

Plagiarism Checker X Originality Report

Similarity Found: 16%

Date: Saturday, September 07, 2024

Statistics: 295 words Plagiarized / 1892 Total words

Remarks: Low Plagiarism Detected - Your Document needs Optional Improvement.

Analisa Perbandingan Algoritma Goldbach Dan LZW Pada File WAV Muhammad Bima Arya Irawan Universitas Budidarma, Sisingamangaraja, Sumatera Utara, Indonesia E-Mail : bima.bimaaryawan111111@gmail.com ABSTRACT WAV stands for Waveform Audio Format, an audio file standard developed by Microsoft and IBM. It is a variant of the RIFF bitstream format, similar to the IFF and AIFF formats used on Amiga and Macintosh systems.

Both WAV and AIFF are compatible with Windows and Macintosh platforms. Although WAV files can contain compressed audio, they are typically used for uncompressed audio. The WAV format is part of Microsoft's RIFF specification for multimedia file storage.

Compression is the process of reducing file size without significantly compromising data quality. The effectiveness of compression is evaluated based on factors such as compression time, memory usage, output quality, and the final file format. This research suggests that compressing audio files can help optimize hard disk space.

Various compression algorithms are available, and reducing audio file size can enhance storage efficiency. Keywords: Compression, WAV Audio Files, Goldbach, LZW

PENDAHULUAN Format WAV (*.WAV) adalah salah satu format file audio yang umum digunakan pada sistem operasi Windows, terutama untuk keperluan game dan multimedia.

WAV merupakan format audio mentah, di mana suara direkam langsung dan diubah menjadi bentuk digital melalui proses kuantisasi. Format ini sederhana untuk dibuat dan diproses, namun secara default tidak mendukung kompresi, sehingga dikenal dengan nama PCM (Pulse Code Modulation). Tantangannya adalah ukuran file WAV yang sangat besar, sehingga jarang digunakan di internet.

Misalnya, sebuah lagu dengan kualitas asli CD audio (stereo, 16-bit, 44.100 Hz) akan membutuhkan ruang penyimpanan sekitar 176.400 byte per detik [1][2]. Oleh karena itu, file audio WAV berdurasi 60 detik dapat memakan sekitar 10,584 MB, dan untuk lagu berdurasi 4 menit, kapasitas yang dibutuhkan bisa mencapai 40 MB. Kompresi adalah proses mengurangi ukuran data dari yang besar menjadi lebih kecil tanpa mengorbankan kualitas data secara signifikan.

Beberapa parameter yang menentukan kualitas kompresi meliputi waktu yang dibutuhkan untuk proses kompresi, efisiensi penggunaan memori, kualitas hasil kompresi, dan format keluaran yang dihasilkan[3][4][5]. Banyak metode yang dapat digunakan untuk melakukan kompresi file, akan tetapi penelitian ini untuk melakukan perbandingan metode dalam melakukan kompresi file WAV.

Berbagai jenis file dapat dikompresi untuk mengurangi ukuran dan menghemat ruang penyimpanan atau mempermudah pengiriman. Kompresi pada setiap tipe file dapat menggunakan metode lossless (tanpa kehilangan kualitas) atau lossy (dengan pengorbanan kualitas), tergantung pada format dan kebutuhan pengguna [6][7][8]. Penelitian ini bisa memberikan kontribusi pada pemahaman cara kerja algoritma dalam konteks yang berbeda dari biasanya.

Ini juga bisa mendorong penelitian lebih lanjut dalam aplikasi algoritma non-tradisional dalam pemrosesan sinyal atau kompresi data. Dengan membandingkan algoritma yang sangat berbeda, dapat didapatkan wawasan yang lebih dalam tentang bagaimana dan mengapa algoritma tertentu bekerja dengan cara tertentu dalam konteks aplikasi yang berbeda. Ini bisa membuka pintu untuk inovasi baru dalam teknik kompresi atau pengolahan data. METODE 2.1

Algoritma Goldbach Algoritma Goldbach merujuk pada algoritma yang digunakan untuk membuktikan atau memverifikasi Konjektur Goldbach, salah satu masalah terkenal dalam matematika. Konjektur Goldbach, yang diajukan oleh matematikawan Christian

Goldbach pada 1742, menyatakan bahwa setiap bilangan bulat genap yang lebih besar dari 2 dapat dinyatakan sebagai jumlah dua bilangan prima.

