kedalam domain frekwensi yang disebut deret *Fourier*. Transformasi *Fourier* untuk fungsi berkesinambungan :

$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt$$

dengan : $e^{-j\theta} = \cos(\theta) - j \sin(\theta)$

Invers *Fourier* untuk fungsi berkesinambungan:

$$F(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{j\omega t} dt$$

Transformasi *Fourier* Diskrit

Karena sampling gelombang bunyi dilakukan pada komputer maka dilakukan transformasi *Fourier* diskrit.

Anggap bahwa sebuah deret bilangan kompleks $x(k)$ dengan sampel sebanyak N adalah:

$$x_0, x_1, x_2, x_3, \dots, x_k, \dots, x_{N-1}$$

dengan x adalah bilangan kompleks

$$x_i = x_{\text{real}} + j x_{\text{imag}}$$

Selanjutnya anggap bahwa deret diluar $0, N-1$ adalah periodik, bahwa $x_k = x_{k+N}$ untuk semua k . Hasil transformasi fourier dari deret tersebut di notasikan sebagai $X(k)$, yang juga memiliki sampel sebanyak N .

Transformasi *Fourier* untuk fungsi diskrit:

$$X(n) = \sum_{k=0}^{N-1} x(k) e^{-jk2\pi n/N} \quad n = 0, 1, \dots, N-1$$

Transformasi invers *Fourier* untuk fungsi diskrit :

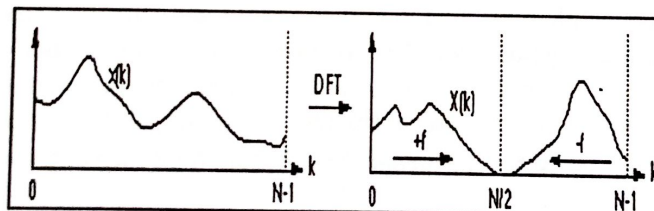
$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{jk2\pi n/N} \quad n = 0, 1, \dots, N-1$$

Pada kondisi sesungguhnya, data yang ada adalah data real saja sehingga bagian imajiner dari data dianggap nol.

Setelah transformasi dilakukan, deret yang didapat adalah nilai real dan imajiner. Selanjutnya dapat di hitung magnitudonya .

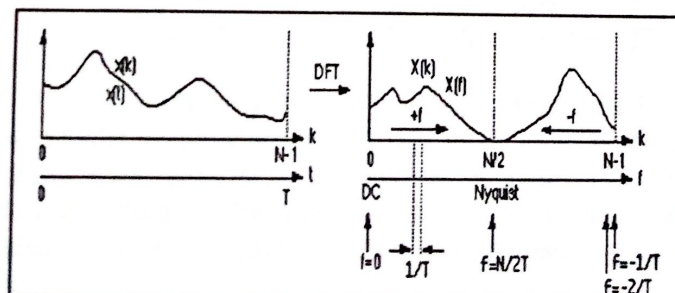
$$|X(n)| = (X_{\text{real}} * X_{\text{real}} + X_{\text{imag}} * X_{\text{imag}})^{0.5}$$

Pada gambar berikut ditunjukkan hubungan deret dalam domain waktu dan domain frekwensinya setelah tranformasi.



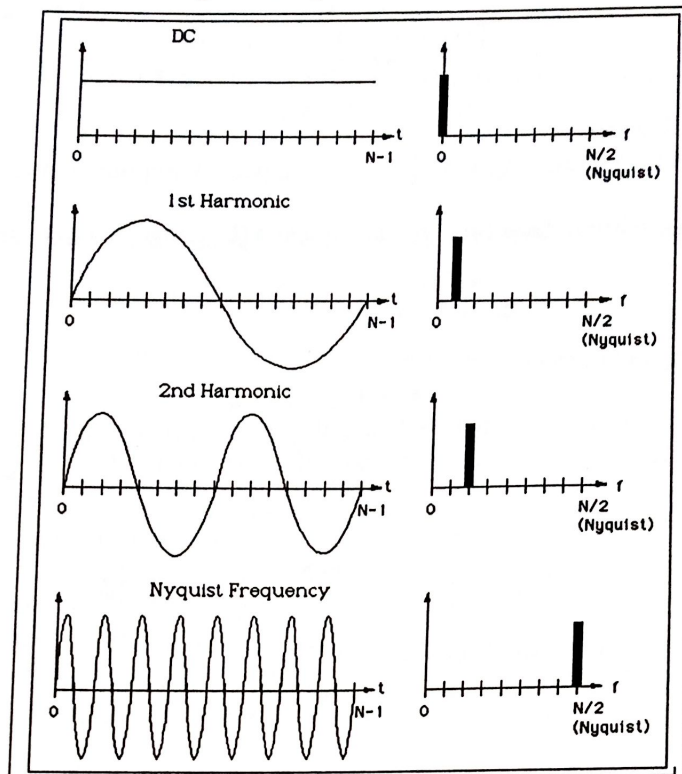
Gambar 1. Hubungan deret dalam domain waktu dan domain frekwensinya

