

ANALISA PENERAPAN ALGORITMA GOLDBACH CODES DAN METODE SHANON-FANO PADA KOMPRESI FILE GAMBAR

Raymond David Simaremare¹, Hery Sunandar², Rian Syahputra³
^{1,2,3}Universitas Budi Darma, Medan
¹raytuansimare@gmail.com, ²herysun1975@gmail.com,
³ryansyah93@gmail.com

ABSTRAK

Dalam era digital modern, pengelolaan ukuran file gambar menjadi sangat penting, terutama untuk perangkat dengan kapasitas penyimpanan yang terbatas. Gambar umumnya memiliki ukuran besar karena menyimpan banyak informasi warna dan detail, yang dapat mempersulit penyimpanan dan transmisi. Untuk mengatasi masalah ini, teknik kompresi data diperlukan untuk mengurangi ukuran file tanpa mengorbankan kualitas gambar secara signifikan, terutama untuk gambar yang dihasilkan oleh perangkat seperti kamera DSLR. Penelitian ini fokus pada perbandingan dua algoritma kompresi, yaitu Goldbach Codes dan Shanon-Fano, dalam mengompresi gambar berformat *.JPG. Analisis dilakukan menggunakan Pengukuran Performa Ekspresional (MPE) untuk menilai efisiensi masing-masing algoritma. Hasil penelitian menunjukkan bahwa algoritma Goldbach Codes memiliki nilai prioritas keputusan yang lebih tinggi (7,976) dibandingkan dengan Shanon-Fano (7,868). Ini menunjukkan bahwa Goldbach Codes lebih efektif dalam kompresi gambar, menawarkan rasio kompresi yang lebih baik dan penghematan ruang yang lebih kecil. Dengan kata lain, Goldbach Codes terbukti lebih unggul dalam mengurangi ukuran file gambar dengan kecepatan dan efektivitas yang lebih baik dibandingkan Shanon-Fano. Temuan ini penting untuk aplikasi yang memerlukan kompresi gambar berkualitas tinggi dalam format yang lebih efisien dan hemat ruang.

Kata Kunci : file gambar, kompresi, goldbach codes, shannon fano, MPE

ABSTRACT

*In the modern digital era, managing image file sizes is crucial, particularly for devices with limited storage capacity. Images typically have large file sizes due to the extensive amount of color and detail information they contain, which can complicate storage and transmission. To address this issue, data compression techniques are essential for reducing file sizes without significantly sacrificing image quality, especially for images captured by DSLR cameras. This study compares two compression algorithms, Goldbach Codes and Shanon-Fano, to evaluate their effectiveness in compressing *.JPG images. Analysis using Measurement of Performance Expressional (MPE) reveals that Goldbach Codes has a higher priority decision value (7.976) compared to Shanon-Fano (7.868). This indicates that Goldbach Codes is more efficient in image compression, providing a better compression ratio and smaller space savings. In other words, Goldbach Codes proves to be superior in reducing image file sizes with greater speed and effectiveness compared to Shanon-Fano. These findings are significant for applications requiring high-quality image compression in a more efficient and space-saving format.*

Keywords: image files, compression, goldbach codes, shannon fano, MPE

A. Pendahuluan

Kebutuhan penyimpanan data digital semakin meningkat pesat, yang tercermin dari kapasitas penyimpanan yang semakin besar pada perangkat modern. Kenaikan ini disebabkan oleh banyaknya aktivitas digital seperti browsing internet, menonton video, dan bermain game (Reniaty, 2022). Salah satu jenis data digital yang memerlukan banyak ruang penyimpanan adalah file gambar (Huffman et al., 2021). Gambar umumnya memiliki ukuran besar karena menyimpan banyak informasi warna dan detail, yang bisa menjadi masalah terutama bagi perangkat dengan kapasitas penyimpanan terbatas (Irwansyah, 2013);(Huda, 2020). Oleh karena itu, teknik kompresi data diperlukan untuk mengurangi ukuran file gambar tanpa mengorbankan kualitasnya, terutama untuk gambar yang dihasilkan dari perangkat seperti kamera DSLR. Kompresi file gambar diterapkan di berbagai situasi, termasuk pada situs web dan aplikasi web (Karim, 2020). Menurut situs web pengembang AMD, kompresi gambar mempercepat waktu pemuatan halaman dan mengurangi konsumsi bandwidth, sehingga mempercepat

pengalaman pengguna. Selain itu, media sosial seperti Facebook, Twitter, dan Instagram seringkali secara otomatis mengompresi gambar yang diunggah untuk menghemat ruang penyimpanan dan mempermudah proses unggah.

