

# Implementasi Algoritma Multi Group Huffman Untuk Kompresi File Video

Nisa Nurjanah

Fakultas Ilmu Komputer Dan Teknologi Informasi, Teknik Informatika, Universitas Budi Darma Medan Indonesia  
Email: chajannah@gmail.com

**Abstrak**—Teknologi berkembang sangat pesat seiring dengan kebutuhan masyarakat dalam memperoleh informasi secara cepat. Teknologi informasi dan komunikasi sudah menjadi bagian penting dalam mengolah data. Aktifitas yang dulunya manual, sekarang dapat diproses dengan menggunakan komputer sehingga dapat menghemat waktu dan tingkat ketelitian yang dihasilkan jauh lebih tinggi. Salah satu contoh dari penggunaan komputer adalah pengolahan data file video, file video ini memiliki ukuran yang cukup besar sehingga membutuhkan media penyimpanan yang besar, untuk itu kompresi data menjadi sangat penting karena memperkecil kebutuhan penyimpanan data, mempercepat pengiriman data. Maka penulis mengembangkan suatu sistem yang baru sebagai alternatif dengan bantuan komputer yang telah dilengkapi dengan sistem. Penulis juga menggunakan pemograman *Microsoft Visual Studio 2010* sebagai pengolah data dalam proses kompresi. Dalam penelitian dilakukan proses implementasi dengan menggunakan metode ini untuk proses kompresi file video, metode ini juga akan membantu meringankan tantangan dari penggunaan file video ketika menyimpan atau mendistribusikan dalam kapasitas yang besar, hal ini tentu akan sangat membebani komputer dalam kapasitas memori, bahkan saat pendistribusian menjadi sangat lambat dan tidak efektif.

**Kata Kunci:** Implementasi, Teknik Multi Group *Huffman* Untuk Kompresi *File Video*

**Abstract**—Technology is developing very rapidly along with the needs of society in obtaining information quickly. Information and communication technology has become an important part in processing data. Activities that were once manual, can now be processed using a computer so that it can save time and the resulting level of accuracy is much higher. One example of using a computer is processing video file data, this video file has a size large enough to require large storage media, for that data compression is very important because it reduces data storage requirements, speeding up data transmission. So the author developed a new system as an alternative with the help of a computer that has been equipped with the system. The author also uses *Microsoft Visual Studio 2010* programming as a data processor in the compression process. In the research, the implementation process was carried out using this method for the video file compression process, this method will also help alleviate the challenges of using video files when storing or distributing in large capacities, this of course will greatly burden the computer in memory capacity, even when distribution becomes difficult. very slow and ineffective.

**Keywords:** Implementation, Huffman Multi Group Technique For Video File Compression

## 1. PENDAHULUAN

Teknologi berkembang sangat pesat seiring dengan kebutuhan masyarakat dalam memperoleh informasi secara cepat. Teknologi informasi dan komunikasi sudah menjadi bagian penting dalam mengolah data. Aktifitas yang dulunya manual, sekarang dapat diproses dengan menggunakan komputer sehingga dapat menghemat waktu dan tingkat ketelitian yang dihasilkan jauh lebih tinggi. Salah satu contoh dari penggunaan komputer adalah pengolahan data *file video*, *file video* ini memiliki ukuran yang cukup besar sehingga membutuhkan media penyimpanan yang besar, untuk itu kompresi data menjadi sangat penting karena memperkecil kebutuhan penyimpanan data, mempercepat pengiriman data. Kompresi file video adalah teknik memperkecil ukuran video dengan cara melakukan kompresi terhadap gambar dan suara yang terdapat didalam video tersebut, hingga batas ambang kualitas yang masih dikatakan baik.

