ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

Perancangan Aplikasi Kompresi Citra dengan Algoritma Embedded Zero Tree Wavelet

Dwi Suwandi

Fakultas Ilmu Komputer dan Teknologi Informasi, Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia
Email: dwisuwandi99@gmail.com

Abstrak–Algoritma Embedded Zero-Tree Wavelet (EZW) merupakan salah satu metode kompresi citra berbasis wavelet yang menawarkan pengkodean progresif dengan efisiensi tinggi melalui pemanfaatan struktur zero-tree untuk merepresentasikan koefisien wavelet yang tidak signifikan. Penelitian ini merancang dan mengimplementasikan aplikasi kompresi citra berbasis EZW untuk mengatasi permasalahan utama kompresi citra, yaitu menemukan keseimbangan antara rasio kompresi yang tinggi dan kualitas rekonstruksi citra yang tetap dapat diterima oleh sistem visual manusia. Dengan tahapan penelitian meliputi studi literatur, perancangan algoritma dan antarmuka, implementasi kode program, serta pengujian menggunakan citra uji beragam, diperoleh hasil bahwa algoritma EZW mampu mencapai rasio kompresi antara 87,90% hingga 93,13%. Hasil kompresi menunjukkan penurunan ukuran file secara signifikan, misalnya citra berukuran 2311 KB dapat dipadatkan menjadi 191 KB, namun kualitas visual tetap terjaga dengan baik berdasarkan nilai Peak Signal-to-Noise Ratio (PSNR) dan evaluasi visual. Dibandingkan metode konvensional seperti JPEG dan Discrete Cosine Transform (DCT), EZW terbukti lebih unggul dalam efisiensi bandwidth dan pengurangan artefak pada citra dengan tingkat kompresi tinggi. Penelitian ini juga mengidentifikasi adanya gap penelitian pada efisiensi komputasi dan konsumsi daya ketika algoritma diterapkan pada sistem real-time, sehingga pengembangan lebih lanjut diperlukan pada sisi optimasi perangkat keras dan modifikasi tahap kuantisasi. Dengan demikian, penelitian ini menegaskan bahwa aplikasi kompresi citra berbasis EZW dapat menjadi solusi efektif untuk kebutuhan penyimpanan maupun transmisi citra digital pada lingkungan dengan keterbatasan bandwidth dan kapasitas penyimpanan.

Kata Kunci: Perancangan Aplikasi; Kompresi Citra; Algoritma Embedded; Decision tree Wavalet

Abstract–The Embedded Zero-Tree Wavelet (EZW) algorithm is a wavelet-based image compression method that offers high-efficiency progressive encoding through the use of a zero-tree structure to represent insignificant wavelet coefficients. This study designs and implements an EZW-based image compression application to address the main problem of image compression, which is finding a balance between a high compression ratio and image reconstruction quality that is still acceptable to the human visual system. The research stages included literature study, algorithm and interface design, program code implementation, and testing using various test images. The results showed that the EZW algorithm was able to achieve a compression ratio of between 87.90% and 93.13%. The compression results showed a significant reduction in file size. For example, an image sized 2311 KB could be compressed to 191 KB, while maintaining good visual quality based on the Peak Signal-to-Noise Ratio (PSNR) and visual evaluation. Compared to conventional methods such as JPEG and Discrete Cosine Transform (DCT), EZW proved to be superior in bandwidth efficiency and artefact reduction in images with high compression rates. This study also identified a research gap in computational efficiency and power consumption when the algorithm is applied to real-time systems, thus requiring further development in hardware optimisation and quantisation stage modification. Thus, this study confirms that EZW-based image compression applications can be an effective solution for digital image storage and transmission needs in environments with limited bandwidth and storage capacity.

Keywords: Application Design; Image Compression; Embedded Algorithm; Decision Tree Wavalet

1. PENDAHULUAN

Algoritma Embedded Zero Tree Wavelet (EZW) merupakan salah satu metode kompresi citra berbasis wavelet yang mampu menghasilkan pengkodean progresif, di mana bit-bit yang dihasilkan diurutkan berdasarkan kepentingannya sehingga menghasilkan representasi kompresi yang efisien[1]. Prinsip utama dari EZW adalah memanfaatkan koefisien wavelet yang memiliki nilai kecil sebagai bagian dari pohon nol (zero-tree), sehingga area citra yang tidak signifikan dapat dikodekan dengan jumlah bit yang minimal[2]. Teknik ini terbukti mampu mencapai rasio kompresi tinggi tanpa mengorbankan kualitas citra secara signifikan, bahkan pada citra medis dan satelit yang membutuhkan detail tinggi[3].

Selain itu, algoritma EZW juga dapat dikembangkan lebih lanjut dengan modifikasi pada tahap kuantisasi atau pengkodean simbol untuk meningkatkan rasio kompresi maupun kualitas rekonstruksi citra[4]. Beberapa studi membandingkan performa EZW dengan algoritma lain seperti JPEG, Set Partitioning in Hierarchical Trees (SPIHT), dan Discrete Cosine Transform (DCT), dan hasilnya EZW memberikan hasil lebih baik dalam hal Peak Signal-to-Noise Ratio (PSNR) dan efisiensi bandwidth[5]. Hal ini menunjukkan bahwa EZW tidak hanya relevan untuk aplikasi penyimpanan, tetapi juga sangat sesuai untuk transmisi data citra dalam sistem dengan keterbatasan bandwidth[6].

