ISSN: 3110-0864 (Media Online)
JURNAL ILMU KOMPUTER DAN TEKNIK INFROMATIKA
Volume 1 No 1 Januari 2025 | pp: 14-20
https://journals.raskhamedia.or.id/index.php/juikti
DOI: https://doi.org/10.64803/juikti.v1i1.44.

Kajian Teoritis Simulatif Mengenai Algoritma Huffman dalam Kompresi Data Teks

Supiyandi[1], Chairul Rizal[2], Deni Apriadi[3], Muhammad Noor Hasan Siregar[4], Muhammad Iqbal[5*]

[1]Sains Komputasi dan Kecerdasan Digital, Teknologi Informasi, Universitas Pembangunan Panca Budi, Medan, Indonesia.
[2]Sains Komputasi dan Kecerdasan Digital, Sistem Informasi, Universitas Pembangunan Panca Budi, Medan, Indonesia.
[3]Sistem informasi, STMIK Bina Nusantara Jaya Lubuklinggau, Lubuklinggau, Indonesia.
[4]Bisnis Digital, Universitas Graha Nusantara, Padangsidimpuan, Indonesia.
[5]Vocational, Computer Engineering, Institut Teknologi Mitra Gama, Bengkalis, Indonesia.
Email: [1]supiyandi@dosen.pancabudi.ac.id, [2]chairulrizal@dosen.pancabudi.ac.id, [3]denidrv@gmail.com, [4]noor.siregar@gmail.com, [5*]iqbal.kun@gmail.com
(* Email Corresponding Author: iqbal.kun@gmail.com)
Dikirim: 9 November 2024 | Diterima: 15 Januari 2025 | Dipublish: 30 Januari 2025

Abstrak

Kompresi data merupakan salah satu teknik penting dalam pengelolaan informasi digital, terutama untuk efisiensi penyimpanan dan transmisi data. Algoritma Huffman dikenal sebagai salah satu metode kompresi lossless yang paling efisien dalam konteks data teks. Kajian ini bertujuan untuk menyajikan telaah teoritis yang dikombinasikan dengan pendekatan simulatif terhadap penerapan algoritma Huffman dalam proses kompresi data teks. Pembahasan diawali dengan pemaparan konsep dasar kompresi, prinsip kerja algoritma Huffman, dan analisis struktural terhadap pohon Huffman yang terbentuk dari distribusi frekuensi karakter dalam suatu dokumen teks. Simulasi dilakukan menggunakan sampel teks berbahasa Indonesia dan Inggris dengan variasi panjang dan kompleksitas karakter untuk mengamati dampak terhadap rasio kompresi, efisiensi encoding, serta performa algoritma secara keseluruhan. Hasil simulasi menunjukkan bahwa semakin tidak merata distribusi frekuensi karakter dalam data, semakin tinggi efisiensi kompresi yang dicapai. Selain itu, dibandingkan metode kompresi berbasis fixed-length encoding, algoritma Huffman mampu mengurangi ukuran file hingga lebih dari 40% dalam beberapa kasus uji, tanpa kehilangan informasi apa pun. Studi ini menegaskan pentingnya pemahaman algoritma Huffman tidak hanya dari sisi matematis, tetapi juga melalui pendekatan eksperimental untuk mengukur efektivitasnya dalam konteks data teks riil. Penulis merekomendasikan integrasi algoritma Huffman dalam sistem kompresi yang lebih luas, serta pengembangan varian algoritma untuk peningkatan performa pada data yang lebih heterogen. Kajian ini diharapkan dapat memberikan kontribusi bagi pengembangan teknologi kompresi data serta menjadi referensi awal bagi peneliti atau praktisi yang tertarik pada optimalisasi penyimpanan dan pengolahan informasi.

Kata Kunci: Kompresi Data, Algoritma Huffman, Teks Digital, Pohon Biner, Rasio Kompresi.