Kompresi *file* adalah sebuah cara untuk memampatkan *file* sehingga ruang yang diperlukan untuk menyimpan *file* menjadi lebih kecil dan efisien dalam penyimpanannya, serta mempersingkat waktu pertukaran *file*. Cara ini memungkinkan *file* disimpan dengan kapasitas kecil dan dikembalikan seperti semula. Teknik kompresi file dibedakan menjadi 2, yaitu *Lossy Data-Compression* dan *Lossless Data-Compression*. *Lossless Data-Compression* yaitu proses kompresi data yang dapat dikembalikan ke data asli secara utuh tanpa adanya informasi yang hilang, sedangkan *Lossy Data-Compression* yaitu proses kompresi data yang sudah dikompresi tidak sama seperti data aslinya[1]–[3].

Dalam penelitian sebelumnya yang dilakukan oleh Monica Borda terhadap kompresi dengan judul *Fundamentals in Information Theory and Coding*, menyebutkan bahwa jenis pengkodean ini digunakan untuk sumber yang menghasilkan aliran karakter milik kelas disjungtif, misalnya sumber menghasilkan huruf burst diikuti oleh angka dan kemudian oleh ruang kosong. Saat merancang encoder seperti itu, pohon diatur untuk setiap kelas simbol. Simbol kegagalan kemudian ditambahkan ke setiap pohon. Setiap karakter diberi kode dengan mempertimbangkan pohon *Huffman* saat ini. Dalam kasus karakter dapat ditemukan di pohon tertentu, kode yang sesuai ditransmisikan. Jika tidak, kode yang sesuai dengan karakter kegagalan ditransmisikan menuju ke pohon lain. Aplikasi lanjutan dari tipe ini dapat ditemukan di driver sistem database[4]. Kompresi *multi-group* lebih unggul daripada pengkodean *Huffman*, memberikan peningkatan rata-rata 25% dalam rasio kompresi dibandingkan dengan metode *Huffman*. Memperluas pendekatan desain bagian 2 ke teknik *multi-group* menyediakan alat cepat dengan efisiensi kompresi tinggi yang ideal untuk transmisi data dan penyimpanan data basis data besar.

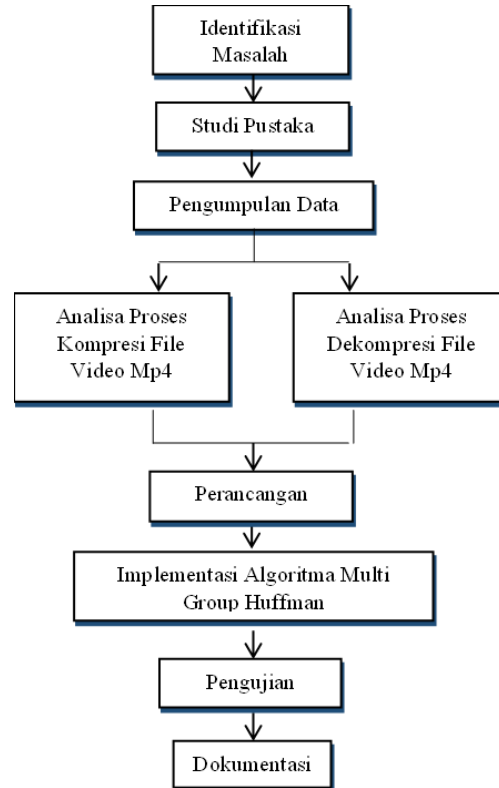
Berdasarkan keterangan tersebut peneliti hendak mengimplementasikan metode ini untuk kompresi file video, karena dengan menerapkan metode ini akan membantu mengurangi penggunaan kapasitas data. Berdasarkan uraian

tersebut, penulis mengangkat penelitian ini dengan judul Implementasi Algoritma *Multi-Group Huffman* Untuk Kompresi File Video.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Berikut ini merupakan gambaran tahapan penelitian, yang dapat dilihat pada gambar 1 dibawah ini:



**Gambar 1.** Kerangka Penelitian

Berikut keterangan dari gambaran tahapan penelitian diatas:

