

Analisis Perbandingan Kompresi Haar Wavelet Transform dengan Embedded Zerotree Wavelet pada Citra

LEDYA NOVAMIZANTI, ADRIAN KURNIA

Program Studi S1 Teknik Telekomunikasi Fakultas Teknik Elektro
Telkom University

Email : ledyaldn@telkomuniversity.ac.id

ABSTRAK

Kompresi data merupakan salah satu teknologi pemicu revolusi multimedia. Haar Wavelet mampu merepresentasikan ciri tekstur dan bentuk, sedangkan Embedded Zerotree Wavelet (EZW) mampu menyusun bit-bit menurut tingkat prioritas, sehingga mampu mencapai kompresi maksimal. Pada penelitian ini telah dilakukan perbandingan Haar Wavelet Transform dengan Embedded Zerotree Wavelet untuk kompresi citra. Pengujian menggunakan 4 citra grayscale berformat bitmap (.bmp) dengan resolusi 256x256 dan 512x512. Rasio Kompresi yang diperoleh dengan menggunakan algoritma Embedded Zerotree Wavelet dan Haar Wavelet, yaitu 99.54% dan 95.35% pada threshold 80. Laju bit antara Embedded Zerotree Wavelet lebih rendah dibandingkan Haar Wavelet, yaitu 0.06 bpp dan 0.13 bpp. Algoritma Haar Wavelet memberikan waktu kompresi lebih baik dibandingkan EZW dimana selisih antara keduanya sekitar 8 detik.

Kata kunci: kompresi citra, threshold, Haar Wavelet, Embedded Zerotree Wavelet

ABSTRACT

Data compression is one of the triggers of the revolution multimedia technology. Haar Wavelet able to represent the characteristics of texture and shape, while Embedded Zerotree Wavelet (EZW) is able to arrange the bits according to priority level, so as to achieve maximum compression. In this study, we had conducted comparison between Haar Wavelet Transform with Embedded Zerotree Wavelet algorithm for image compression. The tests using 4 image format grayscale bitmap (.bmp) with resolution of 256x256 pixels and 512x512 pixels. Compression ratio obtained using Embedded Zerotree Wavelet and Wavelet Haar algorithm, which are 99.54% and 95.35% respectively, at the threshold of 80. The bit rate on Embedded Zerotree Wavelet is lower than Haar wavelet, that is 0:06 bpp and 0:13 bpp respectively. Haar Wavelet algorithm gives a better compression time than the EZW, with the difference between the two is about 8 seconds.

Keywords: image compression, threshold, Haar Wavelet, Embedded Zerotree Wavelet.

1. PENDAHULUAN

Pada kurun waktu 20 tahun terakhir terjadi transformasi atau revolusi dalam cara kita berkomunikasi. Hal ini dapat dilihat pada perkembangan internet, komunikasi *mobile* yang berkembang sangat cepat, dan komunikasi multimedia. Salah satu teknologi pemicu revolusi multimedia adalah teknologi kompresi data (**Sayood, 2000**).

Kompresi data adalah cara untuk merepresentasikan informasi menjadi bentuk kompak. Tujuan kompresi yaitu untuk memperkecil ukuran sebuah data. Beberapa alasan pentingnya kompresi data adalah menghemat kapasitas penyimpanan (*storage space*), mengurangi waktu transmisi, proses transmisi dapat lebih progresif, dan mengurangi komputasi (**Salomon, 1998**). Semakin besarnya ukuran file citra mempengaruhi besarnya tempat untuk menyimpan file tersebut. Besarnya ukuran file citra juga berdampak pada *space* untuk menyimpan data pada *storage*. Ukuran citra yang besar dapat mengakibatkan lamanya proses download pada halaman situs. Untuk itulah, kompresi citra (*image compression*) yang merupakan salah satu bidang penting dalam pengolahan citra digital perlu dipelajari.

Kompresi citra menjadi salah satu alternatif yang dapat dipakai untuk menyelesaikan masalah diatas. Ada banyak metode yang dapat digunakan dalam mengkompresi citra, seperti Huffman Coding, LZW coding, Run Length Coding, dan sebagainya. Salah satu metoda domain transform adalah metode Haar Wavelet Transform dan Embedded Zerotree Wavelet (EZW). Haar Wavelet merupakan metode yang lebih baik digunakan untuk merepresentasikan ciri tekstur dan bentuk (**Arora, 2014**). Embedded Zerotree Wavelet (EZW) mampu menyusun bit-bit menurut tingkat prioritas, hasilnya adalah sebuah kode penempelan penuh (*fully embedded*), sehingga mampu mencapai kompresi yang maksimal (**Shapiro, 1993**).

Anwar (**Anwar, 2008**) telah melakukan penelitian untuk kompresi citra medis menggunakan metode DWT dan EZW pada citra RGB. Diperoleh citra hasil kompresi dengan kualitas layak (reasonable) karena menghasilkan nilai PSNR ≥ 40 dB. Penelitian lainnya yaitu kompresi citra digital telah diimplementasikan menggunakan wavelet Daubechies dan diuji berdasarkan parameter laju bit dan PSNR (**Sianipar, 2003**). Parameter PSNR menunjukkan bahwa wavelet db4 menjadi yang terbaik kecuali untuk citra uji Daubechies.

Pada penelitian ini, akan dibandingkan performansi antara metode Haar Wavelet Transform dengan EZW yang menggunakan filter Haar Wavelet dalam pengkompresian citra. Hasil performansi berupa rasio kompresi (%), laju bit (bpp), MSE, PSNR dan waktu kompresi (detik). Dari hasil parameter performansi tersebut, akan diperoleh metode mana yang terbaik dalam kompresi citra.

1.1 Pemampatan Citra

Sebuah citra 8 bit dengan dimensi 256x256 pixel membutuhkan ruang penyimpanan sebesar $256 \times 256 \times 8 \text{ bit} = 524.288 \text{ bit}$. Semakin besar ukuran citra maka semakin lama waktu pengiriman yang dibutuhkan untuk mengirimkannya.

Pemampatan citra dilakukan untuk mengurangi kebutuhan ruang penyimpanan citra dan kebutuhan *bandwidth* dalam proses pengiriman (**Gonzales, 2002**). Setelah keduanya tercapai, biaya penyimpanan, pengiriman, dan pemrosesan dapat dikurangi.