Mengingat masalah ini dan penerapan teknik kompresi gambar, penelitian ini fokus pada dua algoritma kompresi, yaitu *Goldbach Codes* dan *Shanon-Fano*. Perbandingan kedua metode ini penting karena banyaknya algoritma kompresi yang ada saat ini (Solin, 2022). Penelitian ini mengacu pada studi sebelumnya yang membandingkan algoritma kompresi video untuk menentukan algoritma mana yang lebih baik atau lebih mudah digunakan untuk kompresi file gambar (Simbolon, 2024). *Algoritma Goldbach Codes* menggunakan prinsip dekomposisi bilangan bulat menjadi penjumlahan dua bilangan prima (Almurtada, 2018). Sementara *Shanon-Fano* adalah metode pengkodean entropy yang menghasilkan kode lebih pendek untuk simbol yang sering muncul dan kode lebih panjang untuk simbol yang jarang muncul (Adhitama, 2019);(Pratama, 2017). Kedua

metode ini mewakili pendekatan berbeda dalam kompresi gambar, masing-masing dengan kelebihan dan kekurangan.

Penelitian ini menganalisis penerapan *algoritma Goldbach Codes dan Shannon-Fano* untuk kompresi file gambar, dengan tujuan mencapai kompresi yang efisien dengan rasio kompresi yang tinggi dan kualitas gambar yang baik. Rasio kompresi yang tinggi berarti ukuran file gambar lebih kecil, sehingga memerlukan ruang penyimpanan yang lebih sedikit dan memungkinkan proses pengiriman yang lebih cepat. Kualitas gambar yang baik memastikan bahwa gambar tetap jelas dan memadai setelah dekompresi.

Parameter yang digunakan untuk menilai kualitas kompresi adalah *Ratio of Compression (RC)*, *Compression Ratio (CR)*, dan *Space Saving (SS)*. Perbandingan kinerja kedua *algoritma* ini dilakukan dengan metode eksponensial untuk mengetahui *algoritma* mana yang lebih efisien dalam kompresi. Hasil perbandingan ini akan membantu memilih *algoritma* terbaik untuk kompresi file gambar berdasarkan kriteria efisiensi dan kualitas.

Sehingga berdasarkan latar belakang masalah di atas berikut judul penelitian yang dibahas yaitu “Analisa Penerapan Algoritma Goldbach Codes Dan Metode Shannon-Fano Pada Kompresi File Gambar”.

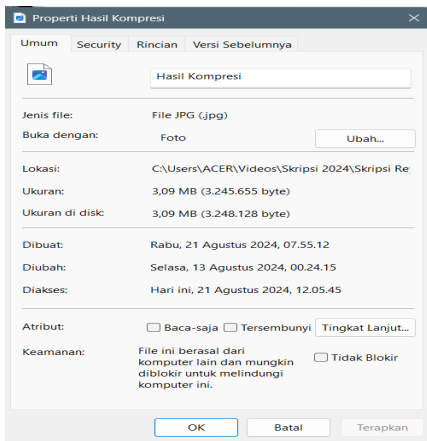
B. Metode Penelitian

Penelitian ini menggunakan metode deskriptif untuk menggambarkan, mendeskripsikan, dan memverifikasi penerapan *algoritma Goldbach Codes* dan metode *Shannon-Fano* dalam kompresi file gambar. Proses penelitian meliputi identifikasi masalah untuk memahami penyebab kapasitas penyimpanan penuh, studi pustaka untuk memperoleh referensi dari berbagai sumber, dan penerapan *algoritma Goldbach Codes* untuk kompresi lossless. Selanjutnya, metode *Shannon-Fano* diterapkan dalam kompresi dan dekompresi file gambar. Analisis dilakukan untuk membandingkan kedua metode dalam hal ukuran file yang dikompresi dan kualitas hasil dekompresi. Perancangan sistem dan implementasi diikuti oleh pengujian aplikasi untuk memastikan kesesuaian dengan penerapan, dan dokumentasi akhir menyimpan

semua hasil pengujian dan perbandingan performa.

Sampel Data

Sampel data yang digunakan dalam penelitian ini adalah jenis gambar dengan ekstensi *.JPG. Adapun spesifikasi sampel data gambar tersebut dapat dilihat pada gambar berikut ini.