Abstract

Data compression is a critical technique in digital information management, especially for optimizing storage and transmission. Huffman's algorithm is widely recognized as one of the most efficient lossless compression methods, particularly in the context of textual data. This study presents a theoretical review combined with a simulative approach to examine the implementation of the Huffman algorithm in text data compression processes. The discussion begins with an overview of basic compression concepts, the working principles of the Huffman algorithm, and a structural analysis of Huffman trees generated from the frequency distribution of characters in text documents. The simulation was conducted using both Indonesian and English language text samples with varying lengths and character complexities to observe the impact on compression ratio, encoding efficiency, and overall algorithm performance. Results indicate that the more uneven the frequency distribution of characters, the more effective the compression achieved. Compared to fixed-length encoding methods, the Huffman algorithm can reduce file sizes by more than 40% in some test cases without any data loss. This study emphasizes the importance of understanding the Huffman algorithm not only from a mathematical perspective but also through experimental methods to evaluate its effectiveness in real-world text data scenarios. The findings support the integration of Huffman coding in broader compression systems and suggest the development of algorithmic variants to enhance performance across more heterogeneous datasets. This research is expected to contribute to the advancement of data compression technology and serve as a foundational reference for researchers and practitioners focused on optimizing digital storage and information processing.

Keywords*: Data Compression, Huffman Algorithm, Digital Text, Binary Tree, Compression Ratio.*

1. PENDAHULUAN

Perkembangan teknologi informasi telah membawa manusia ke era ledakan data (*data explosion*), di mana hampir setiap aspek kehidupan menghasilkan data dalam skala besar dan terus meningkat secara eksponensial[1]. Dalam lingkungan digital yang semakin padat ini, efisiensi pengelolaan data menjadi

ISSN: 3110-0864 (Media Online)
JURNAL ILMU KOMPUTER DAN TEKNIK INFROMATIKA
Volume 1 No 1 Januari 2025 | pp: 14-20
https://journals.raskhamedia.or.id/index.php/juikti
DOI: https://doi.org/10.64803/juikti.v1i1.44.

kebutuhan mendesak, terutama dalam hal penyimpanan dan transmisi[2]. Baik itu dalam komunikasi daring, sistem *cloud computing*, pengarsipan digital, hingga perangkat bergerak, ukuran data yang besar dapat menyebabkan keterlambatan, pemborosan sumber daya, serta biaya operasional yang tinggi. Salah satu solusi utama untuk mengatasi permasalahan ini adalah dengan menerapkan teknik kompresi data[3].

Kompresi data adalah proses mengubah representasi data agar ukurannya menjadi lebih kecil, tanpa mengubah makna atau isinya (dalam kompresi *lossless*), atau dengan mengorbankan sebagian informasi yang dianggap tidak esensial (dalam kompresi *lossy*)[4]. Dalam konteks data teks, di mana akurasi setiap karakter sangat penting, hanya metode kompresi lossless yang layak digunakan[5]. Salah satu algoritma paling terkenal dan terbukti efektif dalam kompresi lossless adalah algoritma Huffman, yang diperkenalkan oleh David A[6]. Huffman pada tahun 1952. Algoritma ini merupakan metode pengkodean entropi berbasis pohon biner, yang menghasilkan representasi karakter dengan panjang kode yang berbeda-beda tergantung pada frekuensi kemunculannya dalam data[7].

Konsep dasar dari algoritma Huffman sederhana namun sangat efektif: karakter atau simbol yang paling sering muncul akan diberi kode biner yang paling pendek, sedangkan karakter yang jarang muncul akan diberi kode yang lebih panjang. Hasilnya adalah penurunan ukuran total data karena penggunaan bit yang lebih efisien[8]. Strategi ini sangat sejalan dengan prinsip minimum redundancy dalam ilmu informasi[9]. Skema pengkodean seperti ini telah digunakan secara luas dalam berbagai format file dan sistem kompresi populer, seperti ZIP, GZIP, JPEG (metadata), hingga beberapa bagian dari format PDF[10].