Sebagai contoh jika dalam domain waktu suatu gelombang dengan panjang T , pada gambar 2.2 di ilustrasikan nilainya dalam domain frekwensi.



Gambar 2. Hubungan deret dalam domain waktu dan nilai dalam domain frekwensinya

Selanjutnya pada beberapa gelombang harmonik berikut akan diilustrasikan hubungan antara time domain dan frekwensi domain gelombang tersebut.



Gambar 3. Hubungan gelombang dalam domain waktu dan nilai dalam domain frekwensinya

3. SUPER RESOLUSI

Gambar dengan resolusi tinggi sangat dibutuhkan dalam aplikasi-aplikasi gambar elektronik. Gambar dikatakan memiliki resolusi tinggi jika tingkat kepadatan *pixel* dalam gambar tersebut tinggi. Karena itu gambar dengan resolusi tinggi dapat menawarkan data yang lebih lengkap, yang mana dalam suatu aplikasi tertentu merupakan suatu hal yang sangat kritis¹. Sebagai contoh, gambar resolusi tinggi pada gambar-gambar medis akan sangat membantu dokter dalam melakukan diagnosa yang benar. Akan terasa lebih mudah untuk mencirikan sebuah objek dari gambar padanannya dengan menggunakan gambar yang memiliki resolusi lebih tinggi, dan juga performansi dari sistim pengenalan pola pada komputer vision bisa ditingkatkan apabila gambar resolusi tinggi tersedia.

Algoritma super resolusi berusaha mendapatkan gambar resolusi tinggi dari beberapa gambar resolusi rendah. Gambar input haruslah menggunakan objek yang sama dan diambil dari sudut yang berbeda.

Metoda ini dikembangkan oleh Peleg *and co-workers*, dengan mengestimasi sebuah *initial guess* untuk gambar resolusi tinggi dan mensimulasikan proses penggambaran (diasumsikan diketahui) untuk mendapatkan sebuah set gambar simulasi resolusi rendah. Kemudian mendefinisikan error antara gambar aktual dengan gambar simulasi, dimana error ini akan di minimalisasi secara iterative sampai tidak ada perbaikan lagi atau sampai jumlah iterasi maksimum telah dicapai. Metoda ini akan memberikan hasil yang baik untuk gambar yang bebas *noise*.

4. SUPER RESOLUSI DENGAN ITERATIVE BACK

Pendekatan yang dilakukan IBP menyerupai permasalahan rekonstruksi gambar objek 2 dimensi dari proyeksi gambar 1 dimensinya pada *computer aided tomography* (CAT). Pada *tomography*, gambar direkonstruksi melalui proyeksinya dari segala arah. Sedangkan dalam kasus super resolusi, setiap gambar resolusi rendah adalah "proyeksi" pada gambar yang ukurannya di

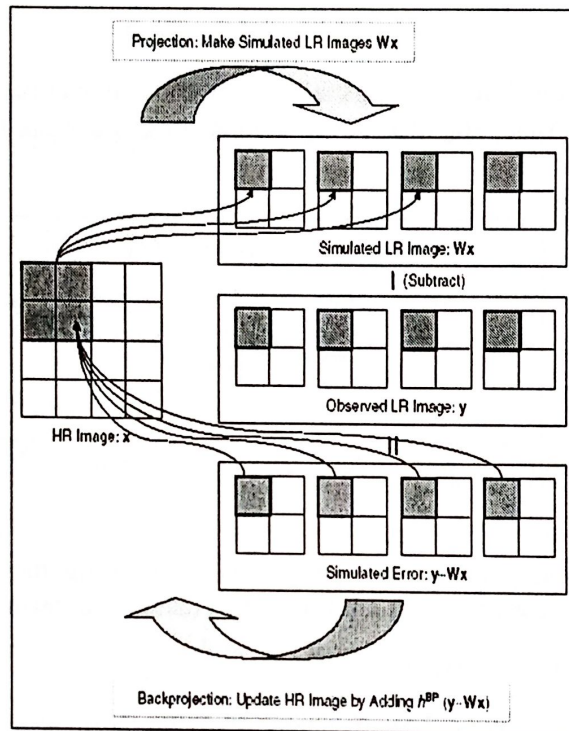
¹ Sung Cheol Park, Min Kyu Park, Moon Gi Kang, "Super Resolution Image Reconstruction" Hal: 21.