1. Identifikasi Masalah  
Mengidentifikasi/merumuskan masalah merupakan langkah awal dalam menemukan suatu masalah yang terfokus pada suatu tujuan.
2. Studi Pustaka  
Pada tahapan ini dilakukan pengumpulan data, data yang dikumpulkan berupa judul tugas akhir pada jurusan informatika yang sudah di *pulish*.
3. Pengumpulan Data  
Pengumpulan data, kegiatan pengumpulan data dapat dilakukan dengan melakukan wawancara stakeholder dan observasi langsung di lokasi penelitian Penerapan Metode.
4. Analisa Proses Kompresi dan Dekompresi *File* Video Mp4  
Pada tahapan ini dilakukan analisa pada *file* Video Mp4 yang telah ditentukan dengan proses analisa kompresi dan setelah itu analisa dekompresi.
5. Perancangan  
Perancangan sistem akan dilakukan dengan merancang atau mendesign suatu aplikasi menggunakan tools pemrograman yaitu *Microsoft Visual Basic 2008..*
6. Implementasi  
Pada tahapan ini digunakan sebagai prosedur melengkapi rancangan terdapat dalam file video.
7. Hasil Pengujian  
Hasil pengujian adalah langkah terakhir dalam menarik kesimpulan tentang nilai hasil yang diperoleh ketika menggunakan metode tertentu menggunakan aplikasi.

### 2.2 Kompresi Data

Proses kompresi merupakan proses mereduksi ukuran suatu data untuk menghasilkan representasi digital yang padat atau mampat (*compact*) namun tetap dapat mewakili kuantitas informasi yang terkandung pada data tersebut. Pada citra, video

dan audio, kompresi mengarah pada meminimasi jumlah *bit rate* untuk representasi digital. Pada beberapa literatur, istilah kompresi sering disebut juga *source coding*[5]. Tujuan kompresi data adalah untuk mengurangi atau mereduksi ukuran suatu data sehingga ukuran data menjadi lebih kecil dan lebih ringan dalam proses transmisi. Kompresi data merepresentasikan data digital dengan sesedikit bit, akan tetapi juga tetap mempertahankan kebutuhan minimum untuk membentuk kembali data aslinya[6].

### 2.2.1 Teknik Kompresi

Teknik kompresi data dapat diklarifikasikan menjadi dua yaitu *lossless compression* dan *lossy compression*[7]:

a. *Lossless Compression*

*Lossless* merupakan teknik kompresi data yang menghasilkan data yang identik atau sama dengan data aslinya. Kompresi data *lossless* digunakan dalam berbagai aplikasi seperti format ZIP dan GZIP. Beberapa format gambar seperti BMP (*Bitmap*), PNG (*Portable Network Graphics*) atau GIF (*Graphics Interchange Format*) hanya menggunakan kompresi *lossless*.

b. *Lossy Compression*

*Lossy* kompresi adalah suatu metode untuk mengkompresi data dan dekompresi data, dimana data mungkin berbeda dari yang aslinya tetapi tidak terlalu kelihatan perbedaannya. Format kompresi *lossy* mengalami *generation loss*, yaitu jika dilakukan secara berulang kali kompresi dan dekompresi file akan menyebabkan kehilangan kualitas citranya [8].

Ada tiga parameter yang umum digunakan untuk mengetahui hasil kompresi berdasarkan algoritma yang digunakan[9], yaitu[10]:

1. *Ration of Compression* ( $R_c$ )

Rasio kompresi (*ration of compression*) adalah suatu parameter kompresi untuk mengetahui hasil dari sebuah kompresi dengan membandingkan hasil data sebelum dan sesudah dikompresi.

$$R_c = \frac{\text{Ukuran bit data sebelum dikompresi}}{\text{Ukuran bit data setelah dikompresi}} \dots\dots\dots (1)$$

2. *Compression Ratio* ( $C_r$ )

*Compression ratio* ( $C_r$ ) merupakan persentase perbandingan antara data yang sudah dikompresi dengan data yang belum dikompresi.