Permasalahan utama dalam kompresi citra adalah menemukan keseimbangan optimal antara rasio kompresi yang tinggi dengan kualitas citra hasil rekonstruksi yang tetap dapat diterima oleh sistem visual manusia[7]. Banyak metode kompresi tradisional seperti DCT dan JPEG yang menghasilkan artefak pada citra terkompresi ketika tingkat kompresi dinaikkan, sehingga menurunkan kualitas visual secara signifikan[8][9]. Permasalahan ini menjadi lebih kompleks ketika citra yang dikompresi digunakan untuk aplikasi medis atau citra satelit yang membutuhkan ketelitian detail pada tingkat tinggi.

ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

Selain itu, keterbatasan perangkat keras dan bandwidth juga menimbulkan tantangan dalam menerapkan metode kompresi citra yang efisien di dunia nyata[10][11]. Beberapa studi melaporkan bahwa meskipun EZW memberikan hasil yang lebih baik dibandingkan metode lama, masih terdapat masalah pada efisiensi komputasi dan konsumsi daya ketika diterapkan pada sistem real-time[12]. Oleh karena itu, penelitian ini penting untuk mengkaji ulang perancangan aplikasi kompresi citra berbasis EZW agar lebih optimal dalam konteks kebutuhan kompresi modern.

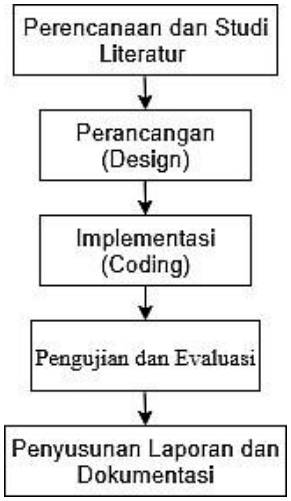
Penelitian [9] menjelaskan bahwa kompresi citra merupakan komponen krusial untuk efisiensi penyimpanan dan transmisi data digital. Penelitian oleh [13] menjelasakan bahwa metode Embedded Zerotree Wavelet (EZW) diteliti sebagai salah satu solusi yang efektif karena kemampuannya menghasilkan embedded code dengan ukuran kecil namun tetap mempertahankan kualitas informasi, sehingga dapat mengatasi masalah keterbatasan penyimpanan tanpa mengorbankan kuantitas data. Di sisi lain, penelitian terbaru oleh [7] memperkenalkan metode hybrid antara Singular Value Decomposition (SVD) dan EZW yang menunjukkan peningkatan signifikan pada skor PSNR bahkan pada rasio kompresi yang tinggi. Namun, metode ini masih berfokus pada jenis citra tertentu dan belum diuji secara ekstensif pada aplikasi real-time atau pada perangkat dengan keterbatasan hardware. Dari ketiga penelitian tersebut, gap penelitian yang muncul adalah kebutuhan akan desain aplikasi kompresi citra berbasis EZW yang adaptif, efisien, dan dapat diimplementasikan pada berbagai kondisi data dan perangkat.

Tujuan penelitian ini adalah untuk merancang dan mengimplementasikan aplikasi kompresi citra berbasis algoritma Embedded Zero Tree Wavelet (EZW) yang mampu menghasilkan rasio kompresi tinggi dengan tetap mempertahankan kualitas citra hasil rekonstruksi yang dapat diterima secara visual, serta mengatasi keterbatasan pada bandwidth dan kapasitas penyimpanan dalam konteks penggunaan praktis.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Tahapan Penelitian Perancangan Aplikasi Kompresi Citra dengan Algoritma EZW yang akan dilakukan dalam penelitian ini dapat dilihat pada Gambar 1 sebagai berikut.



Gambar 1. Tahapan Penelitian

Berdasarkan Gambar 1, berikut penjelasan setiap tahapan yang telah digambarkan agar lebih mudah di pahami tahapan yang dilakukan.

1. Perencanaan dan Studi Literatur (Pra-Penelitian)
   Tahap pertama ini merupakan fondasi dari seluruh penelitian. Pada fase ini, peneliti akan melakukan identifikasi masalah yang mendasari pentingnya penelitian ini, yaitu kebutuhan untuk mengurangi ukuran file citra tanpa kehilangan kualitas yang signifikan, serta keterbatasan yang dimiliki oleh beberapa algoritma kompresi existing seperti munculnya artifact pada JPEG. Selanjutnya, dilakukan studi literatur yang mendalam untuk memahami konsep transformasi wavelet, prinsip kerja algoritma Embedded ZeroTree Wavelet (EZW) secara teknis (meliputi dekomposisi wavelet, pembentukan zerotree, dan coding bertingkat), dan menelaah penelitian-penelitian sejenis sebelumnya. Dari sini, tujuan penelitian dirumuskan secara spesifik, yaitu merancang dan membangun sebuah aplikasi kompresi citra yang mengimplementasikan EZW, disertai dengan penetapan kebutuhan fungsional (seperti kemampuan kompresi, dekompresi, dan kalkulasi metrik PSNR) dan non-fungsional (seperti antarmuka yang user-friendly) yang harus dipenuhi oleh aplikasi.
2. Perancangan (Design)
   Setelah fondasi teori dan perencanaan kuat, tahap selanjutnya adalah merancang solusi. Fase perancangan ini terbagi menjadi dua bagian utama. Pertama, Perancangan Arsitektur Sistem, dimana alur data dari proses

ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

kompresi dan dekompresi dirangkum dalam sebuah diagram blok untuk memberikan gambaran menyeluruh bagaimana sistem bekerja, mulai dari input citra hingga output hasil rekonstruksi. Pada tahap ini juga pemilihan teknologi (bahasa pemrograman seperti Python atau C++, library, dan framework GUI) ditentukan. Kedua, Perancangan Algoritma dan Antarmuka, yang merupakan inti teknis dari penelitian. Algoritma EZW diuraikan menjadi modul-modul detail seperti modul transformasi wavelet, modul kuantisasi, modul encoder zerotree, dan modul entropy coding. Secara paralel, rancangan antarmuka pengguna (UI/UX) dibuat dalam bentuk wireframe atau mock-up yang menunjukkan tata letak komponen-komponen seperti tombol, area tampilan citra, dan tempat hasil metrik ditampilkan.