pisahkan oleh *imaging blur*. Gambar resolusi tinggi di konstruksi menggunakan sebuah pendekatan menyerupai metoda IBP yang digunakan pada CAT.

Skema IBP untuk mengestimasi gambar resolusi tinggi adalah:

$$\sum_{y \in \cup_k y_{k,x}} (g_k(y) - g_k^{(n)}(y)) \frac{(h_{xy}^{BP})^2}{\sum_{y' \in \cup_k y_{k,x}} h_{xy'}^{BP}}$$

Dimana $g_k^{(n)}(y)$ adalah satu set simulasi gambar resolusi rendah yang didapat dari hasil iterasi, yang dipengaruhi oleh (x) . Dimana (x) dan (h_{bp}) adalah kernel back projection yang akan mempengaruhi kontribusi kesalahan (error); dan juga $f^{(n+1)}(x)$. Skema IBP diilustrasikan seperti gambar berikut.



Gambar 4. Skema Iterative Back Projection (IBP)

Perangkat lunak yang dibuat adalah untuk menghasilkan suatu gambar resolusi tinggi dari beberapa jumlah gambar resolusi rendah. Dengan menggunakan algoritma *Iterative Back Projection (IBP)* akan dihasilkan gambar rekonstruksi, dengan membandingkan gambar simulasi dan gambar observasi. Selisih antara perbandingan ini akan dihitung dan digunakan untuk memperbaiki gambar super resolusi sementara dengan *back-projects*ikan setiap value pada gambar yang berbeda kedalam gambar super resolusi tujuan. Proses ini akan dilakukan secara iterasi untuk memperkecil fungsi *error*.

4.1. Proses Manipulasi Gambar

Agar dapat diolah dengan computer digital, maka suatu citra harus direpresentasikan secara numeric dengan nilai-nilai diskrit. Representasi citra dari fungsi kontinu menjadi nilai diskrit disebut dengan digitalisasi. Pada umumnya citra digital berbentuk 4 persegi panjang, dan dimensi ukurannya dinyatakan sebagai tinggi x lebar (panjang x lebar). Citra digital yang tingginya N , dan lebar M dan memiliki L derajat keabuan dapat dianggap sebagai fungsi :

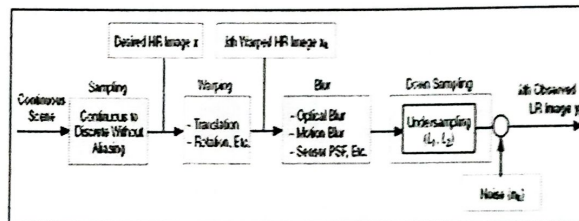
$$f(x, y) \begin{cases} 0 \leq x \leq M \\ 0 \leq y \leq N \\ 0 \leq f \leq L \end{cases}$$

Pixel merupakan elemen terkecil dalam suatu gambar, dengan kata lain *pixel* merupakan titik-titik dengan intensitas warna tertentu sepanjang media gambar yang akan membentuk suatu gambar tersendiri. Citra yang berukuran $N \times M$ mempunyai NM buah *pixel*. Sebagai contoh, misalnya sebuah gambar berukuran 256×256 *pixel* dan direpresentasikan secara numerik dengan matrik yang terdiri dari 256 buah garis dan 256 buah kolom.