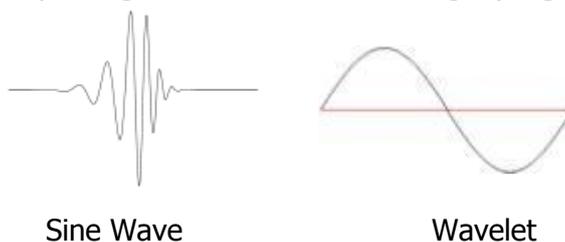
1.2 Wavelet

Wavelet adalah suatu teknik matematika yang bermanfaat untuk analisis numerik dan manipulasi dari kumpulan satu atau dua dimensi sinyal. Proses transformasinya bekerja seperti sebuah mikroskop yang menguji secara rinci dengan membagi sinyal ke dalam komponen frekuensi yang berbeda yang dipetakan ke dalam koefisien dengan energi yang berbeda (Jones, 1999).

Inti dari *wavelet* adalah bagaimana mendekomposisi sinyal f ke dalam basis fungsi ψ_i dengan i merupakan level dekomposisi. Hal ini dapat dirumuskan dalam Persamaan 1:

$$f = \sum a_i \psi_i \quad (1)$$

Untuk mendapatkan representasi yang efisien dari sinyal f yang hanya memanfaatkan sedikit koefisien a_i , maka sangat penting untuk menentukan fungsi yang sesuai dengan data yang ingin direpresentasikan.



Gambar 1. Wave dan wavelet

Wavelet berbeda dengan *wave*, *wave* merupakan fungsi osilasi dari ruang atau waktu. *Wavelet* disebut juga dengan 'wave kecil' yang memiliki energi berhingga dan terlokalisasi di sekitar sebuah titik, serta fungsinya memiliki nilai rata-rata sama dengan nol.

1.3 Discrete Wavelet Transform (DWT)

Discrete Wavelet Transform (DWT) merupakan proses fungsi wavelet dengan pemilihan subset dari skala dan posisi tertentu dalam proses kalkulasi. DWT menggunakan proses pemfilteran untuk menghasilkan koefisien wavelet (Misiti, 2002).

Dalam DWT, sebuah sinyal citra dapat dianalisis dengan melewatkannya melalui proses *filtering* yang dilanjutkan dengan operasi penipisan. Proses *filtering* terdiri dari *low-pass* dan *high-pass filter* pada setiap tahap dekomposisi. Sinyal citra akan terbagi menjadi dua band ketika melewati filter ini. *Low-pass filter*, yang berhubungan dengan operasi rata-rata, akan mengekstraksi informasi kasar dari sebuah sinyal. *High-pass filter*, yang berhubungan dengan *differencing operation*, akan mengekstraksi informasi detail dari sinyal. Hasil operasi *filtering* ini kemudian ditipiskan melalui proses *down-sampling*.

Secara matematis, operasi rata-rata atau *low-pass filtering* merupakan *inner product* antara sinyal dengan fungsi *scaling* (φ). Hal ini dapat dirumuskan dalam Persamaan 2:

$$c_j(k) = \langle f(t), \varphi_{j,k}(t) \rangle = \int f(t), \varphi_{j,k}(t) dt \quad (2)$$

Di sisi lain, operasi detail atau *high-pass filtering* merupakan *inner product* antara sinyal dengan fungsi *wavelet* (ψ). Hal ini dapat dirumuskan dalam Persamaan 3:

$$d_j(k) = \langle f(t), \psi_{j,k}(t) \rangle = \int f(t), \psi_{j,k}(t) dt \quad (3)$$

Fungsi *scaling* dapat didefinisikan dalam Persamaan 4 :

$$\varphi_{j,k}(t) = 2^{j/2}\varphi(2^j t - k) \tag{4}$$

Fungsi *wavelet* dapat didefinisikan dalam Persamaan 5 :

$$\psi_{j,k}(t) = 2^{j/2}\psi(2^j t - k) \tag{5}$$

dengan *j* merupakan indeks *scaling* dan *k* merupakan indeks translasi (Tan, 2001).

1.4 Down-sampling dan Up-sampling

Pada proses dekomposisi dilakukan *down-sampling* setelah proses pemfilteran untuk mengurangi jumlah output dari masing-masing filter sehingga sama dengan jumlah input.



Gambar 2. Down Sampling

Misalkan input sinyal $x(n)$, setelah proses *down-sampling* akan menghasilkan *output* $y(n) = x(2n)$ seperti pada Gambar 2 dengan hanya mempertahankan indeks genap.

Pada proses rekonstruksi dilakukan proses *up-sampling* yang dilanjutkan dengan pemfilteran. Proses *up-sampling* membentangkan sinyal input 2 kali panjang awal dan menyisipkan nilai 0 pada indeks ganjil. Dengan kata lain dapat dituliskan $y(2n) = x(n)$ dan $y(2n+1) = 0$.

1.5 Haar Wavelet Transform

Wavelet Haar merupakan *wavelet* yang paling tua dan sederhana (Subahar, 2009). Alasan menggunakan *Haar Wavelet* karena merupakan metode yang lebih bagus digunakan untuk merepresentasikan ciri tekstur dan bentuk. Disamping itu *Haar Wavelet* memerlukan waktu komputasi yang lebih kecil dari pada transformasi *wavelet* lainnya, ciri diperoleh dari citra yang telah melewati proses dekomposisi, yakni *double* untuk dapat dioperasikan. Gambar 3 ditunjukkan matriks HAAR ukuran $L=8 \times 8$. Baris 1 sampai $L/2$ merupakan filter low pass dan baris $L/2+1$ sampai L adalah filter high pass.

```

    >> MatriksHaar(8)
    0.7071  0.7071  0  0  0  0  0  0
      0  0  0.7071  0.7071  0  0  0  0
      0  0  0  0  0.7071  0.7071  0  0
      0  0  0  0  0  0  0.7071  0.7071
    0.7071 -0.7071  0  0  0  0  0  0
      0  0  0.7071 -0.7071  0  0  0  0
      0  0  0  0  0.7071 -0.7071  0  0
      0  0  0  0  0  0  0.7071 -0.7071
  
```

Gambar 3. Matriks Haar Ukuran 8x8

Basis wavelet merupakan model yang berbeda-beda dari keluarga wavelet yang memiliki kualitas beragam sesuai dengan kriteria yang dimiliki (Misiti et al. 2002).