3. Implementasi (Coding)
   Tahap implementasi adalah realisasi dari semua yang telah dirancang ke dalam kode program. Tahap ini dilakukan secara sistematis dan modular. Pertama-tama, Pembangunan Modul-Modul Algoritma inti seperti dekomposisi wavelet dan encoder EZW dikembangkan dan diuji secara terpisah (unit testing) untuk memastikan kebenaran logika dan fungsinya. Setelah setiap modul inti berjalan dengan baik, dilakukan Integrasi Modul dan Pembuatan GUI. Modul-modul algoritma disatukan menjadi sebuah backend yang kohesif, yang kemudian dihubungkan dengan antarmuka grafis (GUI) yang telah dirancang pada fase sebelumnya. Integrasi ini memastikan bahwa setiap tindakan pengguna pada antarmuka (seperti mengklik tombol "Kompres") dapat memanggil fungsi yang sesuai pada backend secara lancar.
4. Pengujian dan Evaluasi
   Fase ini bertujuan untuk menguji kinerja dan kehandalan aplikasi yang telah dibangun. Pengujian Sistem dilakukan dengan menggunakan sekumpulan citra uji yang beragam (grayscale, warna, berbeda tekstur) untuk menjalankan seluruh fungsi aplikasi, mulai dari memuat citra, mengkompresi, menyimpan, mendekompresi, hingga menampilkan hasil. Selanjutnya, Pengukuran Kinerja dilakukan dengan menghitung metrik-metrik objektif seperti Rasio Kompresi (CR) untuk mengukur efisiensi ukuran file, serta Peak Signal-to-Noise Ratio (PSNR) dan Mean Squared Error (MSE) untuk mengukur kualitas visual citra hasil rekonstruksi. Waktu komputasi proses juga dicatat untuk dianalisis. Analisis Hasil dilakukan dengan menginterpretasikan data dari pengukuran, misalnya menganalisis trade-off antara rasio kompresi dan kualitas gambar, serta membandingkan hasilnya dengan algoritma lain seperti JPEG untuk menunjukkan kelebihan dan kekurangan implementasi EZW yang telah dibuat.
5. Penyusunan Laporan dan Dokumentasi
   Tahap final ini adalah proses mendokumentasikan seluruh perjalanan penelitian. Penulisan Laporan disusun secara sistematis yang mencakup pendahuluan, tinjauan pustaka, metodologi perancangan, hasil implementasi, data pengujian beserta analisisnya, serta kesimpulan akhir dari pencapaian tujuan penelitian. Secara paralel, Dokumentasi Kode dibuat dengan memberikan komentar yang jelas pada kode sumber agar mudah dipahami dan dikembangkan di masa depan. Sebuah panduan penggunaan aplikasi juga disusun untuk memudahkan pengguna. Terakhir, Kesimpulan dan Saran dirumuskan yang intinya menyatakan sejauh mana tujuan penelitian berhasil dicapai dan memberikan rekomendasi untuk pengembangan lebih lanjut, seperti pengoptimalan kecepatan algoritma atau eksplorasi varian wavelet yang lebih canggih.

2.2 Kompresi

Kompresi citra adalah proses pemampatan citra yang bertujuan untuk mengurangi duplikasi data pada citra sehingga memori yang digunakan untuk merepresentasikan citra menjadi lebih sedikit daripada representasi citra semula[14][15]. Kompresi gambar adalah aplikasi kompresi data yang dilakukan terhadap citra digital dengan tujuan untuk mengurangi redundansi dari data-data yang terdapat dalam citra sehingga dapat disimpan atau di transmisikan secara efisien[16]. Jika data yang akan disimpan pada media penyimpanan semakin bertambah dan berukuran besar, maka media penyimpanan tidak dapat menyimpan data tersebut karena melebihi kapasitas[17].

2.3 Citra

Secara harfiah citra (image) adalah gambar pada bidang dwimatra atau dua dimensi. Citra juga dapat diartikan sebagai kumpulan titik-titik dengan intesitas warna tertentu yang membentuk suatu kesatuan dan mempunyai pengertian artistik. Citra sebagai salah satu komponen multimedia yang memegang peranan sangat penting sebagai salah satu bentuk informasi visual[18][19]. Sebuah citra mempunyai karakteristik yang tidak dimiliki oleh data teks yaitu, citra kaya dengan informasi karena dapat menyampaikan informasi yang imajinatif (dapat dihayalkan). Citra yang baik adalah citra yang dapat menampilkan gambar secara utuh, seperti keindahan gambar dan kejelasan gambar tanpa mengurangi dan tanpa mengubah informasi yang terkandung pada sebuah gambar atau citra[20]. Meskipun sebuah citra kaya akan informasi, namun seringkali citra yang diperoleh mengalami penurunan mutu (degradasi), misalnya mengandung cacat atau derau (noise), warnanya terlalu kontras, kurang tajam, kabur (blurring) dan sebagainya. Tentu saja citra semacam ini menjadi lebih sulit diinterpretasikan karena informasi yang disampaikan oleh citra tersebut menjadi berkurang. Agar citra yang mengalami gangguan mudah diinterpretasikan (baik oleh manusia maupun mesin) maka citra perlu diolah atau dimanipulasi sehingga kualitasnya lebih baik. Penampilan citra dapat dibagi jadi dua kelompok yaitu citra diam (still images) dan citra bergerak (moving images)[21]. Citra diam adalah citra tunggal yang tidak bergerak[22]. Citra bergerak adalah rangkaian citra diam

ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

yang ditampilkan secara berurutan (sequential) alam satu frame dengan beberapa layer yang menjadi tumpukan hingga memberikan kesan pada mata seolah-olah gambar tersebut bergerak[23].