Untuk dapat memudahkan dalam proses manipulasi terhadap *pixel-pixel* tersebut kita perlu untuk memisahkan warna-warna dari tiap *pixel* tersebut sesuai dengan warna primernya. Seperti yang telah disinggung diatas bahwa sistem pewarnaan yang kita gunakan disini merupakan sistem pewarnaan 24 bit, dimana dengan sistem pewarnaan ini akan terdapat 2^{24} sehingga akan dihasilkan 16777216 warna. Untuk melakukan proses manipulasi terhadap sejumlah besar warna tersebut akan banyak mengalami kesulitan karena difraksi yang terjadi akan semakin besar sehingga akan perubahan nilai sedikit akan mempengaruhi hasil perhitungan interpolasi yang cukup besar.

4.2. Model Observasi

Langkah awal untuk menganalisa permasalahan gambar super resolusi adalah dengan merumuskan model observasi yang akan menghubungkan gambar asli resolusi tinggi dengan gambar observasi resolusi rendah.



Gambar 5. Model Observasi untuk gambar resolusi rendah

4.3. Konvolusi

Konvolusi terdapat pada operasi pengolahan citra yang mengalikan sebuah citra dengan sebuah mask atau kernel. Konvolusi 2 buah fungsi $f(x)$ dan $g(x)$ didefinisikan dengan :

$$h(x) = f(x) * g(x) = \int_{-\infty}^{\infty} f(a)g(x-a)da$$

yang dalam hal ini, tanda $*$ menyatakan operator konvolusi, dan peubah a adalah peubah Bantu. Pada operasi konvolusi diatas, $g(x)$ merupakan suatu jendela yang dioperasikan secara bergeser pada sinyal masukan $f(x)$, yang dalam hal ini, jumlah perkalian kedua fungsi pada setiap titik merupakan hasil konvolusi yang dinyatakan dengan keluaran $h(x)$.

Untuk fungsi dengan dua peubah (fungsi 2 dimensi), operasi konvolusi didefinisikan sebagai berikut:

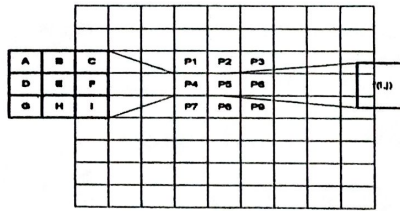
1. untuk fungsi malar

$$h(x, y) = f(x, y) * g(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(a, b)g(x-a, y-b)dadb$$

2. untuk fungsi diskrit

$$h(x, y) = f(x, y) * g(x, y) = \sum_{a=-\infty}^{\infty} \sum_{b=-\infty}^{\infty} f(a, b)g(x-a, y-b)$$

Dalam fungsi diskrit kernel konvolusi dinyatakan dalam matriks (umumnya 3×3), namun ada juga yang berukuran 2×2 atau 2×1 . Ukuran matrik biasanya lebih kecil dari ukuran citra. Setiap elemen matrik disebut koefisien konvolusi.



$$f(i, j) = Ap1 + Bp2 + Cp3 + Dp4 + Ep5 + Fp6 + Gp7 + Hp8 + Ip9$$

Gambar 6. Ilustrasi konvolusi

Operasi konvolusi dilakukan dengan menggeser kernel konvolusi *pixel per pixel*, hasil konvolusi disimpan dalam sebuah matrik yang baru.

4.4. Iterative Back Projection (IBP)

Model Observasi gambar resolusi rendah, berdasarkan sekuensial gambar observasi dimodelkan sebagai berikut :

$$g_k(m, n) = \sigma_k(h(f(x, y)) + \eta_k(x, y)),$$

dimana

- g_k adalah bingkai observasi gambar yang ke-k
- f adalah scene asli
- h adalah operator pengaburan (blurring)
- η_k adalah additive noise
- σ_k adalah sebuah fungsi non-linear yang *mendigitizes* dan *decimates* gambar ke dalam pixel dan *quantizes* besarnya hasil pixel dari intensitas ke dalam tingkat *gray*.
- (x, y) adalah pusat *receptive field* (dalam f) dari pendeteksi yang keluarannya adalah $g_k(m, n)$.