Gambar 1 Sample Data Gambar

C. Hasil Penelitian dan Pembahasan

Penerapan Algoritma Goldbach Codes

Tabel 1 Nilai Hexa yang sebelum dikompresi

| No | CHAR | Biner | Frekuensi | Bit | Bit*Frekuensi |
|-------|------|----------|-----------|-----|---------------|
| 1 | 00 | 00000000 | 19 | 8 | 152 |
| 2 | 01 | 00000001 | 6 | 8 | 48 |
| 3 | 05 | 00000101 | 2 | 8 | 16 |
| 4 | 04 | 00000100 | 1 | 8 | 8 |
| 5 | 1A | 00011010 | 1 | 8 | 8 |
| 6 | 97 | 10010111 | 1 | 8 | 8 |
| 7 | 1B | 00011011 | 1 | 8 | 8 |
| 8 | 9F | 10011111 | 1 | 8 | 8 |
| Total | | | | | 256 |

Berdasarkan tabel 1, satu karakter bernilai delapan bit dalam

Dalam penerapan Goldbach Codes untuk kompresi file gambar, langkah pertama adalah mengonversi file gambar yang berformat .Jpg menjadi format hexadecimal. Selanjutnya, file hexadecimal ini akan dikompresi menggunakan Goldbach Codes dan disimpan dalam dua file terpisah: satu file untuk hasil kompresi dan satu file lagi untuk header yang berfungsi sebagai kunci dalam proses dekompresi.

Proses Kompresi

Dalam melakukan analisa kompresi file gambar dengan menerapkan algoritma *goldmach Codes* menggunakan sampel yaitu nilai *headecimal* dari file gambar yang berisikan = **04 00 01 00 00 00 01 00 00 00 1A 01 05 00 01 00 00 00 97 00 00 00 1B 01 05 00 01 00 00 00 9F 00= 32 Byte**

format biner, sehingga 32 karakter pada string memiliki total nilai biner

sebanyak 256 bit. Algoritma Goldbach Codes memulai proses dengan menghitung frekuensi kemunculan setiap karakter dalam string yang akan dikompresi. Sebelum kompresi, string biner yang digunakan adalah sebagai berikut:
 00000000 00000001 00000101
 00000100 00011010 10010111
 00011011 10011111.

Setelah menentukan frekuensi kemunculan setiap karakter dan nilai binernya, langkah selanjutnya adalah mengurutkan karakter berdasarkan frekuensi kemunculannya yang tertinggi. Kemudian, bilangan biner sebelum kompresi digantikan dengan kode kebenaran dari algoritma Goldbach Codes yang diurutkan sesuai dengan frekuensi kemunculan.

Tabel 2 Data setelah dikompresi menggunakan algoritma *Goldbach Codes*

| No | Char | Freq | N | 2(n+3) | Bilangan Prime | Codeword | bit | Freq *bit |
|-------|------|------|---|--------|----------------|----------|-----|-----------|
| 0 | 00 | 19 | 0 | 0 | 0 | 11 | 2 | 28 |
| 1 | 01 | 6 | 0 | 0 | 1 | 101 | 3 | 18 |
| 2 | 05 | 2 | 0 | 0 | 2 | 011 | 3 | 6 |
| 3 | 04 | 1 | 0 | 0 | 3 | 1001 | 4 | 4 |
| 4 | 1A | 1 | 0 | 0 | 4 | 0101 | 4 | 4 |
| 5 | 97 | 1 | 1 | 10 | 0 | 0011 | 4 | 4 |
| 6 | 1B | 1 | 1 | 10 | 1 | 00101 | 5 | 5 |
| 7 | 9F | 1 | 1 | 10 | 2 | 010001 | 6 | 6 |
| Total | | | | | | | | 75 Bit |

Proses Dekompresi

Proses dekompresi file gambar yang telah dikompresi dilakukan dengan mengembalikan karakter ke format hexadecimal semula. Dimulai dengan nilai hasil kompresi seperti (04) (19,00) (6,01) (1A) (2,05) (97) (1B) (9F), langkah berikutnya adalah mencari nilai counter dan menggantinya dengan nilai di sampingnya sesuai jumlah nilai counter. Hasil akhir dari dekompresi

algoritma Goldbach Codes adalah string bit yang identik dengan data asli, yaitu bilangan hexadecimal “04 00 01 00 00 00 01 00 00 00 1A 01 05 00 01 00 00 00 97 00 00 00 1B 01 05 00 01 00 00 00 9F 00”, menunjukkan kesesuaian antara data yang dikompresi dan data asli.