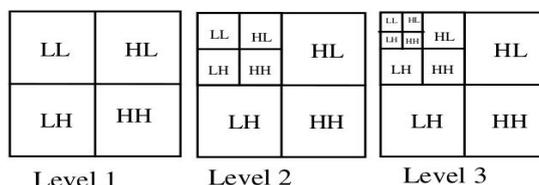
Pada wavelet Haar, *wavelet function* didefinisikan dalam Persamaan 6 :

$$\psi(t) = \begin{cases} 1 & 0 < t < 0.5 \\ -1 & 0.5 < t < 1 \\ 0 & \text{selainnya} \end{cases} \quad (6)$$

Untuk *scaling function* pada *wavelet* Haar didefinisikan dalam Persamaan 7 :

$$\varphi(t) = \begin{cases} 1 & 0 < t < 1 \\ 0 & \text{selainnya} \end{cases} \quad (7)$$

Dalam proses dekomposisi, *wavelet* Haar menerapkan *bank filter* dengan $h(0) = h(1) = 1/\sqrt{2}$ sebagai koefisien *low-pass* yang menghasilkan citra pendekatan dan $g(0) = 1/\sqrt{2}$, $g(1) = -1/\sqrt{2}$ sebagai koefisien *high-pass* yang menghasilkan citra detail. Citra hasil dekomposisi akan berukuran seperempat dari ukuran citra sebenarnya. Ilustrasi proses dekomposisi citra dapat dilihat pada Gambar 4.



Gambar 4. Ilustrasi Proses Dekomposisi Citra

Untuk menghitung transformasi Haar Wavelet dari berbagai sampel n : **(Arora, 2014)**

1. Menghitung rata-rata setiap pasangan sampel. ($n / 2$ average).
2. Menghitung perbedaan antara masing-masing rata-rata dan sampel itu dihitung dari ($n/2$ differences).
3. Tuliskan bagian pertama dari array dengan rata-rata (*averages*).
4. Menulis paruh kedua array dengan perbedaan (*differences*).
5. Ulangi proses pada paruh pertama dari array. Sementara melakukan ini ukuran array harus dibagi dua.

Haar menggunakan tiga langkah dalam pengkodean citra. Berikut citra yang digunakan adalah 8x8.

45	60	21	32	45	21	30	31
30	34	54	22	11	18	8	10
78	36	38	33	31	40	47	50
11	19	9	7	28	25	10	0
11	1	2	3	33	35	0	0
11	56	77	65	43	32	12	0
10	13	42	39	43	12	24	20
27	80	78	40	49	50	27	21

Gambar 5. Ilustrasi Contoh Citra 8x8

Langkah pertama, hitung nilai rata-rata (*average*) untuk setiap pasangan piksel atau sampel pada citra. Yang dilakukan adalah menghitung rata-rata setiap sampel sesuai baris. Dan kemudian diletakkan pada bagian kiri. Proses menghitung rata-rata pasangan piksel sesuai baris untuk contoh piksel pada Gambar 5, dapat dilihat pada Gambar 6.

$(45+60)/2$	$(21+32)/2$	$(45+21)/2$	$(30+31)/2$	$(45-60)/2$	$(21-32)/2$	$(45-21)/2$	$(30-31)/2$
$(30+34)/2$	$(54+22)/2$	$(11+18)/2$	$(8+10)/2$	$(30-34)/2$	$(54-22)/2$	$(11+18)/2$	$(8+10)/2$
$(78+36)/2$	$(38+33)/2$	$(31+40)/2$	$(47+50)/2$	$(78-36)/2$	$(38-33)/2$	$(31+40)/2$	$(47+50)/2$
$(11+19)/2$	$(9+7)/2$	$(28+25)/2$	$(10+0)/2$	$(11-19)/2$	$(9-7)/2$	$(28-25)/2$	$(10-0)/2$
$(11+1)/2$	$(2+3)/2$	$(33+35)/2$	$(0+0)/2$	$(11-1)/2$	$(2-3)/2$	$(33-35)/2$	$(0-0)/2$
$(11+56)/2$	$(77+65)/2$	$(43+12)/2$	$(24+20)/2$	$(11-56)/2$	$(77-65)/2$	$(43-12)/2$	$(24-20)/2$
$(10+13)/2$	$(42+39)/2$	$(43+12)/2$	$(24+20)/2$	$(10-13)/2$	$(42-39)/2$	$(43-12)/2$	$(24-20)/2$
$(27+80)/2$	$(78+40)/2$	$(49+50)/2$	$(27+21)/2$	$(27-80)/2$	$(78-40)/2$	$(49-50)/2$	$(27-21)/2$

Gambar 6. Ilustrasi Proses Perhitungan Rata-rata Pasangan Pixel Sesuai Baris

Hasil dari perhitungan rata-rata pasangan pixel untuk tahap pertama, dapat dilihat pada Gambar 7.

52.5	26.5	33	30.5	-7.5	-5.5	12	-0.5
32	38	14.5	9	-2	16	-3.5	-1
57	35.5	35.5	48.5	21	2.5	-4.5	-1.5
15	8	26.5	5	-4	1	1.5	5
6	2.5	34	0	5	-0.5	-1	0
33.5	71	37.5	6	-22.5	6	5.5	6
11.5	40.5	27.5	22	-1.5	1.5	15.5	2
53.5	59	49.5	24	-26.5	19	-0.5	3

Gambar 7. Ilustrasi Hasil Perhitungan Rata-rata Pasangan Pixel Sesuai Baris

Langkah kedua, untuk 2D, ulangi langkah sebelumnya, namun dilakukan proses perhitungan rata-rata pasangan pixel dari atas ke bawah sesuai kolom. Proses menghitung rata-rata pasangan pixel sesuai kolom untuk contoh pixel hasil pada Gambar 7, dapat dilihat pada Gambar 8.