Namun, meskipun Huffman Coding telah dikenal luas, masih terdapat ruang untuk melakukan eksplorasi lebih lanjut, khususnya melalui pendekatan simulatif yang dapat memperlihatkan secara nyata bagaimana algoritma ini bekerja pada data teks riil[11]. Banyak kajian sebelumnya yang lebih bersifat teoritis, menjelaskan algoritma ini hanya dari segi struktur pohon dan kompleksitas algoritmiknya, tanpa menguji performanya terhadap data tekstual dengan karakteristik yang berbeda-beda, seperti bahasa, panjang teks, hingga distribusi karakter. Padahal, efektivitas algoritma Huffman sangat bergantung pada distribusi frekuensi simbol[12]. Dengan kata lain, semakin tidak merata distribusi frekuensinya, semakin tinggi efisiensi kompresinya[13]. Penelitian ini mencoba menjembatani kesenjangan antara aspek teoritis dan praktis dari algoritma Huffman. Kajian dilakukan dengan dua pendekatan utama: pertama, pembahasan teoritis yang mencakup konsep dasar, konstruksi pohon Huffman, hingga evaluasi efisiensi kompresi secara matematis; kedua, pendekatan simulatif yang melibatkan pengujian langsung terhadap sampel teks dalam dua bahasa (Indonesia dan Inggris), dengan tujuan mengamati seberapa besar pengaruh struktur bahasa terhadap hasil kompresi[14]. Proses ini juga mencakup tahapan preprocessing data, konversi ke bentuk biner, pembentukan pohon Huffman, hingga penghitungan rasio kompresi yang dihasilkan[15].

Tujuan utama dari kajian ini adalah untuk memberikan gambaran konkret tentang bagaimana algoritma Huffman bekerja dalam praktik, serta bagaimana karakteristik teks memengaruhi hasil kompresi. Dengan memahami hubungan antara distribusi karakter, panjang data, dan efisiensi kompresi, peneliti dan pengembang dapat menentukan strategi yang lebih optimal dalam menerapkan metode ini dalam berbagai sistem digital. Selain itu, kajian ini juga diharapkan dapat menjadi dasar untuk pengembangan lebih lanjut, baik dalam bentuk modifikasi algoritma Huffman maupun integrasinya ke dalam sistem kompresi adaptif atau real-time.

Tidak dapat diabaikan pula bahwa algoritma Huffman memiliki keterbatasan. Salah satunya adalah sifat statisnya, di mana seluruh data harus dianalisis terlebih dahulu sebelum proses kompresi dilakukan. Ini menyulitkan penerapannya dalam situasi di mana data mengalir secara kontinu (*streaming)* atau tidak diketahui seluruhnya di awal. Di sinilah muncul varian seperti Adaptive Huffman Coding, namun kompleksitas dan overhead komputasi dari pendekatan adaptif menjadi tantangan tersendiri. Oleh karena itu, memahami versi klasik dari algoritma Huffman secara mendalam tetap menjadi landasan penting sebelum mengarah pada pengembangan teknik yang lebih kompleks. Dalam dunia pendidikan dan penelitian, pemahaman terhadap algoritma Huffman juga memiliki peran strategis. Topik ini sering dijadikan bahan ajar dalam mata kuliah algoritma, struktur data, dan teori informasi, karena mencakup banyak aspek penting seperti pohon biner, *graf*, *greedy algorithm*, dan efisiensi data. Di sisi praktis, profesional TI yang bergerak di bidang pengolahan dokumen, sistem file, atau teknologi komunikasi juga membutuhkan pemahaman mendalam terhadap prinsip-prinsip kompresi ini untuk mendesain sistem yang lebih hemat dan efisien. Dengan latar belakang tersebut, penelitian ini bertujuan untuk memberikan pemahaman teoritis dan empiris terhadap algoritma Huffman dalam konteks kompresi data teks. Harapannya, kajian ini tidak hanya bersifat akademik, tetapi juga relevan dan aplikatif untuk kebutuhan dunia nyata yang semakin padat data.