Penerapan Algoritma Shannon-Fano

Dalam penerapan algoritma Shannon-Fano untuk kompresi file

gambar, langkah pertama adalah mengonversi file .JPG menjadi format hexadecimal. Hasil hexadecimal ini kemudian dikompresi dan disimpan dalam dua file terpisah: satu file untuk hasil kompresi dan satu file header yang berfungsi sebagai kunci untuk dekompresi.

Dalam melakukan analisa kompresi file gambar dengan menerapkan algoritma *goldbach* menggunakan sampel yaitu nilai hexadecimal dari file gambar yang berisikan = **04 00 01 00 00 00 01 00 00 00 1A 01 05 00 01 00 00 00 97 00 00 00 1B 01 05 00 01 00 00 00 9F 00 = 32 Byte**

Proses Kompresi

Tabel 3 Nilai Hexa yang sebelum dikompresi

| No | CHAR | Biner | Frekuensi | Bit | Bit*Frekuensi |
|-------|------|--------------|-----------|-----|---------------|
| 1 | 00 | 0000000 0 | 19 | 8 | 152 |
| 2 | 01 | 0000000 1 | 6 | 8 | 48 |
| 3 | 05 | 0000010 1 | 2 | 8 | 16 |
| 4 | 04 | 0000010 0 | 1 | 8 | 8 |
| 5 | 1A | 0001101 0 | 1 | 8 | 8 |
| 6 | 97 | 1001011 1 | 1 | 8 | 8 |
| 7 | 1B | 0001101 1 | 1 | 8 | 8 |
| 8 | 9F | 1001111 1 | 1 | 8 | 8 |
| Total | | | | | 256 |

Berdasarkan tabel 3 satu karakter bernilai delapan bit bilangan biner sehingga 32 Karakter pada string memiliki total nilai biner sebanyak 256 bit. Algoritma Shannon-Fano memulai dengan menghitung frekuensi kemunculan setiap karakter. String biner sebelum

kompresi adalah: 00000000
 00000001 00000101 00000100
 00011010 10010111 00011011
 10011111. Setelah menentukan frekuensi dan nilai biner setiap karakter, karakter diurutkan berdasarkan frekuensi kemunculan. Kemudian, bilangan biner digantikan

dengan kode Shannon-Fano yang sesuai dengan frekuensi tersebut.

algorithm Shannon-Fano dapat dilihat pada tabel 4.

Hasil kompresi menggunakan

Tabel 4 Data setelah dikompresi menggunakan algoritma *Shannon Fano*

| No | Nilai Hexadecimal | Frekuensi | Shannon Fano | Bit | Bit* Frek |
|-------|-------------------|-----------|--------------|-----|-----------|
| 1 | 00 | 19 | 10 | 2 | 36 |
| 2 | 01 | 6 | 1100 | 4 | 24 |
| 3 | 05 | 2 | 1101 | 4 | 4 |
| 4 | 04 | 1 | 1110000 | 7 | 7 |
| 5 | 1A | 1 | 1110001 | 7 | 7 |
| 6 | 97 | 1 | 1110010 | 7 | 7 |
| 7 | 1B | 1 | 1110011 | 7 | 7 |
| 8 | 9F | 1 | 11101000 | 8 | 8 |
| Total | | | | | 107 bit |

Proses Dekompresi

Proses dekompresi file gambar yang telah dikompresi dilakukan dengan mengembalikan karakter ke format hexadecimal semula. Dimulai dengan nilai hasil kompresi seperti (04) (19,00) (6,01) (1A) (2,05) (97) (1B) (9F), langkah selanjutnya adalah mencari nilai counter dan menggantinya dengan nilai di sampingnya sesuai jumlah counter. Hasil dekompresi menggunakan algoritma Shannon-Fano menunjukkan string bit yang identik dengan data asli, yaitu bilangan hexadecimal "04 00 01 00 00 00 01

00 00 00 1A 01 05 00 01 00 00 00 97 00 00 00 1B 01 05 00 01 00 00 00 9F 00", yang membuktikan kesesuaian antara data yang dikompresi dan data asli.

Metode Perbandingan

Exponential (MPE)

Dalam membandingkan proses pencarian antara algoritma Goldbach Codes dan Shannon-Fano, langkah-langkahnya meliputi: pertama, menentukan alternatif dengan memilih algoritma pencarian yang akan dianalisis; kedua, menetapkan kriteria untuk membandingkan kedua algoritma; ketiga, menentukan bobot

kriteria yang mempengaruhi analisis berdasarkan tingkat pengaruhnya terhadap kecepatan pencarian; keempat, memberikan nilai pada setiap kriteria sesuai dengan hasil analisis algoritma sebelumnya; dan

terakhir, menghitung nilai menggunakan rumus dari metode perbandingan eksponensial (MPE) setelah nilai kriteria diisi.