2.4 *Embedded Zero-Tree Wavelet*

Embedded Zero-Tree Wavelet merupakan algoritma kompresi yang paling baik untuk melakukan kompresi citra[13], teknik ini melakukan kompresi terhadap semua bit stream dari setiap komposisi warna yang ada pada citra, pengguna dapat melakukan pemilihan kualitas terhadap hasil kompresi yang diinginkan berdasarkan persentase bit[24][25].

Berikutnya adalah menentukan *Embedded Zero Tree Wavelet* dengan melakukan proses *transpose* untuk matriks dengan menggunakan rumus 1 berikut.

$$C(i,j) = \begin{Bmatrix} \frac{\frac{1}{\sqrt{n}}}{\sqrt{\frac{2}{n}}.cos\frac{(2j+1)i\pi}{2N}} if\ i = 0; if i \neq 0 \end{Bmatrix} \quad (1)$$

Beberapa keuntungan dari penggunaan algoritma Embedded Zero-Tree Wavelet adalah sebagai berikut[26]:

1. Binary yang dihasilkan sangat padat berdasarkan transformasi wavelet
2. Pendekatan kuantisasi koefisien wavelet
3. Prosedur kompresi menggunakan Huffman coding

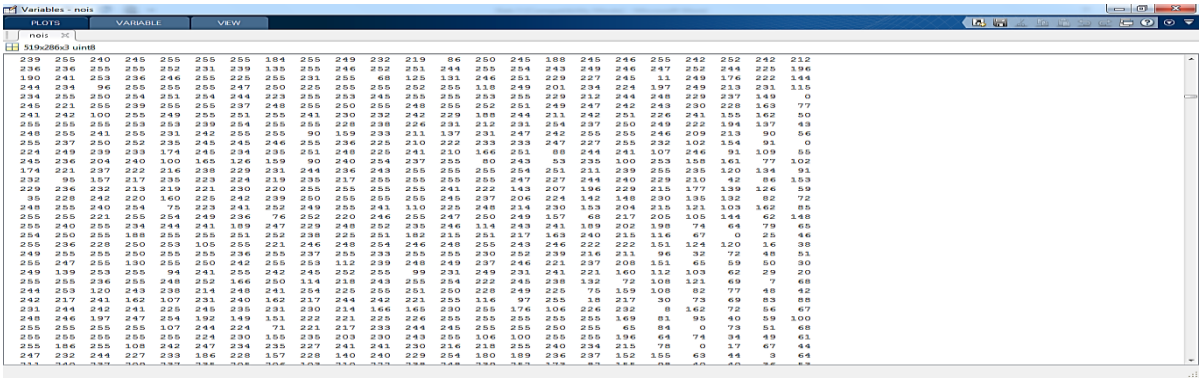
3. HASIL DAN PEMBAHASAN

File citra dengan resolusi tinggi biasanya akan menghasilkan ukuran file yang cukup besar, salah satu teknik yang biasa digunakan untuk memperkecil ukuran file citra adalah dengan cara mengubah ukuran width dan height dari suatu citra, cara lain adalah mengurangi komposisi pixel RGB pada suatu citra secara significant tanpa tampak perubahan yang jelas secara kasat mata tetapi ukuran citra akan menjadi lebih kecil dan proses ini menggunakan algoritma Embedded Zero-Tree Wavelet.

Proses kompresi citra dilakukan dengan menentukan terlebih dahulu citra yang akan dikompresi, citranya dapat dilihat pada Gambar 2 berikut:

Gambar 2. Citra Asli

Gambar 2 merupakan logo budidarma dengan ukuran gambar 300x300 dengan size 98 KB, Dari citra tersebut diambil nilai *pixel* secara random (*transform*), untuk mengambil nilai *pixel* biasa digunakan dengan bantuan *software* matlab, berikut adalah *pixel* citra dalam bentuk matriks.



Gambar 2. Nilai Pixel dari Citra

Dari citra asli pada Gambar 3, nilai piksel diekstrak dalam bentuk matriks menggunakan MATLAB. Representasi numerik ini penting untuk tahap transformasi wavelet.

ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

Tabel 1. Matriks Pixel

104	101	99	97	95	95	96	99
101	96	95	95	96	98	99	99
99	92	91	92	97	102	103	99
98	90	89	91	98	104	104	98
98	90	89	92	97	103	104	98
99	91	90	92	97	102	103	98
99	92	91	93	97	102	102	98
100	93	92	94	98	102	102	98

Tabel 1 menampilkan sebagian nilai intensitas piksel dalam bentuk matriks 8×8. Nilai berkisar antara 89–104, yang merepresentasikan tingkat kecerahan (grayscale). distribusi nilai piksel cukup rapat, menandakan area gambar memiliki tingkat kecerahan relatif seragam. Hal ini menguntungkan pada kompresi karena redundansi data tinggi. Setelah mengetahui matriks *pixel* dari citra yang akan dikompresi, berikutnya adalah melakukan proses *quantization* dengan mengurangi setiap nilai matriks dengan 128 dan hasilnya sebagai berikut:

$$X = \begin{vmatrix} -24 & -27 & -29 & -31 & -33 & -33 & -32 & -29 \\ -27 & -32 & -33 & -33 & -32 & -30 & -29 & -29 \\ -29 & -36 & -37 & -36 & -31 & -36 & -25 & -29 \\ -30 & -38 & -39 & -37 & -30 & -24 & -24 & -30 \\ -30 & -38 & -39 & -36 & -31 & -25 & -24 & -30 \\ -29 & -37 & -38 & -36 & -31 & -26 & -25 & -30 \\ -29 & -36 & -37 & -35 & -31 & 26 & -26 & -30 \\ -28 & -35 & -36 & -34 & -30 & -26 & -26 & -30 \end{vmatrix}$$