$$C_r = \frac{\text{Ukuran bit data setelah dikompresi}}{\text{Ukuran bit data sebelum dikompresi}} \times 100\% \dots\dots\dots (2)$$

3. *Redudancy* ( $R_d$ )

*Redudancy* ( $R_d$ ) merupakan kelebihan yang terdapat didalam data sebelum dikompresi. Hasil dari *redudancy* yaitu selisih antara ukuran data sebelum dikompresi dengan hasil *compression ratio*.

$$R_d = 100\% - C_r \dots\dots\dots (3)$$

### 2.2 Algoritma Multi Group Huffman

Teknik ini membantu dalam kasus di mana ada lebih dari satu sumber yang menghasilkan pola untuk dikodekan. Ini dapat menghasilkan serangkaian huruf, diikuti oleh serangkaian angka dan ini pada gilirannya dapat diikuti oleh serangkaian spasi. Dengan demikian, data dari berbagai jenis[11]. Masing-masing jenis dikodekan sebagai pohon biner yang terpisah. Kode yang akan diterjemahkan diterjemahkan di pohon biner. Jika pohon biner berisi kode itu, maka kode yang setara adalah keluaran. Jika pohon biner tidak mengandung kode tertentu, maka nilai kesalahan pohon biner adalah *output*. Ini merupakan indikasi bahwa kode tertentu tidak ada di pohon biner dan pohon yang berbeda perlu dicari. Setiap pohon biner memiliki nilai tambahan yang terkait dengannya yang disebut nilai kesalahan. Ini adalah output dalam kasus di mana kode tertentu yang dicari tidak ada di pohon[12].

Kompresi multigroup lebih unggul daripada pengkodean *Huffman*, memberikan peningkatan rata-rata 25% dalam rasio kompresi dibandingkan metode *Huffman*. Memperluas pendekatan desain bagian 2 ke teknik multigroup menyediakan alat cepat dengan efisiensi kompresi tinggi yang ideal untuk transmisi data dan penyimpanan data basis data besar. Pertama-tama kita akan menjelaskan ide dasar teknik multigroup. Misalnya pertimbangkan set 7 karakter yang disederhanakan: X, Y, Z, 1, 2, 3, b (di mana b menunjukkan karakter kosong)[13]. Tabel 1 memberikan probabilitas kemunculan karakter-karakter ini dan skema Huffman untuk pengkodeannya.

**Tabel 1.** Serangkaian yang disederhanakan dan kode Huffman-nya

Character	Probability	Huffman's Code
X	0,1	000
Y	0,1	0011
Z	0,1	0010
1	0,3	01
2	0,1	101
3	0,1	100
b	0,2	11

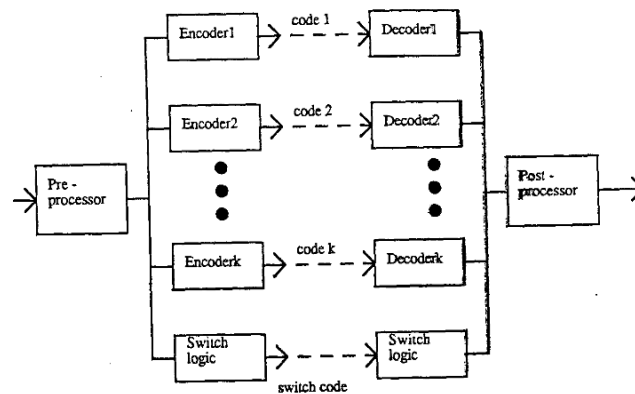
**Tabel 2.** Kode Lokal dari set yang disederhanakan

Group	Character	Local Code
-------	-----------	------------

Alpha	X	01
	Y	11
	Z	10
Digit	A	00
	1	0
	2	101
Blank	3	100
	A	11
	B	1
	&	0

**Tabel 3.** Beralih Kode dari Himpunan yang disederhanakan

Group	Character	Local Code
Alpha	1	Digit
	0	Blank
Digit	1	Alpha
	0	Blank
Blank	1	Digit
	0	Alpha