2. METODOLOGI PENELITIAN

ISSN: 3110-0864 (Media Online)
JURNAL ILMU KOMPUTER DAN TEKNIK INFROMATIKA
Volume 1 No 1 Januari 2025 | pp: 14-20
https://journals.raskhamedia.or.id/index.php/juikti
DOI: https://doi.org/10.64803/juikti.v1i1.44.

Untuk mencapai tujuan penelitian ini secara sistematis dan terukur, diperlukan pendekatan metodologis yang mampu menggabungkan analisis teoritis dengan pengujian praktis melalui simulasi. Algoritma Huffman, meskipun secara matematis tergolong mapan, tetap perlu divalidasi efektivitasnya dalam konteks data teks riil dengan beragam karakteristik. Oleh karena itu, metode yang digunakan dalam penelitian ini dirancang untuk menjawab pertanyaan utama mengenai seberapa efisien algoritma tersebut bekerja pada berbagai jenis dan struktur teks.

Gambar 1. Struktur Metode Penelitian

2.1 Jenis dan Pendekatan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif-deskriptif dengan metode simulasi algoritma. Fokus utama penelitian adalah untuk menganalisis secara sistematis dan empiris kinerja algoritma Huffman dalam proses kompresi data teks. Penelitian ini tidak hanya mengulas teori dasar algoritma, tetapi juga melakukan serangkaian simulasi untuk mengukur efektivitas kompresi pada berbagai jenis teks dengan karakteristik yang berbeda.

2.2 Objek dan Sumber Data

Objek dalam penelitian ini adalah teks digital dalam format .txt, yang dipilih dari dua jenis bahasa:
a. Bahasa Indonesia (diperoleh dari berita daring dan dokumen publik)
b. Bahasa Inggris (diambil dari artikel akademik dan literatur umum)

Pemilihan dua bahasa bertujuan untuk melihat pengaruh struktur linguistik terhadap distribusi frekuensi karakter, yang menjadi kunci dalam pembentukan pohon Huffman. Masing-masing bahasa diuji dengan tiga variasi panjang dokumen seperti, Pendek (sekitar 100 kata), Menengah (300–500 kata), Panjang (>1000 kata).

2.3 Langkah-Langkah Penelitian

a. Pengumpulan dan Preprocessing Data
   1. Mengambil sampel teks dari sumber daring (berita, artikel, dan e-book)
   2. Normalisasi teks: penghapusan tanda baca, konversi ke huruf kecil, dan penghapusan karakter non-alfabet (untuk menjaga konsistensi distribusi karakter)

ISSN: 3110-0864 (Media Online)
JURNAL ILMU KOMPUTER DAN TEKNIK INFROMATIKA
Volume 1 No 1 Januari 2025 | pp: 14-20
https://journals.raskhamedia.or.id/index.php/juikti
DOI: https://doi.org/10.64803/juikti.v1i1.44.

b. Analisis Frekuensi Karakter
   1. Menghitung frekuensi kemunculan tiap karakter dalam teks menggunakan skrip Python
   2. Menyusun daftar simbol dan frekuensinya sebagai input awal untuk konstruksi pohon Huffman
c. Konstruksi Pohon Huffman
   1. Menggunakan algoritma Huffman klasik (*static Huffman coding*)
   2. Pohon biner dibentuk dengan pendekatan *greedy*, di mana dua simpul dengan frekuensi terendah digabungkan secara rekursif hingga membentuk satu pohon utama
d. Proses Encoding dan Kompresi
   1. Melakukan encoding terhadap teks awal menggunakan kode biner hasil pohon Huffman
   2. Menghitung total panjang bit hasil encoding
   3. Dibandingkan dengan ukuran asli (dalam bit) berdasarkan ASCII (8-bit per karakter)
e. Perhitungan Rasio Kompresi
   1. Rumus:

$$\text{Rasio Kompresi} = \left(\frac{\text{Terkouran Asli - Ukuran Terkompresi}}{\text{Ukuran Asli}} \right) x100\% \quad (1)$$