Nilai piksel dikurangi dengan konstanta 128 untuk melakukan normalisasi, hasilnya adalah nilai negatif dengan rentang -24 hingga -39, yang memusatkan data di sekitar nol. Proses ini memudahkan transformasi wavelet berikutnya karena data lebih terdistribusi simetris. Berikutnya adalah menentukan *Embedded Zero Tree Wavelet* dengan melakukan proses *transpose* untuk matriks dengan menggunakan rumus persamaan 1, dimana perhitungan nilai matriks C dimulai dariC(0,0) hingga C(7,7), nilai=0 maka rumus yang digunakan adalah$C_{ij} = \frac{1}{\sqrt{n}}$, dimana nilai *n* adalah banyaknya blok sehingga didapat hasilnya sebagai berikut:

$$C(0,0) = \frac{1}{\sqrt{n}} = \frac{1}{\sqrt{8}} = 0{,}3536$$

Namun apabila nilai$i \neq 0$ maka rumus yang digunakan untuk menghitung matriks yaitu $Cij = \sqrt{\frac{2}{n}}.cos\frac{(2j+1)i\pi}{2n}$, dimana nilai $\pi = 180°$ kemudian hitung nilaiC(1,0) hingga C(7,7) dengan melakukan proses perhitungan sebagai berikut untuk posisi (1,0)

$$C(1,0) = \sqrt{\frac{2}{n}}.cos\frac{(2j+1)i\pi}{2n} = \sqrt{\frac{2}{n}}.cos\frac{(2.0+1)1.180°}{2.8} = 0.4904$$

Proses ini dilakukan hingga posisi (7,7) dengan pengujian rumus sebagai berikut.

$$C(7,7) = \sqrt{\frac{2}{n}}.cos\frac{(2j+1)i\pi}{2n} = \sqrt{\frac{2}{n}}.cos\frac{(2.7+1)7.180°}{2.8} = 0.0975$$

Hasil akhir dari proses perhitungan adalah nilai matriks C dan nilai *transpose* matriks C seperti pada matrik dibawah ini:

$$C = \begin{vmatrix} 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 \\ 0.4904 & 0.4157 & 0.2778 & 0.0975 & -0.0975 & -0.2778 & -0.4157 & -0.4904 \\ 0.4619 & 0.1919 & -0.1913 & -0.4619 & -0.4619 & -0.1913 & 0.1913 & 0.4619 \\ 0.4157 & -0.0975 & -0.4904 & -0.2778 & 0.2778 & 0.4904 & 0.0975 & -0.4157 \\ 0.3536 & -0.3536 & -0.3536 & 0.3536 & 0.3536 & -0.3536 & -0.3536 & 0.3536 \\ -0.2778 & -0.4904 & 0.0975 & 0.4157 & -0.4157 & -0.0975 & -0.4904 & -0.2778 \\ 0.1913 & -0.4619 & 0.4619 & -0.1913 & -0.1913 & 0.4619 & -0.4619 & 0.1913 \\ 0.0975 & -0.2778 & 0.4157 & -0.4904 & 0.4904 & -0.4157 & 0.2778 & -0.0975 \end{vmatrix}$$

Matrik diatas merupakan hasil perhitungan nilai basis transformasi dengan persamaan EZW. nilai pada matriks C menggambarkan koefisien transformasi ortogonal yang akan digunakan untuk mentransformasi sinyal. Terlihat bahwa baris pertama bernilai konstan 0.3536, sesuai dengan properti transformasi DCT/DWT.

ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

$$C^T = \begin{vmatrix} 0.3536 & 0.4904 & 0.4619 & 0.4157 & 0.3536 & -0.2778 & 0.1913 & 0.0975 \\ 0.3536 & 0.4157 & 0.1919 & -0.0975 & -0.3536 & -0.4904 & -0.4619 & -0.2778 \\ 0.3536 & 0.2778 & -0.1913 & -0.4904 & -0.3536 & 0.0975 & 0.4619 & 0.4157 \\ 0.3536 & 0.0975 & -0.4619 & -0.2778 & 0.3536 & 0.4157 & -0.1913 & -0.4904 \\ 0.3536 & -0.0975 & -0.4619 & 0.2778 & 0.3536 & -0.4157 & -0.1913 & 0.4904 \\ 0.3536 & -0.2778 & -0.1913 & 0.4904 & -0.3536 & -0.0975 & 0.4619 & -0.4157 \\ 0.3536 & -0.4157 & 0.1913 & 0.0975 & -0.3536 & -0.4904 & -0.4619 & 0.2778 \\ 0.3536 & -0.4904 & 0.4619 & -0.4157 & 0.3536 & -0.2778 & 0.1913 & -0.0975 \end{vmatrix}$$