Tabel 5. Perhitungan analisa perbandingan dengan menggunakan metode perbandingan MPE

| Kriteria | Bobot | Algoritma Goldbach Codes | Algoritma Shannon Fano |
|-----------------------------------|-------|--------------------------|------------------------|
| CR | 0,3 | 29,30 | 41,80 |
| RC | 0,4 | 3,41 | 2,39 |
| SS | 0,3 | 70,70 | 58,30 |
| Total Nilai $\sum(N)^B$ | 1 | 7,976 | 7,868 |

Menentukan hasil atau prioritas keputusan

Setelah memperoleh nilai akhir dari masing-masing algoritma, langkah selanjutnya adalah menentukan prioritas keputusan berdasarkan nilai tersebut. Analisis menunjukkan bahwa algoritma Goldbach Codes adalah yang tercepat dan paling efektif dalam proses kompresi file gambar, karena semakin kecil total nilai yang diperoleh, semakin sedikit usaha yang diperlukan oleh algoritma untuk melakukan kompresi.

Perancangan dan Pemodelan Sistem

Tahap perancangan dan pemodelan melibatkan pembuatan

system untuk mengelola data yang diperlukan. perancangan ini bertujuan memberikan gambaran keseluruhan kepada pengguna mengenai sistem yang akan dikembangkan

Perancangan

Pada perancangan ini merupakan gambaran dari langkah-langkah pengembangan sistem untuk memahami proses data yang dibutuhkan. Perancangan ini dimaksudkan untuk memberikan gambaran umum kepada pengguna tentang sistem yang akan dibangun.

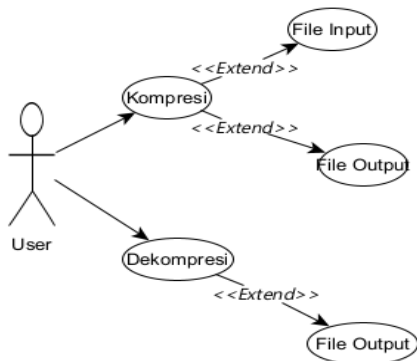
Pemodelan *Unified Modeling Language* (UML)

Dalam penelitian ini penulis menggunakan pemodelan sistem untuk mendapatkan gambaran

lengkap mengenai sistem yang akan dibangun. Proses pemodelan sistem akan dilakukan dengan menggunakan *use case diagram* dan *activity diagram*.

Use Case Diagram

Use Case Diagram adalah diagram dalam rekayasa perangkat lunak yang menggambarkan interaksi antara pengguna (atau sistem lain) dengan sistem yang dikembangkan (Astino, 2023). Bagian dari Unified Modeling Language (UML), diagram ini memberikan gambaran umum mengenai fungsionalitas sistem dari perspektif pengguna atau aktor.



Gambar 2 Use Case Diagram

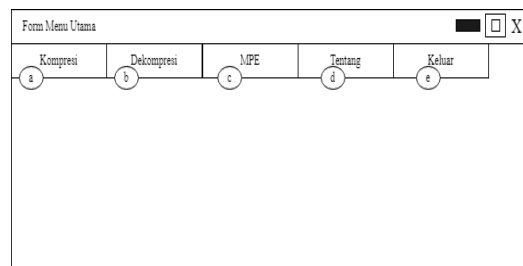
Perancangan Antarmuka (Interface)

Perancangan antarmuka merupakan sarana interaksi antara pengguna (*user*) dengan sistem. Pada penelitian ini, sistem dibangun dan dirancang dengan menggunakan bahasa

pemrograman Microsoft Visual Studio 2010.

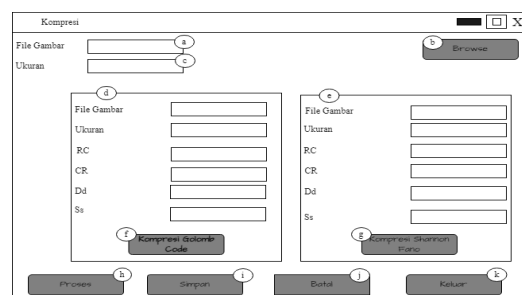
Rancangan Halaman Menu Utama

Menu utama merupakan *form* utama dari sebuah aplikasi, dimana *form* ini nantinya digunakan untuk mengakses *sub-sub* menu pada aplikasi yang dibangun.