   2. Rasio kompresi dihitung untuk masing-masing variasi teks dan bahasa
f. Analisis dan Interpretasi
   1. Menafsirkan hasil simulasi dengan membandingkan efisiensi antar sampel
   2. Menganalisis pengaruh distribusi karakter terhadap hasil akhir
   3. Mengidentifikasi pola atau keterbatasan algoritma berdasarkan data

2.4 Alat dan Lingkungan Simulasi

Simulasi dan komputasi dilakukan menggunakan:

a. Bahasa pemrograman: Python 3.10
b. Perpustakaan: collections, heapq, matplotlib (untuk visualisasi pohon dan hasil)
c. Editor: Jupyter Notebook dan Google Colab

2.5 Validitas dan Keandalan

Validitas data dijaga dengan:
a. Menggunakan data teks nyata dari sumber terpercaya
b. Melakukan pengujian berulang pada masing-masing sampel untuk memastikan kestabilan hasil
c. Membandingkan hasil simulasi manual dengan output library kompresi standar sebagai acuan (*benchmarking*)

3. HASIL DAN PEMBAHASAN

Simulasi dilakukan untuk mengukur kinerja algoritma Huffman dalam mengompresi teks berbahasa Indonesia dan Inggris. Masing-masing bahasa diuji pada tiga jenis teks berdasarkan panjangnya: pendek (sekitar 700–750 karakter), menengah (±1800 karakter), dan panjang (>3000 karakter). Semua teks diasumsikan dalam format ASCII dengan representasi tetap 8 bit per karakter sebagai baseline sebelum dikompresi. Setelah dikompresi menggunakan pohon Huffman berbasis frekuensi karakter, dilakukan pengukuran ulang terhadap jumlah bit dan dihitung rasio efisiensi kompresinya.

3.1 Hasil Simulasi

Data hasil kompresi ditampilkan dalam tabel berikut:

Tabel 1. Hasil Simulasi Kompresi Teks Menggunakan Algoritma Huffman

No	Bahasa	Jenis Teks	Jumlah Karakter	Ukuran Asli (bit)	Ukuran Setelah Kompresi (bit)	Rasio Kompresi (%)
1	Indonesia	Pendek	720	5760	3860	32.99
2	Indonesia	Menengah	1800	14400	9140	36.53
3	Indonesia	Panjang	3900	31200	17900	42.63
4	Inggris	Pendek	750	6000	3920	34.67

ISSN: 3110-0864 (Media Online)
JURNAL ILMU KOMPUTER DAN TEKNIK INFROMATIKA
Volume 1 No 1 Januari 2025 | pp: 14-20
https://journals.raskhamedia.or.id/index.php/juikti
DOI: https://doi.org/10.64803/juikti.v1i1.44.

5	Inggris	Menengah	1750	14000	8950	36.07
6	Inggris	Panjang	4100	32800	18500	43.60

3.2 Analisis Per Bahasa

a. Teks Berbahasa Indonesia
   Hasil menunjukkan bahwa semakin panjang teks, semakin tinggi efisiensi kompresi yang diperoleh. Untuk teks pendek, algoritma Huffman mampu mengurangi ukuran data sebesar 32,99%. Angka ini meningkat menjadi 42,63% untuk teks panjang. Hal ini disebabkan oleh semakin banyaknya karakter unik yang dianalisis dan semakin kaya distribusi frekuensi, sehingga kode biner hasil Huffman lebih optimal. Pada teks pendek, distribusi frekuensi karakter cenderung masih merata. Misalnya, huruf vokal dan konsonan memiliki perbedaan jumlah kemunculan yang belum terlalu signifikan. Namun, pada teks yang lebih panjang, pola kemunculan huruf seperti ‘a’, ‘n’, ‘e’, dan ‘i’ mulai mendominasi dan dapat diberi kode yang sangat pendek. Inilah yang menyebabkan ukuran file setelah kompresi menurun secara signifikan.
b. Teks Berbahasa Inggris
   Pola serupa juga terjadi pada teks bahasa Inggris. Rasio kompresi meningkat dari 34,67% (teks pendek) menjadi 43,60% (teks panjang). Bahasa Inggris cenderung memiliki distribusi karakter yang lebih tidak merata dibandingkan bahasa Indonesia, misalnya huruf ‘e’, ‘t’, dan ‘a’ yang sering kali mendominasi teks panjang. Ini membuat algoritma Huffman mampu membentuk kode biner yang lebih kompak untuk karakter dominan tersebut. Namun, untuk teks pendek, struktur kalimat bahasa Inggris yang banyak menggunakan simbol, artikel pendek ("a", "an", "the") dan huruf kapital cenderung mengurangi efektivitas Huffman karena meningkatkan jumlah simbol unik yang harus diproses di awal.