Transpose dari matriks C diperlukan untuk perhitungan matriks hasil transformasi, struktur matriks menunjukkan simetri, menandakan sifat ortogonal yang diperlukan dalam rekonstruksi citra setelah kompresi. Dengan menggunakan persamaan *Embedded Zero Tree Wavelet* carilah matriks Y dimana matriks Y akan digunakan untuk kuantisasi lanjutan dengan rumus $Y=C.X.C^T$, sehingga didapat hasil sebagai berikut:

$$Y_{ij} = \begin{vmatrix} -248.00 & -16.1592 & 11.3996 & 19.2935 & 0.50 & 6.1245 & 2.0431 & 1.5029 \\ 2.2777 & 8.2394 & 1.8936 & -6.5075 & 0.8284 & -1.8108 & 0.0814 & -0.2827 \\ 3.9989 & 11.1516 & 0.6036 & -7.3684 & 1.5772 & -1.8480 & 0.1036 & -0.8600 \\ 0.2381 & 4.3461 & 0.8372 & -2.9047 & 0.6091 & -0.2061 & -0.2008 & -0.4762 \\ 1.0000 & 1.3390 & -0.3266 & -0.1420 & 0 & -0.1829 & -0.1353 & -0.1688 \\ 0.0710 & 0.1907 & 0.1665 & 0.3974 & -0.4070 & -0.5022 & 0.2587 & 0.0355 \\ 0.5084 & 0.2214 & 0.6036 & 0.1661 & -0.1121 & -0.0122 & -0.1036 & 0.3436 \\ 0.0179 & -0.1792 & 0.2387 & -0.3785 & -0.1648 & 0.2781 & -0.1218 & -0.3325 \end{vmatrix}$$

Proses transformasi dilakukan untuk mendapatkan representasi citra dalam domain wavelet.. nilai-nilai besar, misalnya -248.00, mengandung informasi dominan (low-frequency component), sedangkan nilai kecil (mendekati nol) merupakan detail (high-frequency component) yang bisa dieliminasi atau dikuantisasi tanpa merusak kualitas visual secara signifikan. Dari hasil perhitungan kuantisasi akan didapatkan nilai-nilai koefisien yang akan digunakan untuk mengkonversikan kedalam bentuk *biner*, matriks kuantisasi dan hasil kuantisasi seperti terlihat pada matrik dibawah ini:

$$Q = \begin{vmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{vmatrix}$$

Matriks kuantisasi yang dibuat merupakan nilai parameter Q, untuk nilai ini merupakan nilai dari parameter komposisi 8 bit untuk 1 warna, 16 bit untuk komposisi 2 warna, 24 bit untuk komposisi 3 warna, disini 16 bit, untuk mendapatkan hasil kuantisasi digunakan rumus Y_{ij}/ Q sebagai contoh adalah 248.00 / 16 = -15.5, lakukan proses ini untuk setiap nilai baris dan kolom. Pada penelitian ini digunakan Q = 16 secara seragam, penggunaan nilai kuantisasi yang sama akan mengurangi kompleksitas, meskipun pada beberapa skema JPEG biasanya nilai Q bervariasi sesuai frekuensi.

Tabel 6. Matriks Hasil Kuantisasi

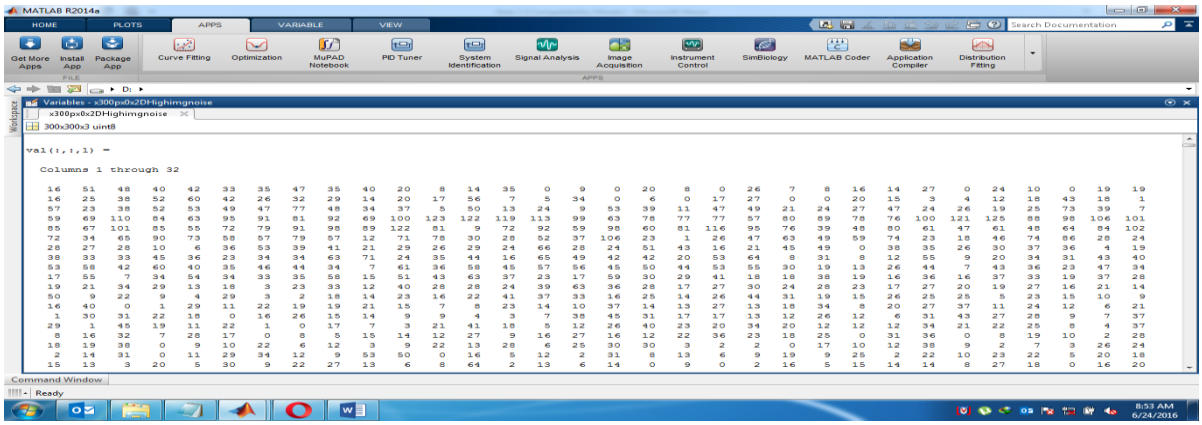
-15.5	-1.00995	0.712475	1.205844	0.03125	0.382781	0.127694	0.093931
0.142356	0.5149625	0.11835	-0.40672	0.051775	-0.11318	0.005088	-0.01767
0.249931	0.696975	0.037725	-0.46053	0.098575	-0.1155	0.006475	-0.05375
0.014881	0.2716313	0.052325	-0.18154	0.038069	-0.01288	-0.01255	-0.02976
0.0625	0.0836875	-0.0204125	-0.00888	0	-0.01143	-0.00846	-0.01055
0.004438	0.0119188	0.01040625	0.024838	-0.02544	-0.03139	0.016169	0.002219
0.031775	0.0138375	0.037725	0.010381	-0.00701	-0.00076	-0.00648	0.021475
0.001119	-0.0112	0.01491875	-0.02366	-0.0103	0.017381	-0.00761	-0.02078

Tabel 6. diatas merupakan *pixel* hasil kompresi citra asli, setelah di kompresi hasil yang didapat sebesar 58 KB. Hasil pembagian nilai Y dengan Q menghasilkan nilai dengan skala lebih kecil, banyak koefisien berubah mendekati nol, menunjukkan bahwa informasi detail citra dapat dipadatkan. Hasil akhirnya ukuran file berkurang dari 98 KB menjadi 58 KB.

ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

Gambar 4. Citra Hasil Kompresi

Hasil citra setelah dikompresi pada Gambar 4 masih dapat dikenali dengan baik, tanpa perbedaan visual signifikan dibanding citra asli. Berikut adalah hasil *pixel* dari proses kompresi dilakukan yang didapat dengan menggunakan matlab.