Gambar 3 Form Menu Utama
Rancangan Form Kompresi

Form kompresi adalah antarmuka pengguna yang memungkinkan pengguna memilih file gambar untuk dikompresi. Desain antarmuka dari form kompresi dapat dilihat pada gambar berikut:

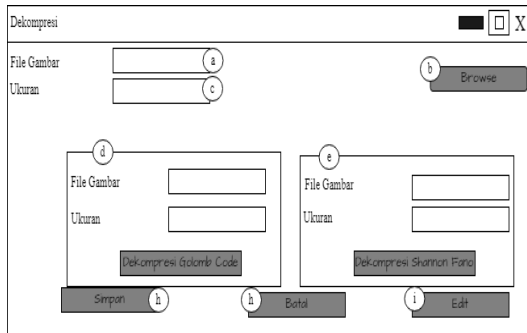


Gambar 4 Form Kompresi

Rancangan Form Dekompresi

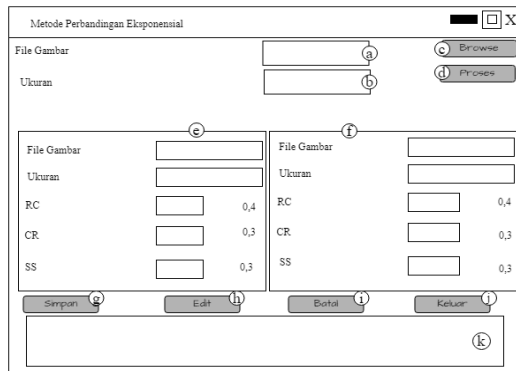
Form dekompresi merupakan *interface* bagi pengguna (*user*) pada saat melakukan kegiatan

dekompresi. Pada form ini, *user* dapat memilih *file* gambar yang telah dikompresi untuk dikembalikan ke bentuk semula. Adapun *interface* dari *form* dekompresi dapat dilihat pada gambar berikut :



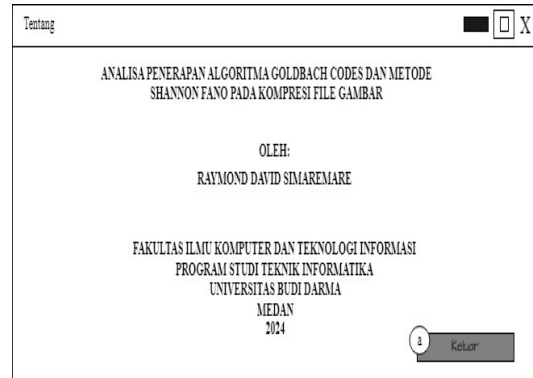
Gambar 5 Form Dekompresi Form Metode Perbandingan Eksponensial

Form ini adalah *form* yang digunakan untuk menampilkan perbandingan antara dua algoritma yang dipakai. Adapun *form* tersebut dapat dilihat pada gambar ini:



Gambar 6 Form Metode Perbandingan Eksponensial Perancangan Form Tentang

Form Penulis merupakan data diri dari penulis pada penelitian ini. Adapun rancangan dari *form* penulis dapat dilihat pada gambar dibawah ini :



Gambar 7 Form Tentang Implementasi

Implementasi program mencakup tahap pengujian sistem yang telah dirancang, termasuk spesifikasi perangkat keras dan perangkat lunak serta tampilan sistem saat berjalan. Spesifikasi perangkat keras yang diperlukan meliputi processor Intel(R) Celeron(R) CPU N3050 @ 1.60GHz, RAM 4 GB, harddisk 500 GB, monitor dengan resolusi 1366 x 768 pixel, serta keyboard dan mouse. Untuk perangkat lunak, spesifikasi minimum termasuk sistem operasi Windows 10, aplikasi Hxd, Microsoft Visual Studio 2008, dan file gambar yang diperlukan untuk pengujian.