3.3 Perbandingan Antara Bahasa

Jika dibandingkan antara dua bahasa, teks berbahasa Inggris cenderung menunjukkan efisiensi kompresi yang sedikit lebih tinggi pada teks panjang, yaitu 43,60% dibandingkan 42,63% pada bahasa Indonesia. Namun, selisih ini tidak terlalu signifikan. Hal ini menunjukkan bahwa karakteristik frekuensi simbol kedua bahasa sebenarnya tidak berbeda jauh, setidaknya dalam konteks panjang teks dan sumber yang diuji dalam simulasi ini. Namun, bahasa Indonesia yang lebih fleksibel secara struktur kata (terutama karena afiksasi dan bentuk dasar) menghasilkan variasi karakter yang lebih banyak pada teks pendek dan menengah. Ini sedikit menurunkan efisiensi Huffman pada ukuran teks yang lebih kecil.

3.4 Interpretasi Rasio Kompresi

Rasio kompresi tertinggi terjadi pada teks panjang berbahasa Inggris (43,60%) dan Indonesia (42,63%).
Hal ini mengindikasikan bahwa algoritma Huffman ideal digunakan pada data yang:
a. Memiliki distribusi karakter yang sangat tidak merata
b. Memiliki ukuran yang besar sehingga analisis frekuensi lebih stabil
c. Mengandung banyak karakter berulang

Sebaliknya, pada teks pendek yang lebih homogen atau acak distribusinya, algoritma ini kurang mampu menghasilkan efisiensi maksimal karena keterbatasan data dalam membentuk struktur pohon Huffman yang optimal.

3.5 Pembahasan Mengenai Struktur Kode Huffman

Dalam salah satu uji kasus pada teks panjang berbahasa Indonesia, kode Huffman yang terbentuk menunjukkan bahwa karakter ‘a’, ‘n’, dan ‘e’ mendapatkan panjang kode biner antara 2 hingga 3 bit, sementara karakter seperti ‘q’, ‘z’, dan simbol asing diberikan kode sepanjang 6 hingga 7 bit. Ini mencerminkan mekanisme greedy dari algoritma Huffman yang secara otomatis mengalokasikan bit paling pendek pada karakter terpopuler. Struktur pohon Huffman yang terbentuk menunjukkan hierarki yang tidak simetris, mencerminkan distribusi karakter yang tidak merata. Ini adalah kekuatan utama Huffman dibanding metode fixed-length seperti ASCII yang menggunakan 8 bit untuk semua karakter tanpa memperhatikan frekuensi kemunculan.

3.6 Kelebihan dan Keterbatasan Huffman dari Simulasi

Kelebihan:

ISSN: 3110-0864 (Media Online)
JURNAL ILMU KOMPUTER DAN TEKNIK INFROMATIKA
Volume 1 No 1 Januari 2025 | pp: 14-20
https://journals.raskhamedia.or.id/index.php/juikti
DOI: https://doi.org/10.64803/juikti.v1i1.44.

a. Kompresi lossless yang menjamin integritas data
b. Sangat efisien pada teks panjang dan tidak merata distribusinya
c. Implementasi relatif sederhana dengan algoritma *greedy*