Algoritma Goldbach bekerja berdasarkan Hipotesis Goldbach, yang merupakan sebuah dugaan matematika yang menyatakan bahwa setiap bilangan genap lebih besar dari 2 dapat dinyatakan sebagai jumlah dari dua bilangan prima. Algoritma ini tidak membuktikan hipotesis tersebut, tetapi mencoba menemukan dua bilangan prima yang menjumlahkan bilangan genap tertentu[9]. 2.2

Algoritma LZW Sedangkan Algoritma LZW (Lempel-Ziv-Welch) adalah algoritma kompresi data yang populer dan tanpa kehilangan (lossless), dikembangkan oleh Terry Welch pada tahun 1984. Algoritma ini merupakan penyempurnaan dari metode LZ78 yang diciptakan oleh Abraham Lempel dan Jacob Ziv pada akhir 1970-an. LZW bekerja dengan menggantikan pola berulang dalam data dengan kode yang lebih pendek, mengurangi ukuran data tanpa mengorbankan kualitas.

Algoritma LZW beroperasi dengan prinsip membangun kamus (dictionary) yang berisi pola karakter (string) yang sering muncul dalam data. Ketika pola ditemukan lagi, algoritma menggantinya dengan entri yang sesuai dari kamus, sehingga mengurangi ukuran keseluruhan data. Tabel kode awal berisi semua simbol dasar (karakter) dari data yang akan dikompresi, masing-masing dengan kode unik.

Misalnya, untuk data berbasis teks, tabel awal akan berisi semua karakter ASCII standar dengan kode 0 hingga 255 [10]. HASIL DAN PEMBAHASAN Tahap awal yang dilakukan melibatkan pengambilan nilai heksadesimal menggunakan aplikasi binary viewer, kemudian melanjutkan dengan proses kompresi sesuai dengan aturan dan langkah-langkah algoritma yang digunakan.

Setelah proses kompresi selesai, hasil kompresi dapat dikembalikan ke nilai awal melalui proses dekompresi. Masalah umum yang sering terjadi adalah ukuran file WAV yang cukup besar, yang dapat menyebabkan proses transmisi data memerlukan waktu lama dan alokasi penyimpanan yang besar. Oleh karena itu, dengan melakukan kompresi pada file WAV, dapat membantu mengurangi ruang penyimpanan dan mempercepat transmisi data.

Untuk ukuran file WAV yang akan dikompres sebesar 40 MB. Untuk keperluan perhitungan manual, diambil sampel sebanyak 7 karakter dari nilai heksadesimal file `dwsample.wav`. Nilai sampel heksadesimal yang diambil adalah: 69 73 6F 6D 70 34 32. Tabel 1. Nilai Hexadesimal Nilai _Frekuensi _69 _1 _73 _1 _6F _3 _6D _3 _70 _1 _34 _1 _32 _1 _ Pembacaan frekuensi dilakukan dengan menghitung jumlah

kemunculan setiap nilai. Kemudian, karakter-karakter diurutkan berdasarkan frekuensi kemunculannya, mulai dari karakter dengan frekuensi tertinggi hingga yang terendah.

Tabel 2. Hexa Yang Belum dikompresi Hexa_Binner_Frekuensi_Bit_Frekuensi x Bit _69 _01101001 _1 _8 _8 _73 _01110011 _1 _8 _8 _6F _01101111 _3 _8 _24 _6D _01101101 _3 _8 _24 _70 _01110000 _1 _8 _8 _34 _00110100 _1 _8 _8 _32 _00110010 _1 _8 _8 _TOTAL BIT _88 _Berdasarkan tabel di atas, nilai (karakter) heksadesimal setara dengan 8 bit dalam biner. Dengan demikian, 16 digit heksadesimal memiliki nilai biner sebesar 128 bit.