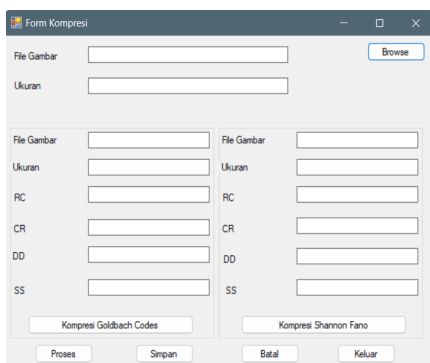
Implementasi Sistem

Implementasi sistem merupakan Implementasi akhir dari antarmuka sistem yang dirancang. Pada sistem ini terdapat beberapa tampilan halaman, yaitu *Form* Menu Utama,

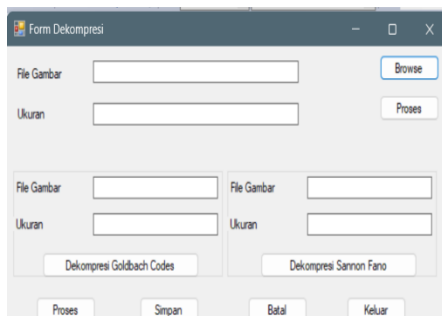
Form Kompresi, Form dekompresi, eksponensial dan Form Penulis.



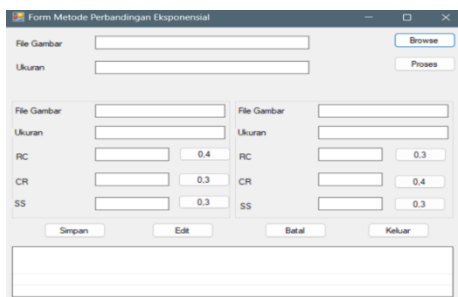
Gambar 8 Form Menu Utama



Gambar 9 Form kompresi



Gambar 10 Form Dekompresi



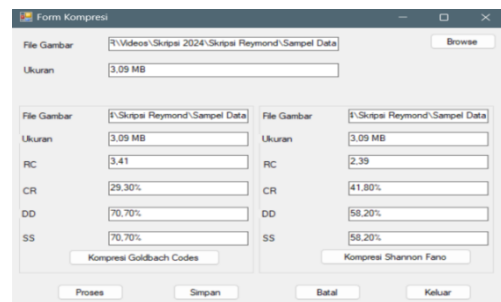
Gambar 11 Form MPE Hasil Pengujian

Pengujian sistem bertujuan untuk mengetahui bagaimana sistem yang telah dibangun akan

berjalan. Sistem ini dirancang menjadi sesederhana mungkin agar pengguna dapat dengan mudah mengaksesnya. Dalam hal ini, penulis memberikan print out saat sistem sedang berjalan.

Form Kompresi

Pada form ini akan menampilkan perbandingan proses pengkompresian gambar menggunakan algoritma Goldbach dengan algoritma Shannon Fano, yang dimana user menginputkan terlebih dahulu file gambar yang akan dikompresi.

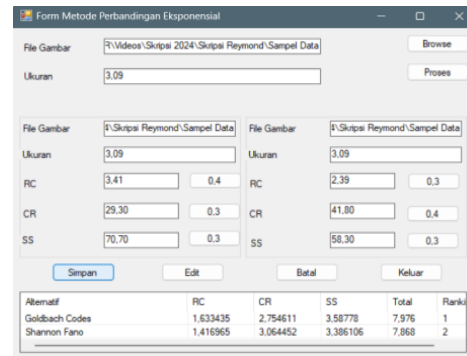


Gambar 12 Hasil Kompresi

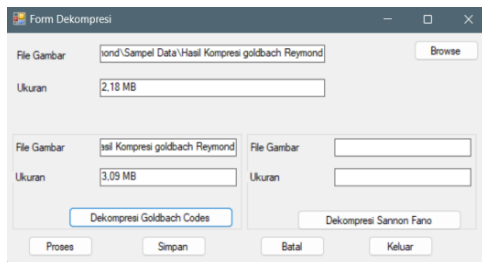
Setelah file gambar berhasil dikompresi dan ukurannya lebih kecil dari file asli, pengguna dapat memilih tombol simpan untuk menyimpan file terkompresi, tombol batal untuk mengulang proses pemilihan file, atau tombol kembali untuk kembali ke menu utama.

Form Dekompresi algoritma goldbach Codes dan Shannon Fano

Pada Form dekomposisi ini menampilkan form dimana sistem dapat melakukan proses dekomposisi file gambar yang telah diinput. Setelah file gambar didekomposisi maka menghasilkan file gambar terdekomposisi.