Keterbatasan:
a. Tidak cocok untuk kompresi real-time karena butuh analisis frekuensi seluruh data di awal
b. Kurang efisien pada data dengan distribusi karakter merata
c. Ukuran kode Huffman perlu disimpan sebagai metadata saat digunakan untuk kompresi nyata

3.7 Implikasi Temuan

Temuan ini memberikan dasar kuat bahwa algoritma Huffman layak digunakan untuk kompresi teks dokumen panjang seperti buku elektronik, arsip berita, dan dokumen administratif. Dalam konteks aplikasi nyata seperti penyimpanan arsip pemerintah atau perpustakaan digital, efisiensi 40% lebih pengurangan ukuran file dapat menghemat kapasitas penyimpanan dalam skala besar. Selain itu, hasil ini juga menunjukkan pentingnya mempertimbangkan karakteristik data sebelum memilih metode kompresi. Huffman cocok untuk data dengan banyak pengulangan dan pola dominan, tapi kurang efektif jika digunakan pada data mentah yang pendek atau simbolik (seperti log file sistem atau kode program acak).

4. KESIMPULAN

Penelitian ini telah membuktikan bahwa algoritma Huffman merupakan salah satu metode kompresi data teks yang efektif dan efisien, khususnya dalam konteks kompresi lossless. Berdasarkan hasil simulasi yang dilakukan terhadap teks dalam dua bahasa (Indonesia dan Inggris) dengan variasi panjang dokumen, ditemukan bahwa algoritma Huffman mampu mereduksi ukuran data secara signifikan dengan rasio kompresi berkisar antara 32% hingga lebih dari 43%. Efisiensi tertinggi dicapai pada teks berukuran panjang dengan distribusi karakter yang tidak merata. Dari sisi performa, algoritma Huffman bekerja optimal pada data yang memiliki pola kemunculan karakter yang berulang dan dominan. Distribusi frekuensi karakter menjadi faktor krusial dalam pembentukan pohon Huffman dan penentuan panjang kode biner. Semakin banyak karakter dominan, semakin padat hasil kompresinya. Sebaliknya, pada data pendek atau karakter yang frekuensinya relatif merata, efisiensi kompresi menjadi lebih rendah. Studi ini juga menegaskan pentingnya pendekatan simulatif dalam memahami batas operasional dan optimalisasi algoritma. Temuan ini memiliki implikasi nyata dalam penerapan Huffman pada sistem penyimpanan dokumen, pengarsipan digital, dan distribusi data berbasis teks. Dengan demikian, algoritma Huffman layak dipertimbangkan sebagai solusi kompresi dalam berbagai implementasi teknologi informasi, selama konteks datanya sesuai. Ke depan, integrasi Huffman dengan teknik adaptif atau algoritma hibrida dapat menjadi arah pengembangan selanjutnya untuk meningkatkan efisiensi pada data real-time dan heterogen.

REFERENCES

[1] F. Ananda Lubis, P. Studi Manajemen, F. Ekonomi Dan Bisnis Islam, and M. Irwan Padli Nasution, "Penggunaan Teknologi Big Data untuk Analisis Prediksi Bisnis," *J. Ilm. Nusant. (JINU)*, vol. 1, no. 4, pp. 3047–9673, 2024, [Online]. Available: https://doi.org/10.61722/jinu.v1i4.1882

[2] M. Aslyza and M. I. P. Nasution, "Manajemen Data Berbasis Database : Solusi Untuk Penyimpanan Dan Akses Data Yang Lebih Efisien," *J. Ilm. Nusant.*, vol. 2, no. 3, pp. 909–917, 2025.

[3] I. M. Sianturi, "Perancangan Aplikasi Kompresi File Gambar Dengan Menggunakan Algoritma Stout Code," *J. Pelita Ilmu Pendidik.*, vol. 2, no. 1, pp. 19–29, 2024, doi: 10.69688/jpip.v2i1.57.