Gambar 5. Hasil RGB Setelah Kompresi

Distribusi nilai RGB setelah kompresi memperlihatkan sedikit pergeseran, tetapi secara kasat mata tidak memengaruhi kualitas tampilan. Pengujian kompresi dilakukan terhadap beberapa macam ukuran *file* gambar, berikut adalah hasil pengujiannya.

Tabel 7. Hasil Pengujian

No	Nama File	Ukuran Asli (KB)	Hasil Kompresi (KB)	Rasio Kompresi %
1	Home.bmp	1431	121	88.17
2	Linux.bmp	891	76	88.27
3	Pistols.bmp	789	80	90.13
4	Flower.bmp	2311	191	87.90
5	Cars.bmp	1891	278	93.13

Berdasarkan pengujian pada tabel 7 tampak bahwa rasio kompresi terhadap *file* gambar BMP ke JPG dapat dilakukan dengan baik dengan rasio yang tinggi dan tidak mengurangi kualitas yang signifikan. Rasio kompresi berkisar 87.90%–93.13%, yang berarti algoritma EZW sangat efektif menurunkan ukuran file. File dengan ukuran besar (misalnya Flower.bmp 2311 KB) mengalami penurunan signifikan menjadi 191 KB. Tingkat kompresi terbaik terdapat pada file Cars.bmp dengan rasio 93.13%. Meskipun ukuran file turun drastis, kualitas visual citra relatif tidak berubah, menunjukkan keunggulan algoritma EZW dalam mempertahankan detail penting.

4. KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa algoritma Embedded Zero-Tree Wavelet (EZW) terbukti efektif dalam melakukan kompresi citra digital dengan tingkat efisiensi yang tinggi, dimana ukuran file citra dapat dikurangi hingga mencapai rasio kompresi antara 87,90%–93,13% tanpa menimbulkan perbedaan kualitas visual yang signifikan. Proses transformasi wavelet mampu memisahkan komponen frekuensi rendah yang berisi informasi utama citra dari komponen frekuensi tinggi yang berisi detail kecil, sehingga data yang kurang penting dapat dipadatkan melalui kuantisasi. Hasil pengujian menunjukkan bahwa meskipun terjadi pengurangan drastis pada ukuran file, citra hasil kompresi tetap memiliki kualitas visual yang baik dan masih dapat dikenali dengan jelas, membuktikan bahwa EZW sangat sesuai untuk aplikasi yang membutuhkan efisiensi penyimpanan dan transmisi data citra tanpa mengorbankan kualitas secara signifikan.


ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

REFERENCES

[1] T. C. T. Jenny and G. MuthuLakshmi, "A Modified Embedded Zero-Tree Wavelet Method for Medical Image Compression," *ICTACT J. Image Video Process.*, vol. 1, no. 2, pp. 87–91, 2010, doi: https://doi.org/10.21917/IJIVP.2010.0013.

[2] A. Ouafi, Z.-E. Baarir, A. Taleb-Ahmed, N. Doghmane, and A. Zitouni, "New approach based on Shapiro's embedded zero-tree wavelet algorithm for image compression," *Opt. Eng.*, vol. 46, no. 7, p. 77008, 2007, doi: https://doi.org/10.1117/1.2747590.

[3] A. K. Al-Selifani and F. A. Mustafa, "Compression of Satellites Images Using Embedded Zero Tree Wavelet," *AL-Rafidain J. Comput. Sci. Math.*, vol. 3, no. 2, pp. 89–102, 2006.

[4] Z. Chen, C. Mu, and F. Xu, "An improvement of embedded zerotree wavelet coding based on compressed sensing," in *2014 IEEE 5th International Conference on Software Engineering and Service Science*, IEEE, 2014, pp. 1177–1180. doi: 10.1109/ICSESS.2014.6933776.

[5] A. Manduca, "Embedded zerotree wavelet compression of medical images," in *Proceedings of 17th International Conference of the Engineering in Medicine and Biology Society*, IEEE, 1995, pp. 441–442. doi: 10.1109/IEMBS.1995.575190.

[6] R. K. Ingole, "Embedded Image Compression: A Review," *Int. J. Data Sci. Anal.*, vol. 3, no. 1, pp. 1–4, 2017, doi: 10.11648/j.ijdsa.20170301.11.

[7] R. Naveen Kumar, "An efficient image compression using modified embedded zero tree coding with SVD," *Multimed. Tools Appl.*, vol. 83, no. 13, pp. 37795–37812, 2024, doi: https://doi.org/10.1007/s11042-023-16725-8.

[8] J. Zhang, Y. Lu, T. Li, and G. Lei, "Study on the application of embedded zero-tree wavelet algorithm in still images compression," in *ICMIT 2005: Information Systems and Signal Processing*, SPIE, 2006, pp. 93–97. doi: https://doi.org/10.1117/12.664294.

[9] S. Alfarisi, R. M. R. Sitanggang, and A. Christina, "Applied Algebra for Image Compression: A Systematic Literature Review," *J. Ilmu Komput. dan Inform.*, vol. 4, no. 2, pp. 117–126, 2024, doi: https://doi.org/10.54082/jiki.208.

[10] M. F. Febriansyah and T. Sutabri, "Kompresi dan Optimasi Video Streaming Berbasis AI Untuk Pengalaman Penggunaan Multimedia yang Lebih Baik," *J. SAINS STUDENT Res.*, vol. 3, no. 2, pp. 390–394, 2025, doi: https://doi.org/10.61722/jssr.v3i2.4323.