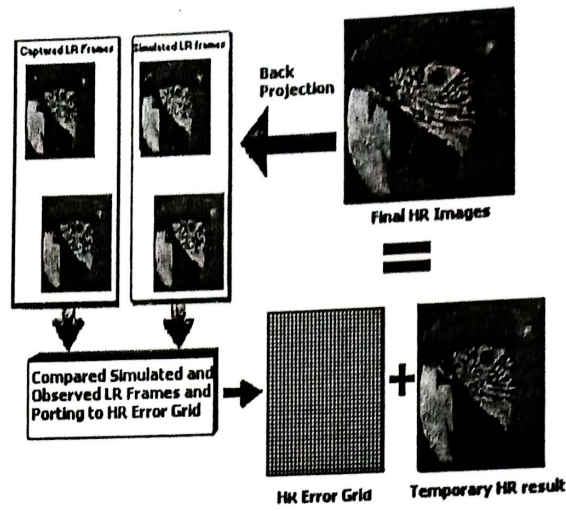
Algoritma yang diperkenalkan untuk masalah super resolusi adalah iterative. Diawali dengan sebuah *initial guess* $f^{(0)}$ untuk gambar resolusi tinggi, proses penggambaran disimulasikan untuk mendapatkan sebuah set gambar dengan resolusi rendah $\{g_k^{(0)}\}$ akan dihubungkan dengan gambar input observasi $\{g_k\}$. Jika $f^{(0)}$ adalah gambar dengan resolusi tinggi yang diinginkan, maka gambar simulasi $\{g_k^{(0)}\}$ akan identik dengan gambar observasi $\{g_k\}$. Kemudian perbedaan gambar $\{g_k - g_k^{(0)}\}$ akan dihitung dan hasilnya digunakan untuk memperbaiki *initial guess* dengan *memback proyeksikan* setiap value pada gambar-gambar yang berbeda kedalam *receptive field*-nya di $f^{(0)}$. Proses ini akan diulang secara iterasi untuk meminimalisasi fungsi error

$$e^{(n)} = \sqrt{\sum_k \sum_{(x,y)} (g_k(x, y) - g_k^{(n)}(x, y))^2}$$

Dimana,

$g_k(x, y)$ adalah gambar observasi resolusi rendah

$g_k^{(n)}(x, y)$ adalah gambar simulasi resolusi rendah



Gambar 7. Blok deskripsi Iterative Back Projection

Notasi –notasi yang digunakan adalah :

- f , gambar target resolusi tinggi yang akan di dibangun (tidak diketahui);
- $f^{(n)}$, f yang didapat setelah n iterasi;
- $\{g_k\}$, observasi yang ke k gambar resolusi rendah;
- $\{g_k^{(n)}\}$, gambar resolusi rendah yang didapat dari simulasi proses penggambaran ke $f^{(n)}$. Jika $f^{(n)}$ adalah hasil resolusi tinggi yang benar maka $\{g_k^{(n)}\} = \{g_k\}$;
- h^{PSF} , fungsi *point spread* dari *imaging blur*;
- h^{BP} , kernel back projection;
- x , menunjukkan pixel resolusi tinggi ;
- y , menunjukkan pixel resolusi rendah (yang dipengaruhi oleh x);

5. ANALISA

5.1. Analisa Kompresi menggunakan DFT

Analisa dari kompresi menggunakan DFT dilakukan dengan mengkompresi file ding.wav (file ding.wav ada direktori windows).

Header File ding.wav :

SampleRate = 22.050

NumChannels = 2

BitsPerSample = 16

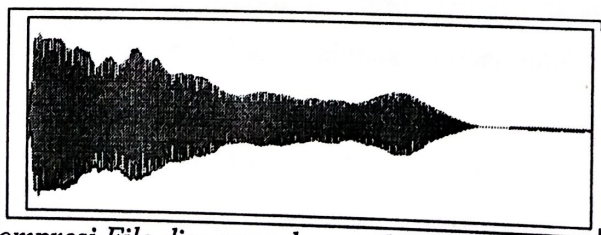
NumSamples = 20191

5.1.1. Kompresi File ding.wav dengan koefisien fourier 1 sampai 1000

MSE = 29.005

Waktu yang dibutuhkan :

- Pembacaan file wav : 0,01 detik
- Kompresi dengan DFT : 18 detik
- Dekompresi dengan infers DFT : 19 detik
- Total Waktu = 37,01 detik



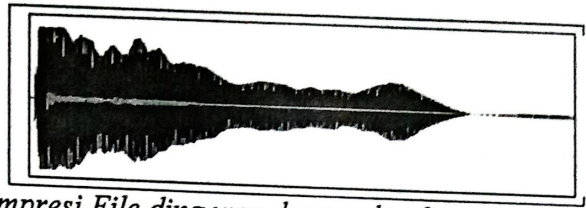
Gambar 8. Kompresi File ding.wav dengan koefisien fourier 1 sampai 1000