**Gambar 2.** Perangkat Keras Untuk Kompresi Multigroup

### 2.3 File MPEG (Moving Picture Expert Group)-4 (MP4)

Pada awalnya MPEG Video Layer-4 banyak dipakai oleh para pengguna komputer. *File-file* MPEG video layer-4 disimpan dengan ekstensi nama *file* MP4[14]. Kemudian MPEG Audio Layer-4 selanjutnya banyak dikenal sebagai MP4. *File* MPEG terdiri dari bagian kecil yang disebut *frame*. Biasanya tiap *frame* dapat berdiri sendiri. Tiap *frame* memiliki *header* yang berisi informasi *frame* tersebut. Pada *file* MPEG tidak ada *header file*. Lain halnya pada MP3, beberapa *frame* bisa merupakan bagian yang saling bergantung[15].

## 3. HASIL DAN PEMBAHASAN

### 3.1 Analisa Proses Kompresi dan Dekompresi File Video MPEG-4

Algoritma *Multi-Group Huffman* adalah algoritma kompresi yang mengubah bentuk file awal menjadi file yang lain dan berjenis kompresi *loosy*. Untuk hasil kompresi yang dapat dijalankan dengan aplikasi pembukanya dan ada juga yang tidak. Pada pembagian analisa ini file hasil kompresi video mp4 yang akan diubah menjadi file lain tidak akan dapat dijalankan sebagaimana fungsi file video mp4. Karena bagian analisa ini hanya bertujuan untuk melakukan kompresi atau mengubah ukuran file menjadi kecil dengan teknik yang digunakan algoritma *Multi-Group Huffman*. Pada pembagian file video mp4 akan disesuaikan dengan struktur file video mp4, file video tersebut terdiri atas beberapa bagian, yaitu satu bagian file video mp4 atau bagian yang disebut *Raw Header* dan dua bagian byte data video file video mp4 sebagai *Data Conten*, jadi semua ada tiga bagian grup yang akan dikompresi dengan algoritma *Multi-Group Huffman*. Dalam melakukan analisa dengan tepat disediakan satu sampel file video mp4, berikut sampel file dalam bentuk gambar yang



spasi	175
6	15
1	9
4	8
7	4
8	4
2	3
9	3
C	2
A	2
F	2
B	2
D	2
5	1
3	1
E	1

**Tabel 5.** Struktur Biner Pohon Huffman

Character	Frequency	Assignment	Space Savings
0	364	0	$364 \times 8 - 364 \times 1 = 2548$ bits
space	287	11	$287 \times 8 - 287 \times 2 = 1722$ bits
1	40	1000	$40 \times 8 - 40 \times 4 = 160$ bits
2	20	101111	$20 \times 8 - 20 \times 6 = 40$ bits
5	16	101101	$16 \times 8 - 16 \times 6 = 32$ bits
9	15	101100	$15 \times 8 - 15 \times 6 = 30$ bits
7	14	101011	$14 \times 8 - 14 \times 6 = 28$ bits
b	14	101000	$14 \times 8 - 14 \times 6 = 28$ bits
e	14	101001	$14 \times 8 - 14 \times 6 = 28$ bits
6	13	100111	$13 \times 8 - 13 \times 6 = 26$ bits
d	12	100110	$12 \times 8 - 12 \times 6 = 24$ bits
a	11	100101	$11 \times 8 - 11 \times 6 = 22$ bits
8	10	100100	$10 \times 8 - 10 \times 6 = 20$ bits
3	10	1011101	$10 \times 8 - 10 \times 7 = 10$ bits
4	9	1010101	$9 \times 8 - 9 \times 7 = 9$ bits
f	9	1011100	$9 \times 8 - 9 \times 7 = 9$ bits
c	5	1010100	$5 \times 8 - 5 \times 7 = 5$ bits