[11] F. Fitroh, J. Nurhidayah, and Z. Zulfiandri, "Tren dan Tantangan Arsitektur Komputasi Neuromorfik: Tinjauan Literatur Sistematis," *J. Teknol. Sist. Inf.*, vol. 6, no. 1, pp. 103–113, 2025, doi: https://doi.org/10.35957/jtsi.v6i1.10046.

[12] D. Suryadi, C. S. Octiva, T. I. Fajri, U. W. Nuryanto, and M. L. Hakim, "Optimasi Kinerja Sistem IoT Menggunakan Teknik Edge Computing," *J. Minfo Polgan*, vol. 13, no. 2, pp. 1456–1461, 2024, doi: 10.33395/jmp.v13i2.14102.

[13] R. Parapat, "Kompresi Video Digital Menggunakan Metode Embedded Zerotree Wavelet (EZW)," *J. Comput. Informatics Res.*, vol. 1, no. 3, pp. 65–70, 2022, doi: https://doi.org/10.47065/comforch.v1i3.320.

[14] A. Yasir and B. S. Hasugian, "Penggunaan Teknik Kompresi Jpeg Dalam Perancangan Kompresi Citra Digital Memakai Fungsi Gui Pada Matlab," *War. Dharmawangsa*, vol. 16, no. 4, pp. 1056–1066, 2022, doi: https://doi.org/10.46576/wdw.v16i4.2454.

[15] I. Syuhada, "Implementasi Algoritma Arithmetic Coding dan Sannon-Fano Pada Kompresi Citra PNG," *TIN Terap. Inform. Nusant.*, vol. 2, no. 9, pp. 527–532, 2022, doi: https://doi.org/10.47065/tin.v2i9.1027.

[16] O. Irawati, N. Nurhasanah, A. T. Lumbantoruan, and S. F. Zahrani, "Analisis Peforma Algoritma Kompresi Huffman, LZW, BZIP2, Zstandard dan Brotli dalam Efisiensi Penyimpanan Data dan Transfer Data," *RIGGS J. Artif. Intell. Digit. Bus.*, vol. 4, no. 2, pp. 5163–5169, 2025, doi: https://doi.org/10.31004/riggs.v4i2.1396.

[17] J. L. Phandany, A. M. Sambul, and A. S. M. Lumenta, "Studi Perbandingan Algoritma Kompresi Optimal Citra Digital Menggunakan Python," *J. Tek. Elektro dan Komput.*, vol. 11, no. 1, pp. 23–34, 2022, doi: https://doi.org/10.35793/jtek.v11i1.37209.

[18] T. Susim and C. Darujati, "Pengolahan citra untuk pengenalan wajah (face recognition) menggunakan OpenCV," *J. Syntax Admiration*, vol. 2, no. 3, pp. 534–545, 2021, doi: https://doi.org/10.46799/jsa.v2i3.202.

[19] R. I. Dinata and M. Pratama, "Hubungan antara social comparison dengan body image dewasa awal pengguna media sosial tiktok," *Ranah Res. J. Multidiscip. Res. Dev.*, vol. 4, no. 3, pp. 217–224, 2022, doi: https://doi.org/10.38035/rrj.v4i3.477.

[20] E. Halawa and S. Purba, "Aplikasi Watermarking Dengan Metode Discrete Cosine Transform (Dct)," *J. Sains dan Teknol. ISTP*, vol. 18, no. 2, pp. 142–148, 2023, doi: https://doi.org/10.59637/jsti.v18i2.219.

[21] M. S. Moelya, P. S. Ramadhan, and M. G. Suryanata, "Perbandingan Metode Canny, Sobel, Dan Laplacian of Gaussian Dalam Mendeteksi Tepi Citra Objek Bergerak," *J. Sist. Inf. Triguna Dharma (JURSI TGD)*, vol. 3, no. 4, pp. 450–460, 2024, doi: https://doi.org/10.53513/jursi.v3i4.6466.

[22] B. D. Raharja, "PENERAPAN DISCRETE COSINE TRANSFORM (DCT) TERHADAP KOMPRESI CITRA DIGITAL," *Indones. J. Bus. Intell.*, vol. 4, no. 1, pp. 31–36, 2021, doi: http://dx.doi.org/10.21927/ijubi.v4i1.1790.

ADA Journal of Information System Research
Volume 2, No 2, February 2025 Page: 68−76
ISSN 3025-9568 (media online)
DOI: doi.org/10.64366/adajisr.v2i2.76

[23] H. Rayra, Y. Yassir, and R. Rachmawati, "Penggunaan Koreksi Gamma Dengan Metode Robert, Prewitt, Dan Sobel Untuk Penyempurnaan Gambar Pada Citra Dalam Air," *J. TEKTRO*, vol. 7, no. 1, pp. 65–71, 2023, doi: http://dx.doi.org/10.30811/tektro.v7i1.3863.

[24] H. S. Pal, A. Kumar, A. Vishwakarma, and L. K. Balyan, "A hybrid 2d ecg compression algorithm using dct and embedded zero tree wavelet," in *2022 IEEE 6th Conference on Information and Communication Technology (CICT)*, IEEE, 2022, pp. 1–5. doi: 10.1109/CICT56698.2022.9997915.

[25] P. V Bindu and A. Jabeena, "ROI and Non-ROI Image Compression Using Optimal Zero Tree Wavelet and Enhanced Convolutional Neural Network for MRI Images," *SN Comput. Sci.*, vol. 5, no. 1, pp. 1–9, 2023, doi: https://doi.org/10.1007/s42979-023-02335-6.

[26] P. Padmapriya and V. Rajamani, "A Real-Time Epilepsy Detection Method Using Embedded Zero Tree Wavelet Approach and Support Vector Machine," *Behav. Neurol.*, vol. 2025, no. 1, pp. 1–21, 2025, doi: https://doi.org/10.1155/bn/5916201.