Untuk mengonversi jumlah bit menjadi byte, bagi total bit dengan 8. Jadi, $128/8 = 16$ byte akan dihasilkan. Tabel 3. Codeword Goldbach N_{2(n+3)} _Primes _Codeword _1 _8 _3+5 _11 _2 _10 _3+7 _101 _3 _12 _5+7 _011 _4 _14 _3+11 _1001 _5 _16 _5+11 _0101 _6 _18 _7+11 _0011 _7 _20 _7+13 _00101 _8 _22 _5+17 _010001 _9 _24 _11+13 _00011 _10 _26 _7+19 _0010001 _11 _28 _11+17 _000101 _12 _30 _13+17 _000011 _13 _32 _13+19 _0000101 _14 _34 _11+23 _00110001 _Karakter yang diubah menjadi codeword adalah karakter yang telah diurutkan sebelumnya berdasarkan frekuensinya. Tabel 4.

Kompresi Nilai WAV Goldbach Karakter_Frek_{2[n+3]} _Prima _Codeword _Bit_FrekxBit _69 _1 _8 _3+5 _11 _2 _2 _73 _1 _10 _3+7 _101 _3 _3 _6F _3 _12 _5+7 _011 _3 _3 _6D _3 _14 _3+11 _1001 _4 _12 _70 _1 _16 _5+11 _0101 _4 _4 _34 _1 _18 _7+11 _0011 _4 _4 _32 _1 _20 _7+13 _00101 _5 _5 _Total Bit _39 _Metode LZW akan memulai dengan inisialisasi berdasarkan kode ASCII, yaitu dari 0 hingga 255, dan indeks dalam kamus akan dimulai dari 256 dan seterusnya.

Berikut adalah tahapan penerapan kompresi menggunakan metode LZW. Tabel 5. Inisialisasi Sampel _Decimal _Sampel _Decimal _69 _105 _00 _00 _73 _115 _06 _6 _6F _111 _57 _87 _6D _109 _0A _12 _70 _112 _6D _109 _34 _52 _6F _111 _32 _50 _76 _118 _Selanjutnya dilakukan proses sesuai dengan tahapan yang ada pada algoritma LZW. Sehingga menghasilkan kompresi sebagai berikut ini. Tabel 6.

Kompresi INPUT= P _NxCH=C _OUTPUT _ADDTODICTIONARY _ _ _INDEX _INDEX _STRING _00 _00 _0 _256 _0000 _00 _00 _ _ _00 _18 _0 _257 _0018 _18 _66 _24 _258 _1866 _66 _74 _102 _259 _6674 _74 _79 _116 _260 _7479 _79 _70 _121 _261 _7970 _70 _64 _112 _262 _7064 _64 _61 _100 _263 _6461 _61 _73 _97 _264 _6173 _73 _68 _115 _265 _7368 _68 _00 _2 _266 _6800 _00 _00 _ _ _00 _00 _ _ _0000 _00 _256 _267 _000000 _00 _69 _0 _268 _0069 _69 _73 _105 _269 _6973 _73 _6F _115 _270 _736F _6F _36 _111 _271 _6F36 _36 _6D _54 _272 _366D _6D _70 _109 _273 _6D70 _70 _34 _112 _274 _7034 _34 _31 _52 _275 _3400 _31 _00 _49 _276 _3100 _

_00_00_0000_02_0_277_000002_02_60_2_278_0260_60_6D_96_279_606D_6D_6F_109_280_6D6F_6F_6F_111_281_6F6F_6F_76_111_282_6F76_76_-_118_283_76_

Berdasarkan analisis perbandingan kinerja algoritma Goldbach dan algoritma Lempel-Ziv-Welch, dapat disimpulkan bahwa algoritma Goldbach terbukti lebih efektif dan optimal dalam mengompresi file gambar dibandingkan dengan algoritma Lempel-Ziv-Welch (LZW).

KESIMPULAN Berdasarkan analisis yang dilakukan, penulis menarik kesimpulan sebagai berikut: **Kompresi file audio WAV** telah berhasil dilakukan menggunakan metode kompresi perbandingan antara algoritma Goldbach Codes dan LZW. Hal ini menunjukkan bahwa aplikasi berfungsi sesuai dengan teknik kompresi yang digunakan. Penelitian ini menunjukkan bahwa dengan menggunakan algoritma Goldbach Codes dan LZW, file audio WAV yang awalnya berukuran besar dapat dikompresi menjadi ukuran yang lebih kecil, sehingga **mengurangi kebutuhan ruang penyimpanan**. Pengujian menunjukkan bahwa algoritma Goldbach Codes mencapai efisiensi kompresi sebesar 55%.