$(52.5+32)/2$	$(26.5+38)/2$	$(33+14.5)/2$	$(30.5+9)/2$	$(-7.5+-2)/2$	$(-5.5+16)/2$	$(12+-3.5)/2$	$(-0.5+-1)/2$
$(57+15)/2$	$(35.5+8)/2$	$(35.5+26.5)/2$	$(48.5+5)/2$	$(21+-4)/2$	$(2.5+1)/2$	$(-4.5+1.5)/2$	$(-1.5+5)/2$
$(6+33.5)/2$	$(2.5+71)/2$	$(34+37.5)/2$	$(0+6)/2$	$(5+-22.5)/2$	$(-0.5+6)/2$	$(-1+5.5)/2$	$(0+6)/2$
$(11.5+53.5)/2$	$(40.5+59)/2$	$(27.5+49.5)/2$	$(22+24)/2$	$(-1.5+-26.5)/2$	$(1.5+19)/2$	$(15.5+-0.5)/2$	$(2+3)/2$
$(52.5-32)/2$	$(26.5-38)/2$	$(33-14.5)/2$	$(30.5-9)/2$	$(-7.5-2)/2$	$(-5.5-16)/2$	$(12-3.5)/2$	$(-0.5-1)/2$
$(57-15)/2$	$(35.5-8)/2$	$(35.5-26.5)/2$	$(48.5-5)/2$	$(21--4)/2$	$(2.5-1)/2$	$(-4.5-1.5)/2$	$(-1.5-5)/2$
$(6-33.5)/2$	$(2.5-71)/2$	$(34-37.5)/2$	$(0-6)/2$	$(5-22.5)/2$	$(-0.5-6)/2$	$(-1-5.5)/2$	$(0-6)/2$
$(11.5-53.5)/2$	$(40.5-59)/2$	$(27.5-49.5)/2$	$(22-24)/2$	$(-1.5-26.5)/2$	$(1.5-19)/2$	$(15.5-0.5)/2$	$(2-3)/2$

Gambar 8. Ilustrasi Proses Perhitungan Rata-rata Pasangan Pixel Sesuai Kolom

Hasil dari perhitungan rata-rata pasangan pixel untuk tahap kedua dapat dilihat pada Gambar 9.

42.25	32.25	23.75	19.75	-4.75	5.25	4.25	-0.75
36	21.75	31	26.75	8.5	1.75	-1.5	1.75
19.75	36.75	35.75	3	-8.75	2.75	2.25	3
32.5	49.75	38.5	23	-14	10.25	7.5	2.5
10.25	-5.75	9.25	10.75	-2.75	-10.75	7.75	0.25
21	13.75	4.5	21.75	12.5	0.75	-3	-3.25
-13.75	-34.25	-1.75	-3	13.75	-3.25	-3.25	-3
-21	-9.25	-11	-1	12.5	-8.75	8	-0.5

Gambar 9. Ilustrasi Hasil Perhitungan Rata-rata Pasangan Pixel Sesuai Kolom

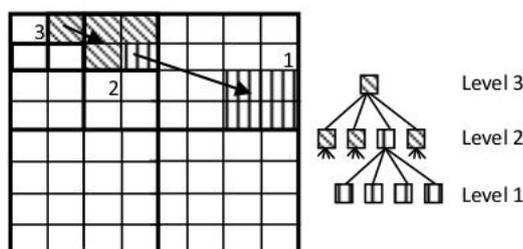
Langkah ketiga, ambil matriks awal dengan ukuran (baris/2) x (kolom/2). Matriks yang diambil merupakan matriks *Approximate*, matriks ini merupakan hasil dari kompresi dengan menggunakan metode Haar. Hasil dari matrik kompresi, dapat dilihat pada Gambar 10.

42.25	32.25	23.75	19.75
36	21.75	31	26.75
19.75	36.75	35.75	3
32.5	49.75	38.5	23

Gambar 10. Ilustrasi Hasil Kompresi Metode Haar

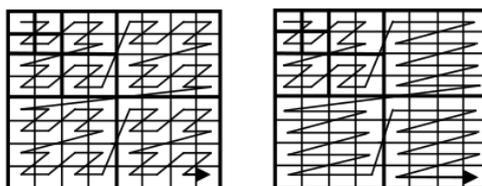
1.6 Embedded Zerotree Wavelet (EZW)

DWT menghasilkan struktur subbidang hierarki, yaitu koefisien pada setiap subbidang dan pada setiap tingkatan dapat dihubungkan dengan satu set koefisien-koefisien di tingkat yang lebih rendah pada subbidang yang sesuai. Suatu koefisien pada tingkat yang lebih tinggi dinamakan induk dari semua koefisien di orientasi ruang yang sama pada tingkat yang lebih rendah. Hubungan induk dan anakan dalam hierarki DWT ditunjukkan pada Gambar 11.



Gambar 11. Hierarki DWT

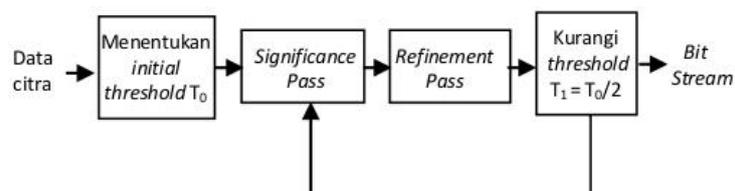
Penelusuran koefisien-koefisien dilakukan sedemikian sehingga tidak ada anak yang ditelusuri sebelum induknya. Penelusuran koefisien dapat dilakukan dengan dua cara, yaitu raster scan dan morton scan seperti pada Gambar 12.



Gambar 12. Raster Scan dan Morton Scan

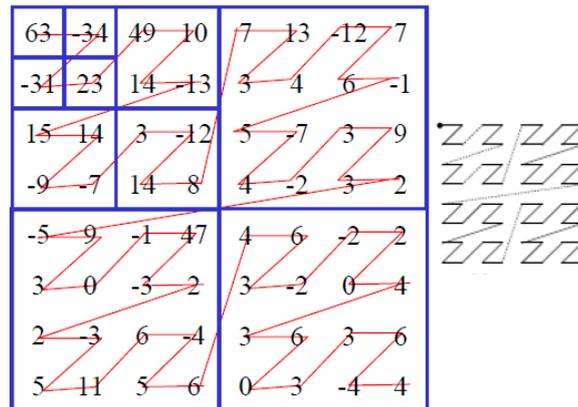
EZW menggunakan dua langkah dalam pengkodean citra, yaitu significance pass dan refinement pass. Significance pass menghasilkan significance_map yang berisi kumpulan simbol-simbol dari data citra dan subordinate_list yang berisi kumpulan nilai koefisien data citra yang signifikan. Refinement pass merupakan proses perbaikan nilai pada subordinate_list dan pengkodean yang menghasilkan refinement berisi 0 atau 1 untuk tiap nilai subordinate_list (Sasongko, 2008).