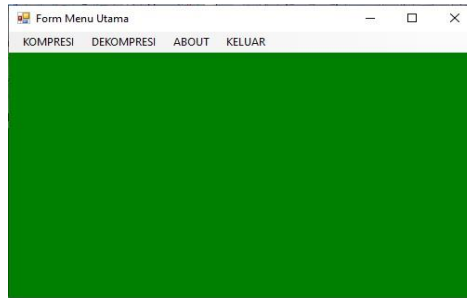
### 3.2 Hasil Pengujian

Untuk mendapatkan hasil dari penelitian ini maka dilakukan perhitungan menggunakan alat bantu yaitu *software* Microsoft Visual Basic 2008. Pada bagian ini merupakan hasil dari pengujian rancangan berupa tampilan layar aplikasi yang sedang berjalan. *Form* ini merupakan langkah awal sebelum masuk kedalam program, *Admin/user* harus memasukkan id dan password terlebih dahulu untuk masuk ke dalam program tersebut. Adapun tampilan form login dapat dilihat pada gambar dibawah ini:



**Gambar 6.** Form Login

*Form* Menu Utama merupakan *Form* yang berfungsi untuk memudahkan user untuk memanggil *form-form* yang telah dibuat dalam satu proyek aplikasi. Adapun tampilan form menu utama dapat dilihat pada gambar dibawah ini:



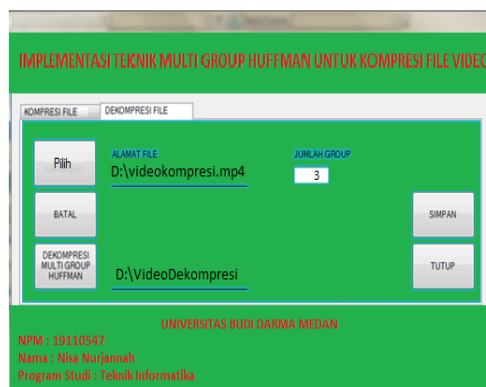
**Gambar 7.** *Form* Menu Utama

Pada proses kompresi maka yang awal dilakukan adalah memilih file video Mp4 yang akan dikompresi dan memilih lokasi penyimpanan yang akan digunakan untuk hasil kompresi. Adapun gambar proses tersebut dapat dilihat pada gambar dibawah ini:



**Gambar 8.** *Form* Kompresi

Pada form ini digunakan untuk proses dekompresi, hal yang pertama dilakukan adalah memilih file dokumen yang akan digunakan untuk menyimpan hasil dekompresi, selanjutnya diikuti dengan melakukan proses dekompresi terhadap file tersebut. Adapun gambar proses tersebut dapat dilihat pada gambar dibawah ini:



**Gambar 9.** *Form* Dekompresi

## 4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, hasil akhir penelitian ini mengemukakan beberapa kesimpulan penting dari pembahasan sebelumnya. Pertama, prosedur kompresi yang melibatkan algoritma multi group huffman telah berhasil dalam menjalankan proses kompresi file video berformat \*.mp4 dengan baik, memastikan bahwa teknik kompresi berjalan dengan efektif. Kedua, penerapan algoritma multi group huffman juga telah terbukti berhasil dalam mengompresi file gambar berukuran besar, mengubahnya menjadi ukuran yang lebih kecil. Terakhir, penelitian ini mencatat bahwa kompresi file video dapat diimplementasikan melalui aplikasi Microsoft Visual Basic Studio 2010 dengan memanfaatkan algoritma Multi group huffman, memberikan harapan bahwa hal ini akan mempermudah proses

kompresi ukuran file video di masa mendatang. Kesimpulan tersebut menggarisbawahi nilai penting algoritma multi group huffman dalam proses kompresi file video dan gambar, serta kemungkinan penggunaannya dalam pengembangan aplikasi kompresi di masa depan.