5.1.2. Kompresi File ding.wav dengan koefisien fourier 600 sampai 700

MSE = 3.869.513

Waktu yang dibutuhkan :

- Pembacaan file wav : 0,01 detik
- Kompresi dengan DFT : 1 detik
- Dekompresi dengan infers DFT : 2 detik
- Total Waktu = 3,01 detik



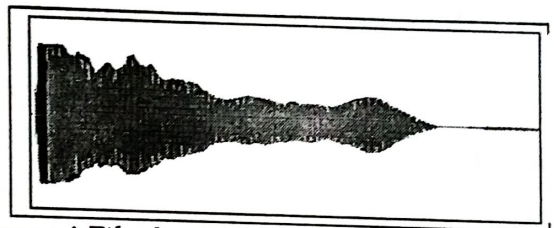
Gambar 9. Kompresi File ding.wav dengan koefisien fourier 600 sampai 700

5.1.3. Kompresi File ding.wav dengan koefisien fourier 600 sampai 800

MSE = 326.845

Waktu yang dibutuhkan :

- Pembacaan file wav : 0,01 detik
- Kompresi dengan DFT : 3 detik
- Dekompresi dengan infers DFT : 4 detik
- Total Waktu = 7,01 detik



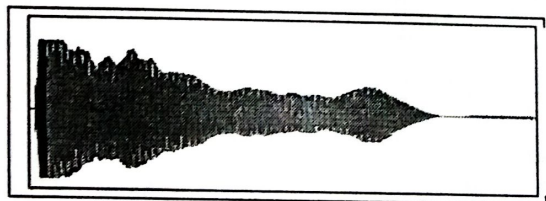
Gambar 10. Kompresi File ding.wav dengan koefisien fourier 600 sampai 800

5.1.4. Kompresi File ding.wav dengan koefisien fourier 600 sampai 1000

MSE = 326.716

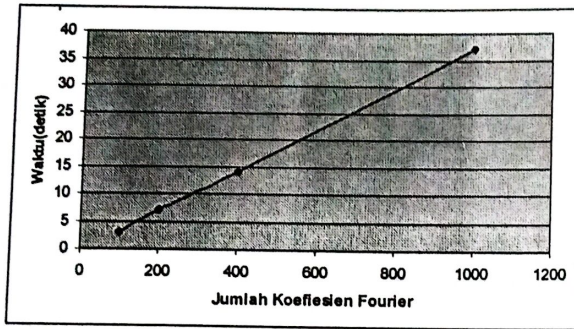
Waktu yang dibutuhkan :

- Pembacaan file wav : 0,01 detik
- Kompresi dengan DFT : 7 detik
- Dekompresi dengan infers DFT : 7 detik
- Total Waktu = 14,01 detik

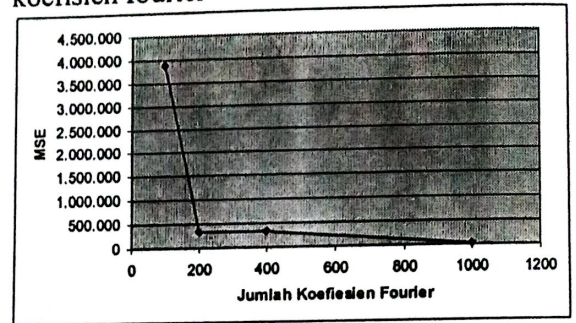


Gambar 11. Kompresi File ding.wav dengan koefisien fourier 600 sampai 1000

Grafik perbandingan waktu terhadap jumlah koefisien fourier



Grafik perbandingan MSE terhadap jumlah koefisien fourier



5.2. Analisa Peningkatan Resolusi Gambar dengan IBP

Analisa perangkat lunak dilakukan dengan perbandingan PSNR Super Resolusi IBP dengan Zoom menggunakan Adobe Photoshop

Data Gambar : Bunga (Warna 24 bit)

Panjang x Lebar : 88 x 109

Perbesaran IBP 2x

PSNR = 49 MSE = 464

Perbesaran Adobe Photoshop 2x

PSNR = 46 MSE = 649

Perbesaran IBP 4x

PSNR = 39 MSE = 1274

Perbesaran Adobe Photoshop 4x

PSNR = 39 MSE = 1925

Data Gambar : Burung (Gray Scale)