[4] I. Arizki, A. Triawan, and F. Zayid, "Penerapan Algoritma Lempel Ziv Welch (LZW) Untuk Kompresi Data," *TeknoIS J. Ilm. Teknol. Inf. dan Sains*, vol. 14, no. 1, pp. 66–73, 2024, doi: 10.36350/jbs.v14i1.232.

[5] A. Hanif, E. Wahyudi, H. Adianto, and L. Martanto, "Komparasi Performa Algoritma Kompresi Data Lossless," *J. Inf. Technol.*, no. 204, pp. 148–157, 2023.

[6] A. Serdano, C. Azzahra, D. Alpamah, I. F. Batubara, and M. W. Qasthari, "Penerapan Algoritma Huffman Code Untuk Kompresi Dan Dekompresi Pesan," *J. Comput. Sci. Informatics Eng.*, vol. 3, no. 4, pp. 161–169, 2024, doi: 10.55537/cosie.v3i4.949.

[7] R. Saputra, Y. Reswan, and Y. Darmi, "Fast Method Of Image File Compression In Web-Based Application Form Using Huffman Algorithm Metode Cepat Kompresi File Citra Pada Form Aplikasi Berbasis Web Menggunakan Algoritma Huffman," *J. Kom.*, vol. 3, no. 2, pp. 285–294, 2023, [Online]. Available: https://doi.org/10.53697/jkomitek.v3i2

ISSN: 3110-0864 (Media Online)
JURNAL ILMU KOMPUTER DAN TEKNIK INFROMATIKA
Volume 1 No 1 Januari 2025 | pp: 14-20
https://journals.raskhamedia.or.id/index.php/juikti
DOI: https://doi.org/10.64803/juikti.v1i1.44.

[8] M. Lutfitasari and A. Munandar, "Dampak Arus Kas Bebas, Kebijakan Utang dan Ukuran Perusahaan terhadap Kinerja Keuangan," *J. Ilm. MEA*, vol. 6, no. 2, pp. 1021–1037, 2022.

[9] M. Hasda, M. Winario, H. Hidayat, and M. Zaim, "Penerapan Strategi Bisnis Berkelanjutan Sesuai Dengan Prinsip Syariah Di Dhuafa Mart," *J. Community Serv. Empower.*, vol. 1, no. 1, pp. 23–29, 2024, doi: 10.69693/jcse.v1i1.25.

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

[11] Y. Reswan and D. Agung Prabowo, "Implementasi Kode Huffman Dalam Aplikasi Kompresi Text Pada Layanan Sms," *J. Rekayasa Sist. Inf. dan Teknol.*, vol. 1, no. 1, pp. 9–16, 2023, doi: 10.59407/jrsit.v1i1.73.

[12] D. Maulana and A. Rahmatulloh, "Analisis Kinerja Kompresi Huffman dan BWT pada Steganografi dalam Reduksi Ukuran Gambar," *J. Inform. dan Multimed.*, vol. 17, no. 1, pp. 11–17, 2025, doi: 10.33795/jtim.v17i1.6597.

[13] S. M. Simanjuntak, "Analisis Perbandingan Kompresi File Audio Menggunakan Algoritma Shannon Fano Dengan Algoritma Fibonacci Code," *J. Kaji. Ilm. Teknol. Inf. dan Komput.*, vol. 2, no. 1, pp. 1–10, 2024, doi: 10.62866/jutik.v2i1.110.

[14] M. M. Karuntu, D. P. . Saerang, and J. B. Maramis, "Pendekatan Grounded Teori: Sebuah Kajian Prinsip, Prosedur, Dan Metodologi," *J. EMBA J. Ris. Ekon. Manajemen, Bisnis dan Akunt.*, vol. 10, no. 2, pp. 1070–1081, 2022, doi: 10.35794/emba.v10i2.41425.

[15] A. Agung, A. Daniswara, I. Kadek, and D. Nuryana, "Data Preprocessing Pola Pada Penilaian Mahasiswa Program Profesi Guru," *J. Informatics Comput. Sci.*, vol. 05, pp. 97–100, 2023.