UCAPAN TERIMKASIH **Penulis mengucapkan terimakasih ke** pembimbing yang telah banyak memberikan masukan dalam penyelesaian penelitian ini. Ucapan terima kasih juga penulis ucapkan ke program studi **yang telah memberikan kesempatan untuk** melakukan penelitian ini. DAFTAR PUSTAKA **Muhammad Abdul Rivai Hi Wahab, 2017, "Implementasi Mekanisme Keamanan Data Dalam Bentuk Steganografi Dengan Metode Least Significant Bit (Lsb) Pada File Audio Wav", Juristek, Vol. 5, No. 2, Januari 2017, ISSN 2301-704X, Hal. 202-211, Magister Ilmu Komputer, Sistem Informasi, Universitas Budi Luhur Jakarta Nadia.FR dkk, 2020. "Penerapan Algoritma Elias Delta Codes Dalam Kompresi File Teks".**

Building of Informatics, Technology and Science (BITS) Volume 2, No 2, December 2020 Page 109-114, Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia. Jamiatul Sisca, 2021. "Penerapan Algoritma Elias Delta Code Untuk Kompresi File Video Pada Aplikasi Video Downloader". Rekayasa Teknik Informatika dan Informasi ISSN 2745-7966 (Media Online) Vol 1, No 4, Maret 2021 Hal 254-264, Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia. Ledi Varia Simanjuntak, 2020. "Perbandingan Algoritma Elias Delta Code dengan Levenstein Untuk Kompresi File Teks".

Journal of Computer System and Informatics (JoSYC) Volume 1, No. 3, May 2020, Page 184-190, Program Studi Teknik Informatika, STMIK Budi Darma, Medan, Indonesia. Simatupang, Julianto, 2019. "Perancangan Sistem Informasi Pemesanan Tiket Bus Pada PO. Handoyo Berbasis Online". Volume 3, No. 2 Oktober 2019 ISSN. 2549-0222 Jurnal

Intra-Tech. Supiyandi dan Okta Frida, "kompresi data huffman audio," vol. 2, no. 1, pp. 71–80, 2020.

UlfaLu'luilmaknun, Nilza Humaira Salsabila, "Penggunaan Metode Run Length Encoding Untuk Kompresi Data," J. Tek. Inform. Unika St. Thomas, vol. 03, no. 1, pp. 45–51, 2020,[Online].Available:<http://ejournal.ust.ac.id/index.php/JTIUST/article/view/245/263> F. Masruri and S. Teknik, "Kompresi Citra Digital Menggunakan Kode Huffman," 2020. [9] M.

Yogie, "PenerapanAlgoritma Goldbach Codes pada Kompresi File Gambar TerenkripsiVigenereCihper," Pelita Inform. Budi Darma, vol. 17, no. April, pp. 316–320, 2020. [10] H. Yuni, A. Sinaga, and L. Sitorus, "Pengamanan File Citra Digital Dengan Menggunakan Metode Least Significant Bit Dan End Of File," vol. 02, pp. 33–41, 2017. [11] W. F. Mahmudy and J.

Matematika, "Steganografi Pada File Citra Bitmap 24 Bit Untuk Pengamanan Data Menggunakan Metode Least Significant Bit (LSB) Insertion," no. 2, pp. 38–44. [12] I. F. Interpretasi, "Format Citra dan Struktur Data untuk Citra," 2021.

INTERNET SOURCES:

<1% - <https://restream.io/learn/what-is/wav-file-format/>
1% - <http://www.hootech.com/freeware/burn-wav-to-cd.htm>
<1% - <http://everything.explained.today/WAV/>
<1% - <http://soundfile.sapp.org/doc/WaveFormat/>
<1% - <https://hivo.co/blog/streamline-and-condense-the-power-of-file-compression>
<1% -
<https://www.akurat.co/infotech/1302018964/Perbedaan-File-Audio-MP3-dan-WAV>
<1% - <https://carisinyal.com/macam-macam-format-kompresi/>
<1% -
<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2018-2019/Makalah/Makalah-Matdis-2018-123.pdf>
<1% -
<https://repository.unikom.ac.id/45255/1/Bab%208%20-%20Kompresi%20Audio%20Video.pdf>
<1% -
<https://ichi.pro/id/analisis-algoritma-bagian-1-belajar-menganalisis-algoritma-138327494110473>
1% - https://www.wikiwand.com/id/Konjektur_Goldbach
<1% -