Diagram blok algoritma EZW dapat dilihat pada Gambar 13.



Gambar 13. Diagram Blok Algoritma EZW

Contoh :



Gambar 14. Contoh 3-level WT dari Citra 8x8

Pada Gambar 14, karena nilai piksel terbesar adalah 63, maka pilih nilai threshold antara (31.5 , 63]. Misal dipilih nilai threshold 32. Pada Gambar 15 diperoleh hasil piksel setelah melewati thresholding pertama.

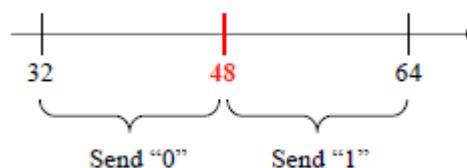
63	-34	49	10	7	13	-12	7
-34	-23	14	-13	3	4	6	-1
15	-14	3	-12	5	-7	3	9
-9	-7	14	8	4	-2	3	2
-5	9	-1	47	4	6	-2	2
3	0	-3	2	3	-2	0	4
2	-3	6	-4	3	6	3	6
5	-11	5	6	0	3	-4	4

Gambar 15. Nilai setelah thresholding pertama

Pada Gambar 15, diperoleh nilai rekonstruksi dari dominant pass yang bernilai tengah antara (32,64] yaitu 48.

Tabel 1. Tabel Proses Dominant Pass

Magnitude Koefisien	Magnitude Rekonstruksi
63	48
-34	-48
49	48
47	48



Gambar 16. Citra Rekonstruksi Hasil Dominant Pass

Kemudian proses selanjutnya adalah refining (subordinate pass) dari magnitude yang telah ditetapkan, yaitu (32,64].

Didapatkan stream dari Sub Pass pertama: 1 0 1 0.

Selanjutnya akan di rekonstruksi berdasarkan nilai magnitude setelah thresholding. Apabila nilai magnitude (32,48] maka akan direkonstruksi dengan nilai 40, yaitu nilai tengah antara (32,48]. Dan apabila nilai magnitude (48,64] maka akan direkonstruksi dengan nilai 56, yaitu nilai tengah antara (48,64]. Maka hasil rekonstruksinya dapat dilihat dari Tabel 2.

Tabel 2. Tabel Proses Subordinate Pass

Magnitude Koefisien	Simbol	Magnitude Rekonstruksi
63	1	56
-34	0	-40
49	1	56
47	0	40

Iterasi akan terus dilakukan, sampai threshold berada di bawah nilai minimum threshold.

2. METODOLOGI PENELITIAN

Tujuan dari penelitian ini adalah mengimplementasikan dan menganalisis metode Haar Wavelet Transform dengan EZW yang menggunakan filter Haar Wavelet dalam pengkompresian citra. Hasil performansi berupa rasio kompresi (%), laju bit (bpp), MSE, PSNR dan waktu kompresi (detik). Maka untuk mencapai tujuan tersebut, langkah-langkah metodologi penelitian yang dilakukan adalah sebagai berikut :

1. Pencarian referensi meliputi studi pustaka tentang :
 - a. Kompresi Citra
 - b. Performansi Kompresi Citra
 - c. Wavelet Transform
 - b. Metode Haar Wavelet
 - c. Metode Embedded Zerotree Wavelet (EZW)
2. Pengumpulan data
Bertujuan untuk mendapatkan data citra yang akan menjadi input dalam sistem. Data diperoleh dari laman <http://sipi.usc.edu/database/>, yaitu salah satu *database* gambar digital untuk mendukung penelitian dalam pengolahan citra, analisis citra, dan *machine vision*. Citra uji yang digunakan sebanyak 4 citra *grayscale* berformat bitmap (.bmp) dengan resolusi 256x256 dan 512x512.
3. Implementasi perangkat lunak
Sistem dirancang dengan menggunakan Matlab R2012b.
4. Analisis performansi
Melakukan analisis performansi dari metode Haar Wavelet Transform dengan EZW yang menggunakan filter Haar Wavelet dalam pengkompresian citra.
5. Membandingkan Metode
Membandingkan metode mana yang paling baik untuk melakukan kompresi citra. Hasil performansi berupa rasio kompresi (%), laju bit (bpp), MSE, PSNR dan waktu kompresi

(detik). Dari hasil parameter performansi tersebut, akan diperoleh metode mana yang terbaik dalam kompresi citra.

6. Mengambil kesimpulan

Mengambil kesimpulan setelah melakukan pengujian dengan metode Haar Wavelet Transform dengan EZW yang menggunakan filter Haar Wavelet

3. HASIL DAN DISKUSI

3.1 Penghitungan MSE, PSNR, Laju Bit, dan Compression Ratio

3.1.1 Mean Square Error (MSE)

Mean Square Error (MSE) merupakan ukuran control kualitas yang digunakan untuk mengetahui kualitas dari suatu proses. MSE menghitung seberapa besar pergeseran data antara sinyal sumber dan sinyal hasil keluar, dimana sinyal sumber dan sinyal hasil keluaran memiliki ukuran yang sama. Nilai MSE yang baik adalah mendekati 0 ($MSE \approx 0$). Perhitungan MSE dirumuskan pada Persamaan 8 :

$$MSE = \sum_{i=1}^n (S - S_e)^2 \quad (8)$$

Dimana :

MSE = Mean Square Ratio

S = Sinyal input

Se = Sinyal output

n = Panjang sinyal

3.1.2 Peak Signal to Noise Rasio (PSNR)

Parameter ukur yang digunakan untuk mengetahui gambar digital yang dihasilkan dari proses restorasi dalam penelitian ini adalah PSNR. PSNR (Peak Signal to Noise Ratio) merupakan nilai perbandingan antara nilai maksimum dari gambar hasil filtering dengan nilai rata-rata kuadrat error (MSE), yang dinyatakan dalam satuan decibel (dB).