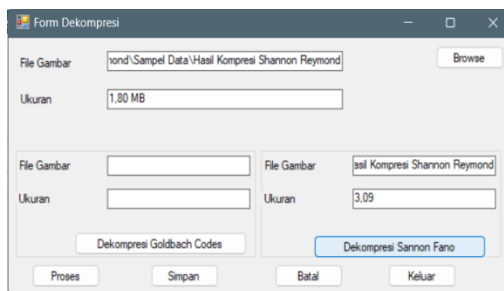
Gambar 15 Hasil MPE



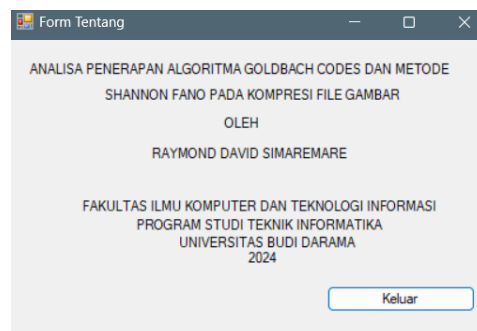
Gambar 13 Hasil Dekomposisi
Goldbach Codes

Form Tentang

Pada *form* penulis berisi tentang biodata penulis penelitian ini. Adapun *form* tentang dapat dilihat pada gambar dibawah ini :



Gambar 14 Hasil Dekomposisi
Shannon Fano



Gambar 16 Tentang

Pengujian proses Metode Perbandingan Eksponential

Pada proses ini pertama yang akan dilakukan adalah mencari file yang akan dibandingkan dengan menekan button “Browse” yang tersedia. Dapat dilihat pada gambar dibawah :

D. Kesimpulan

Berdasarkan hasil penelitian, dapat disimpulkan bahwa penerapan algoritma Goldbach Codes dan Shannon-Fano untuk kompresi file gambar menunjukkan kinerja yang baik, karena dekomposisi tidak mengurangi kapasitas file asli. Pengujian sistem menggunakan Visual Basic 2008 berhasil dilakukan, memungkinkan kompresi dan

perbandingan hasil kompresi file gambar. Meskipun setiap algoritma memiliki kinerja yang berbeda, analisis menggunakan metode eksponensial (MPE) menunjukkan bahwa algoritma Goldbach Codes, dengan nilai 7,976, sedikit lebih unggul dibandingkan algoritma Shannon Fano yang memiliki nilai 7,868. Hasil ini menunjukkan bahwa Goldbach Codes memiliki rasio kompresi yang lebih baik dan nilai space savings yang lebih kecil, sehingga lebih efektif dalam kompresi.

DAFTAR PUSTAKA

- Adhitama. (2019). PERBANDINGAN ALGORITMA HUFFMAN DENGAN ALGORITMA SHANNON-FANO. *JURNAL TEKNIK DAN INFORMATIKA*, 3(1).
- Almurtada. (2018). PENERAPAN ALGORITMA GOLDBACH CODES PADA KOMPRESI FILE TEKS TERENKRIPSI HILL CIHPER. *Jurnal Pelita Informatika*, 6(1), 473–478.
- Astino, Y. R. (2023). Pengembangan Dan Penerapan Sistem Computer Assisted Test (CAT) Untuk Mengelola Ujian Berbasis Website. *JURNAL INFORMATIKA DAN REKAYASA PERANGKAT LUNAK (JATIKA)*, 4(1).
- Huda. (2020). Media Animasi Digital Berbasis Hots (Higher Order Thinking Skill). In *Unp Press*.
- Huffman, A. K., Yonathan, F. D., Nasution, H., & Priyanto, H. (2021). Aplikasi Pengaman Dokumen Digital Menggunakan Algoritma Kriptografi Hybrid dan. *Jurnal Edukasi Dan Penelitian Informatika*, 7(2), 181–195.
- Irwansyah. (2013). Sistem informasi geografis: prinsip dasar dan pengembangan aplikasi . In *DigiBook Yogyakarta*.
- Karim. (2020). Pengantar teknologi informasi. In *Yayasan Labuhanbatu Berbagi Gemilang*.
- Pratama, A. M. (2017). Penerapan algoritma huffman dan shannon-fano dalam pemampatan file teks. *Majalah Ilmiah INTI*, 5(1), 30–35.
- Reniati. (2022). Model pengembangan sumberdaya manusia di era digital. In *CV. Dotplus Publisher*.
- Simbolon, A. D. (2024). Perbandingan Algoritma Elias Omega Dan Algoritma Additive Code Dalam Mengkompresi File Audio Video Interleave (AVI). *KOMIK (Konferensi Nasional Teknologi Informasi Dan Komputer)*, 7(3), 10–17.
- Solin, I. (2022). Perancangan Aplikasi Kompresi File Video Menggunakan Algoritma Transformasi Walsh Hadamard. *JoGTC: Journal Global Tecnology Computer*, 1(2), 63–70.