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2007-2008/Makalah/MakalahIF2153-0708-110.pdf>

1% - https://id.wikipedia.org/wiki/Konjektur_Goldbach

<1% -

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2015-2016/Makalah-2016/MakalahStima-2016-103.pdf>

<1% - <https://ejurnal.stmik-budidarma.ac.id/index.php/inti/article/viewFile/422/370>

1% -

https://www.researchgate.net/publication/378455661_Perancangan_Aplikasi_Kompresi_File_MP3_Dengan_Menggunakan_Algoritma_Lempel_Ziv_Welch_LZW

<1% - <https://ejurnal.dipaneegara.ac.id/index.php/sisiti/article/view/41-50/249>

<1% - <https://www.capcut.com/id-id/resource/how-to-compress-wav-files>

<1% -

<https://www.neliti.com/publications/282583/analisis-perbandingan-algoritma-arithmetic-coding-dengan-algoritma-lempel-ziv-we>

<1% - <https://epub.imandiri.id/repository/docs/journal/Jurnal%20361501013.pdf>

<1% -

https://www.researchgate.net/publication/351280913_Implementasi_Algoritma_Elias_Omega_Coding_Untuk_Kompresi_File_Teks_Pada_Aplikasi_Media_Pembelajaran_Buku_Sekolah_Digital_Untuk_Pelajar

<1% -

<https://sumsel.tribunnews.com/2023/12/14/4-contoh-ucapan-terima-kasih-dalam-jurnal-skripsi-singkat-dan-penuh-rasa-syukur?page=all>

<1% - <https://ejurnal.co.id/contoh-ucapan-terimakasih-pada-jurnal>

1% -

<http://repository.potensi-utama.ac.id/jspui/bitstream/123456789/5872/3/Daftar%20Pustaka.pdf>

<1% - <http://juristek.utama.ac.id/index.php/jtk/article/download/12/14>

1% -

<https://www.semanticscholar.org/paper/Penerapan-Algoritma-Elias-Delta-Codes-Dalam-File-Rizky-Nasution/02a088b07aa5c2986d5a7eb69505e19396711d45>

<1% -

<https://pdfs.semanticscholar.org/80b5/737773cb20db3e176cb64d5c152e66ca93da.pdf>

1% - <https://journal.grahamitra.id/index.php/bios/article/view/148>

<1% - <http://djournals.com/resolusi/article/download/767/565/3706>

<1% -

<http://download.garuda.kemdikbud.go.id/article.php?article=1604488&val=18036&title=Penerapan%20Algoritma%20Ant%20Colony%20Optimization%20Pada%20Aplikasi%20Pemandu%20Wisata%20Provinsi%20Sumatera%20Utara%20Berbasis%20Android>

<1% - <https://www.semanticscholar.org/author/Julianto-Simatupang/90599297>

<1% -

https://www.researchgate.net/publication/347677297_Analisis_Perbandingan_Kompresi_File_Wav_Menggunakan_Metode_Huffman_dan_Run_Length_Encoding/fulltext/6384f5f27b0e356feb92e768/Analisis-Perbandingan-Kompresi-File-Wav-Menggunakan-Metode-Huffman-dan-Run-Length-Encoding.pdf

<1% - <https://sistemasi.ftik.unisi.ac.id/index.php/stmsi/article/view/4175>

2% - <https://ejurnal.stmik-budidarma.ac.id/index.php/komik/article/view/5729>

<1% -

<https://www.neliti.com/publications/282555/pengamanan-file-citra-digital-dengan-menggunakan-metode-least-significant-bit-da>

1% - <https://ojs3.unpatti.ac.id/index.php/barekeng/article/view/8997>

1% -

https://www.researchgate.net/publication/280698037_Steganografi_pada_file_citra_bitmap_24_bit_untuk_pengamanan_data_menggunakan_metode_least_significant_bit_LSB_insertion

<1% -

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2020-2021/04-Format-citra-dan-struktur-data-citra-2021.pptx>