Perhitungan PSNR dirumuskan pada Persamaan 9 :

$$PSNR = 10 \log_{10} \left(\frac{Max\ sinyal^2}{MSE} \right) \quad (9)$$

Dimana :

MSE = Mean Square Error

Max sinyal = Nilai maksimum dari gambar

3.1.3 Bit per Pixel (BPP)

Dalam satu gambar bisa mempunyai banyak macam warna. Hal ini dikarenakan jumlah pixel yang digunakan berhubungan dengan jumlah warna yang digunakan. Ini mengacu pada kedalaman bit atau kedalaman warna. Banyak bit per pixel berarti banyak pula warna yang terdapat di dalamnya.

Laju bit adalah parameter untuk membandingkan elemen-elemen tak nol dari citra terkuantisasi dengan elemen-elemen tak nol pada citra asli. Tentu saja, dengan menurunnya parameter ini, kemampuan wavelet sebagai pengkompresi citra akan semakin lebih baik. Perhitungan laju bit dirumuskan dalam Persamaan 10 :

$$Laju\ bit\ (bpp) = \frac{(jlpiks - jlpiksnol) \times 8}{jlpiks} \quad (10)$$

dimana $jlpiks$ adalah jumlah total piksel pada citra dan $jlpiksnol$ adalah jumlah piksel yang bernilai nol pada citra terkuantisasi.

3.1.4 Compression Ratio

Kompresi data adalah ukuran presentase data yang berhasil dikompres. Secara matematis persamaan kompresi rasio data, dirumuskan pada Persamaan 11 :

$$\text{Rasio kompresi (\%)} = \frac{M_0 - M_1}{M_0} \times 100 \quad (11)$$

Dimana :

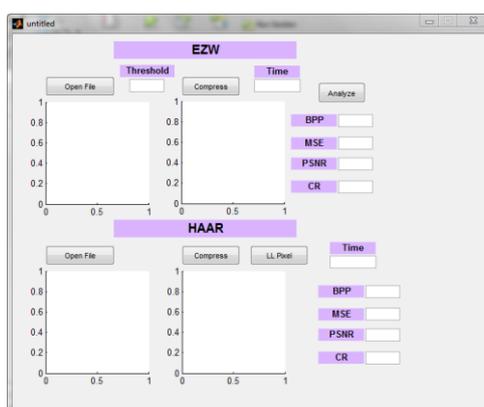
M_0 = ukuran citra asli

M_1 = ukuran citra hasil kompresi

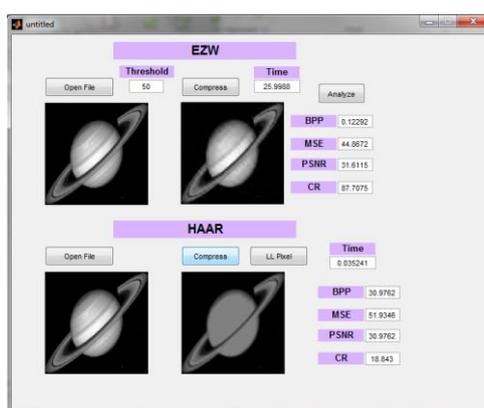
3.2 Implementasi dan Analisis Program

3.2.1 Tampilan GUI

Pada antarmuka aplikasi kompresi tersebut terdapat tombol open file yang berfungsi untuk menginputkan gambar yang akan dimasukkan untuk diproses menjadi gambar terkompresi dengan kedua teknik tersebut. Selain itu juga terdapat beberapa kotak tampilan yang akan menampilkan keluaran gambar hasil kompresi EZW dan Haar wavelet. Ada juga kotak tampilan yang akan menampilkan hasil perhitungan beberapa parameter seperti BPP, MSE, PSNR, waktu kompresi, dan rasio kompresi.



Gambar 16. Tampilan Antarmuka Program



Gambar 17 Tampilan Setelah Input Gambar

Langkah langkah untuk menjalankan program ini adalah :

1. Buka file matlab yang bernama untiled.m yang merupakan kodingan dari aplikasi tersebut.

2. Untuk kompresi EZW langkah pertama klik open file kemudian pilih gambar yang akan dikompresi. Lalu masukkan nilai threshold yang diinginkan, setelah itu klik tombol compress. Terakhir klik tombol Analyze untuk menganalisa performansi kompresi.
3. Untuk kompresi Haar Wavelet langkah pertama klik open file kemudian pilih gambar yang akan dikompresi. Lalu masukkan nilai threshold yang diinginkan, setelah itu klik tombol compress. Kemudian klik LL Pixel untuk mengambil area yang diperkirakan sebagai citra output dan menganalisa hasil kompresi.

3.3 Hasil Simulasi

Pada penelitian ini, simulasi dilakukan dengan menggunakan algoritma Haar Wavelet Transform dan algoritma Embedded Zero Transform (EZW) sebagai pembandingnya. Citra uji yang digunakan sebanyak 4 citra grayscale berformat bitmap (.bmp) dengan resolusi 256x256 dan 512x512. Performansi sistem pada simulasi aplikasi kompresi citra berupa laju bit (bpp), MSE, PSNR, citra hasil kompresi (byte), rasio kompresi (%), dan waktu kompresi (s).

3.3.1 Hasil kompresi dengan Haar Wavelet Transform

Tabel 3 menunjukkan hasil pengujian dari proses kompresi terhadap 4 citra uji, yaitu lenna.bmp, finger.bmp, fingerprint.bmp, dan city.bmp menggunakan algoritma Haar Wavelet Transform dengan berbagai variasi nilai threshold.