## REFERENCES

- [1] R. A. Sandra, "Implementasi Kombinasi Algoritma Tunstall Code Dan Boldi-Vigna Untuk Kompresi File Pdf," *J. Ilmu Komputer, Teknol. Dan Inf.*, vol. 1, no. 2, pp. 34–42, 2023.
- [2] P. S. Hasmita and C. F. Sianturi, "Implementasi Algoritma Punctured Elias Code Untuk Kompresi File Audio Pada Aplikasi Lagu Rohani," *Pelita Inform. Inf. dan Inform.*, vol. 10, no. 1, pp. 46–50, 2021.
- [3] D. Riyansyah, "Perancangan Aplikasi Kompresi File Video Menggunakan Algoritma Interpolative Coding," *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 3, no. 1, 2019.
- [4] S. Siahaan, "Penerapan Algoritma Sequitur Pada Kompresi Record Database Pada Database," *JURIKOM (Jurnal Ris. Komputer)*, vol. 6, no. 5, pp. 511–516, 2019.
- [5] J. M. B. Panjaitan, "Penerapan Algoritma Fibonacci Codes Pada Kompresi Aplikasi Audio Mp3 Berbasis Dekstop," *Bull. Multi-Disciplinary Sci. Appl. Technol.*, vol. 1, no. 1, pp. 27–33, 2021.
- [6] A. A. Pramadi, S. D. Nasution, and B. Purba, "Penerapan Algoritma Even-Rodeh Pada Aplikasi Kompresi File Gambar," *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 3, no. 1, 2019.
- [7] D. Cahayati, A. M. H. Pardede, and H. Khair, "Implementasi Algoritma Elias Gamma Kompresi Pada File Teks," *Algoritm. J. ILMU Komput. DAN Inform.*, vol. 6, no. 1, 2022.
- [8] K. Mahesa, "DEKOMPRESI PADA CITRA DIGITAL," *J. Process.*, vol. 12, no. 1, pp. 948–963, 2017.
- [9] H. Sartika, T. Zebua, and R. Parapat, "PERANCANGAN DAN IMPLEMENTASI ALGORITMA ELIAS GAMMA CODE UNTUK MENGKOMPRESI RECORD DATABASE PADA APLIKASI RANGKUMAN," *KOMIK*, vol. 3, pp. 259–265, 2019, doi: 10.30865/komik.v3i1.1600.
- [10] D. Pratiwi and T. Zebua, "ANALISIS PERBANDINGAN KINERJA ALGORITMA FIXED LENGTH BINARY ENCODING DAN ALGORITMA ELIAS GAMMA CODE DALAM KOMPRESI FILE TEKS," *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 3, no. 1, Nov. 2019, doi: 10.30865/KOMIK.V3I1.1623.
- [11] S. A. Abed, M. A. Ismail, and S. Khotijah, "Application of Huffman's Algorithm to Wave File Compression," *Int. J. Informatics Technol.*, vol. 1, no. 1, pp. 21–30, 2023.
- [12] H. Hendrik, "Kombinasi Algoritma Huffman dan Algoritma ROT 13 Dalam Pengamanan File Docx," *J. Inf. Syst. Res.*, vol. 2, no. 1, pp. 40–46, 2020.
- [13] R. N. Sihombing, "Implementasi Algoritma Multi Group Huffman Dalam Kompresi File Teks Dokumen," *Bul. Ilm. Inform. Teknol.*, vol. 1, no. 3, pp. 76–82, 2023.
- [14] S. M. Panjaitan, S. D. Nasution, and B. Purba, "Penerapan Algoritma Gopala-Hemachandra Code2 (GH-2 (n)) Pada Kompresi File Audio," *KOMIK (Konferensi Nas. Teknol. Inf. dan Komputer)*, vol. 4, no. 1, 2020.
- [15] I. Solin, "Perancangan Aplikasi Kompresi File Video Menggunakan Algoritma Transformasi Walsh Hadamard," *J. Glob. Technol. Comput.*, vol. 1, no. 2, pp. 63–70, 2022.