Panjang x Lebar : 144 x 108

Perbesaran IBP 2x

PSNR = 54 MSE = 283

Perbesaran Adobe Photoshop 2x

PSNR = 49 MSE = 303

Perbesaran IBP 4x

PSNR = 44 MSE = 771

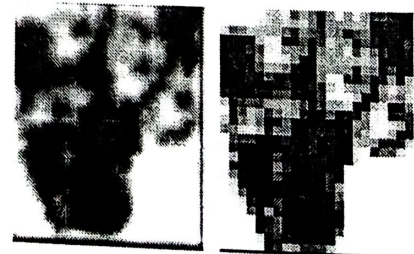
Perbesaran Adobe Photoshop 4x

PSNR = 40 MSE = 1157



(a)

(b)



(c)

(d)

Gambar 12. Bunga
 (a) Perbesaran 2x IBP (b) Perbesaran skala 2x
 (c) Perbesaran 4x IBP (d) Perbesaran skala 4x



Gambar 13. Burung
 (a) Perbesaran 2x IBP (b) Perbesaran skala 2x
 (c) Perbesaran 4x IBP (d) Perbesaran skala 4x

6. KESIMPULAN

Dari hasil pembuatan dan analisa penelitian ini maka dapat ditarik beberapa kesimpulan, yaitu :

Kompresi Audio

1. Kompresi menggunakan DFT dengan menggunakan 2000 koefisien (1000 koefisien sinus dan 1000 koefisien cosinus) mempunyai kualitas suara yang hampir sama dengan aslinya, tetapi mempunyai proses encoding dan decoding 37 detik.
2. Karena yang di kompresi adalah suara manusia maka tidak semua koefisien yang dipakai cukup koefisien yang sama dengan frekwensi suara manusia, pada penelitian ini di uji dengan menggunakan koefisien sinus cosinus antara 600 dan 800 (200 koefisien sinus dan 200 koefisien cosinus).
3. Perbandingan data wav asli dengan durasi 1 detik dan data terkompresi dengan menggunakan DFT adalah 1 : 50 jika menggunakan koefisien 400.
4. Kecepatan mengkompresi adalah berbanding lurus dengan banyaknya koefien yang dipakai.

Kompresi Video

1. Dengan Interactive Back Projection dapat meningkatkan kualitas image (1 frame video) yang lebih baik jika meningkatkan kualitas image dengan memperbesar pixel (pada penelitian ini perbesaran image IBP dibandingkan dengan perbesaran dengan Tool Adobe Photoshop)
2. Metode IBP adalah membandingkan dengan menggunakan iterasi/pengulangan tetapi untuk mendapatkan hasil yang optimal cukup dilakukan dengan 3 sampai 5 kali iterasi, karena setelah itu hasilnya akan konvergen.

7. REFERENSI

1. Matthew, Fourier and the Frequency Domain, <http://www.spd.eee.strath.ac.uk/~interact/FFT/ft.html>
2. Gold, B dan Morgan, N , Speech and Audio Signal Processing, John Wiley and Sond inc, New York.
3. Bourke, P 1993, Dicrete Fourier Transform, <http://astronomy.swin.edu.au/~pbourke/analysis/dft/>
4. Giuseppe Messina, Sebastiano Battiato, Massimo Mancuso and Antonio Buemi, Improving Image Resolution by adaptive Back Projection Correction Techniques
5. IEEE Signal Processing Magazine, may 2003 vol 20, no 3.
6. Michal Irani, Shmuel Peleg, Image Sequence Enhancement Using Multiple Motions Analysis Institute of Computer Science The Hebrew University of Jerusalem, 91904 Jerusalem, Israel.
7. Michal Irani and S. Peleg, Improving Resolution by Image Registration, Department of Computer Science The Hebrew University of Jerusalem, Jerusalem, Israel. Accepted May 25, 1990
8. Michal Irani and S. Peleg, Super Resolution from Image Sequence. Department of Computer Science The Hebrew University of Jerusalem, Jerusalem, Israel.
9. Nick Efford, Digital Image Processing, a practical introduction using java, School of Computer Studies University of Leeds".
10. Sung Cheol Park, Min Kyu Park, Moon Gi Kang, " Super Resolution Image Reconstruction".