Tabel 3. Hasil Kompresi dengan Haar Wavelet Transform

Image (.bmp)	Pixel	Threshold	Laju bit (bpp)	MSE	PSNR	citra uji (byte)	citra hasil kompresi (byte)	rasio kompresi (%)	waktu kompresi (s)
Lenna	256x256	5	2.22	60.66	30.82	65536	26963	58.86	55.09
		10	1.35	58.19	30.88	65536	15128	76.92	41.96
		20	0.79	69.63	29.94	65536	9138	86.06	26.71
		40	0.35	134.55	26.84	65536	9685	85.22	19.55
		80	0.18	257.11	24.18	65536	15070	77.01	12.07
finger	256x256	5	2.99	60.66	27.82	65536	29505	54.98	63.59
		10	1.56	58.19	27.38	65536	13775	78.98	35.96
		20	0.82	69.63	27.94	65536	8936	86.36	20.71
		40	0.48	119.55	26.84	65536	8360	87.24	13.55
		80	0.21	197.11	24.18	65536	11166	82.96	11.07
fingerprint	512x512	5	2.86	113.88	28.15	262144	68166	74.00	1183.30
		10	1.53	114.51	27.01	262144	42184	83.91	814.75
		20	0.71	118.62	27.88	262144	20245	92.28	327.16
		40	0.29	139.40	27.01	262144	14721	94.38	118.29
		80	0.13	174.76	26.02	262144	12179	95.35	97.89
city	512x512	5	1.89	147.11	25.85	262144	71491	72.73	1241.24
		10	1.08	162.33	27.30	262144	46315	82.33	868.54
		20	0.65	197.04	29.81	262144	26061	90.06	251.38
		40	0.59	215.23	28.69	262144	19394	92.60	158.43
		80	0.19	237.11	28.53	262144	15660	94.03	97.86

Berdasarkan Tabel 3 diperoleh hasil bahwa citra uji dapat dikompresi minimal 54.98 % dengan ukuran citra hasil kompresi 29505 byte dan dapat dikompresi maksimal 95,35 % dengan ukuran citra hasil kompresi 12179 byte. Perbandingan antara nilai PSNR dan MSE dari pengujian pada tabel, 3 yaitu apabila threshold semakin kecil maka nilai PSNR akan

semakin besar dan nilai MSE akan semakin kecil. Artinya citra hasil kompresi mirip dengan citra asli, sehingga kualitas citra hasil kompresi semakin bagus. Laju bit paling rendah yaitu sekitar 0.13 bpp pada citra fingerprint.bmp dengan nilai threshold 80 dan laju bit paling tinggi yaitu sekitar 2.99 bpp pada citra finger.bmp dengan nilai threshold 5. Parameter threshold berbanding lurus dengan rasio kompresi namun berbanding terbalik dengan nilai PSNR, laju bit, dan waktu kompresi-nya.

3.3.2 Hasil kompresi dengan Embedded Zerotree Wavelet

Tabel 4 menunjukkan hasil pengujian dari proses kompresi terhadap 4 citra uji, yaitu lenna.bmp, finger.bmp, fingerprint.bmp, dan city.bmp menggunakan algoritma Embedded Zerotree wavelet menggunakan filter Haar Wavelet dengan berbagai variasi nilai threshold.

Tabel 4. Hasil Kompresi dengan Embedded Zerotree Wavelet

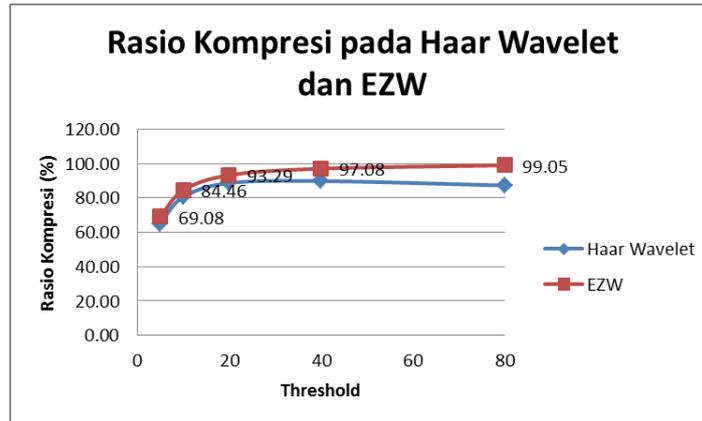
Image (.bmp)	Pixel	Threshold	Laju bit (bpp)	MSE	PSNR	citra uji (byte)	citra hasil kompresi (byte)	rasio kompresi (%)	waktu kompresi (s)
lenna	256x256	5	2.18	45.63	34.82	65536	23596	64.00	61.75
		10	1.05	48.19	33.38	65536	11897	81.85	46.01
		20	0.51	59.63	32.44	65536	5279	91.94	29.08
		40	0.27	117.05	29.34	65536	2258	96.56	20.60
		80	0.11	166.61	26.68	65536	905	98.62	12.61
finger	256x256	5	2.73	50.66	30.32	65536	26141	60.11	72.56
		10	1.60	53.19	29.88	65536	10547	83.91	40.64
		20	0.79	59.63	29.94	65536	5079	92.25	23.17
		40	0.42	109.55	29.34	65536	1758	97.32	14.99
		80	0.06	172.11	26.68	65536	302	99.54	11.40
fingerprint	512x512	5	1.89	98.88	30.65	262144	61875	76.40	1191.88
		10	1.09	104.51	29.51	262144	35860	86.32	819.34
		20	0.36	108.62	30.38	262144	13694	94.78	329.29
		40	0.25	114.40	29.51	262144	7028	97.32	119.16
		80	0.09	154.76	28.52	262144	2541	99.03	98.28
city	512x512	5	1.94	117.11	33.59	262144	63375	75.82	1246.91
		10	1.14	123.33	32.24	262144	37360	85.75	871.78
		20	0.71	175.62	31.76	262144	15194	94.20	253.33
		40	0.50	201.37	29.69	262144	7528	97.13	159.75
		80	0.13	213.44	28.76	262144	2591	99.01	98.04

Berdasarkan Tabel 4 diperoleh hasil bahwa citra uji dapat dikompresi minimal 64.00 % pada citra lenna.bmp dengan nilai threshold 5 dan ukuran citra hasil kompresi 23596 byte, kemudian dapat dikompresi maksimal 99,54 % pada citra finger.bmp dengan nilai threshold 80 dengan ukuran citra hasil kompresi 302 byte. Perbandingan antara nilai PSNR dan MSE dari pengujian pada tabel 3 yaitu apabila threshold semakin kecil maka nilai PSNR akan semakin besar dan nilai MSE akan semakin kecil. Artinya citra hasil kompresi mirip dengan citra asli, sehingga kualitas citra hasil kompresi semakin bagus. Laju bit paling rendah yaitu sekitar 0.06 bpp pada citra finger.bmp dengan nilai threshold 80 dan laju bit paling tinggi yaitu sekitar 2.73 bpp pada citra finger.bmp dengan nilai threshold 5. Parameter threshold berbanding lurus dengan rasio kompresi namun berbanding terbalik dengan nilai PSNR, laju bit, dan waktu kompresi-nya.

3.3.3 Analisis Perbandingan Algoritma Haar Wavelet dengan EZW

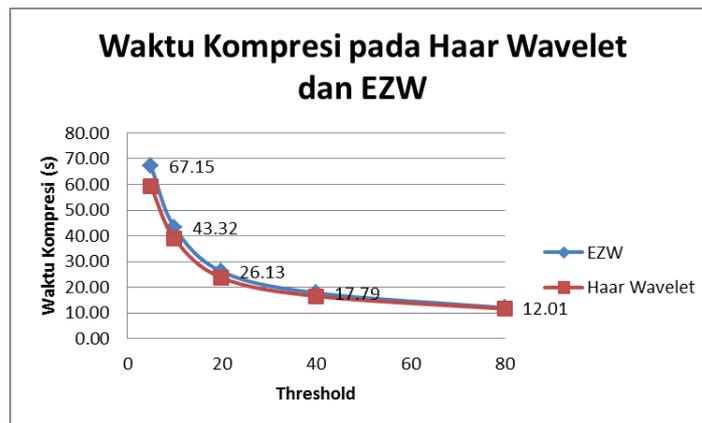
Berdasarkan pengujian algoritma Haar Wavelet dan Embedded Zerotree Wavelet pada 4 citra grayscale berformat bitmap (.bmp) dengan resolusi 256x256 dan 512x512, diperoleh hasil sebagai berikut :

1. Pengujian hasil rasio kompresi terhadap variasi nilai threshold
 Algoritma EZW memberikan hasil rasio kompresi yang lebih baik, mengalami peningkatan rata-rata 6.9% dibandingkan hanya menggunakan algoritma Haar Wavelet. Rata-rata rasio kompresi maksimal pada 4 citra uji sebesar 99.05%.



Gambar 18. Grafik Rasio Kompresi pada Algoritma Haar Wavelet dan EZW

2. Pengujian hasil waktu kompresi terhadap variasi nilai threshold
 Perbandingan waktu kompresi antara algoritma Haar Wavelet dengan EZW tidak begitu signifikan jika dibandingkan dengan rasio kompresi yang diperoleh. Selisih antara keduanya sekitar 8 detik. Algoritma Haar Wavelet memberikan waktu kompresi lebih baik dibandingkan EZW. Rata-rata waktu kompresi maksimal pada 4 citra uji sebesar 67.15 s.



Gambar 19. Grafik Waktu Kompresi pada Algoritma Haar Wavelet dan EZW

4. KESIMPULAN

Telah dilakukan penelitian untuk membandingkan performansi hasil kompresi antara algoritma Haar Wavelet dengan Embedded Zerotree Wavelet, dengan menggunakan 4 citra grayscale berformat bitmap (.bmp) dengan resolusi 256x256 dan 512x512. Adapun hasil yang diperoleh sebagai berikut :

1. Kompresi citra menggunakan algoritma Haar Wavelet diperoleh tingkat akurasi maksimum 95.35% dan laju bit minimum 0.13 pada threshold 80.
2. Kompresi citra menggunakan algoritma Embedded Zerotree Wavelet diperoleh tingkat akurasi maksimum 99.54% dan laju bit minimum 0.06 pada threshold 80.
3. Parameter threshold berbanding lurus dengan rasio kompresi namun berbanding terbalik dengan nilai PSNR, laju bit, dan waktu kompresi-nya.

DAFTAR RUJUKAN

Khalid Sayood. (2000). *Introduction to Data Compression*. Academic Press.

D. Salomon. (1998). *Data Compression: The Complete Reference*. Springer.

Y. S. B. a. S. K. Sangeeta Arora. (2014) *Haar Wavelet Transform For Solution Of Image Retrieval*, International Journal of Advanced Computer and Mathematical Sciences, vol. 5, no. 2, pp. 27-31.

J. M. Shapiro. (1993). *Embedded Image Coding Using Zerotrees of Wavelet Coefficients*, IEEE Trans. on Signal Processing, Vol. 41, No. 12, pp. 3445– 3462.

Khairil Anwar, Aris Sugiharto, dan Priyo Sidik Sasongko. (2008). *Kompresi Citra Medis Menggunakan Discrete Wavelet Transform (DWT) dan Embedded Zerotree Wevelet(EZW)*, Jurnal Matematika Vol. 11, No.2, pp. 73-77.

Rismon H Sianipar, Sri Muliani WJ. (2003). *Kompresi Citra Digital Berbasis Wavelet: Tinjauan PSNR dan Laju Bit*, Jurnal Informatika Vol. 4, No. 2, pp. 81 – 87.

Gonzales RC, Woods RE. 2002. *Digital Image Processing*. Ed ke-2. New Jersey: Prentice Hall.

Jones CL, Lonergan GT, Mainwaring DE. (1999). *Wavelet packet computation of the hurst exponent*. Journal of Physics A: Mathematical and General 29:2509- 2527.

Misiti M, Misiti Y, Oppenheim G, Poggi JM. (2002). *Wavelet Toolbox*. USA: The Mathwork, Inc.

Tan CL. (2001). *Still image compression using wavelet transform*, [Tesis]. Australia: The University of Queensland.

Anwar K, Sugiharto A, Sidik P, Sasongko. (2008). *Kompresi citra medis menggunakan discrete wavelet transform (DWT) dan embedded zerotree wavelet (EZW)*. Jurnal Matematika 11:73-77.

Mitra et al. _____. (2000). *Modified Embedded Zerotree Scheme for Efficient Coding of Discrete Wavelet Coded Frames*, Journal, Kharagpur: Department of Electronics and Electrical Communication Engineering IIT.

Subanar dan Suhartono. (2009). *Wavelet Neural Networks untuk Peramalan Data Time Series Finansial*, Laporan Penelitian Dasar Perguruan Tinggi, FMIPA, Universitas Gajah Mada, Yogyakarta.