

Implementasi Algoritma Elias Gamma Code Untuk Kompresi File Video Pada Aplikasi Chatting Berbasis Android

Hendy Hermawan

Fakultas Ilmu Komputer dan Teknologi Informasi, Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia

Email: hendihermawan148@gmail.com

Abstrak—Pada perkembangan teknologi saat ini sangat begitu pesat sehingga banyaknya masyarakat membutuhkan informasi yang sangat begitu pesat. Salah satunya yaitu aplikasi chatting sebagai alat komunikasi yang dapat membantu berkomunikasi tanpa ada batasan waktu dan bebas dari hambatan. Tetapi dalam berkomunikasi melalui aplikasi chatting tidak bisa leluasa atau sebebas dalam berkomunikasi pada umumnya. Dimana aplikasi chatting memiliki jumlah bit yang telah ditetapkan, sehingga dalam pengiriman video pada aplikasi chatting tidak dapat sekaligus atau dalam sekali pengiriman melainkan dalam beberapa kali pengiriman. Adapun Solusi dalam permasalahan ini adalah bagaimana video tersebut dapat dikompresi guna untuk mempercepat proses pengiriman dan penyimpanan file video. Kompresi pada file video dilakukan dengan memperkecil ukuran video dengan proses mengurangi bit pada video, akan tetapi tidak menghilangkan data informasi didalamnya. Dalam penelitian ini, algoritma yang digunakan adalah Elias Gamma, dengan menggunakan metode tersebut, hasil kompresi dari nilai Elias Gamma mempunyai hasil yang berbeda-beda dari setiap nilainya, dan hasil kompresi akan menguntungkan dalam melakukan pengiriman, dan pemindahan file video akan semakin mudah.

Kata Kunci: Kompresi Video; Algoritma Elias Gamma; Aplikasi Chatting.

Abstract—In today's technology development is so fast that many people need information very rapidly. One of them is a chat application as a communication tool that can help communicate without time constraints and free from obstacles. But in communicating through chat applications, we cannot be as free or as free as in communicating in general. Where the chat application has a predetermined number of bits, so that the video delivery in the chat application cannot be sent all at once or in one delivery but in several times. The solution to this problem is how the video can be compressed in order to speed up the process of sending and storing video files. Compression of video files is done by reducing the size of the video by reducing the bits in the video, but not eliminating the information data in it. In this study, the algorithm used is Elias Gamma, by using this method, the compression results of the Elias Gamma n value have different results for each value, and the compression results will be beneficial in sending, and transferring video files will be easier.

Keywords: Video Compression; Elias Gamma Algorithm; Chatting Application.

1. PENDAHULUAN

Pada perkembangan teknologi saat ini begitu pesat dengan kebutuhan masyarakat dalam memperoleh informasi secara cepat. Banyaknya media yang dapat mempercepat informasi di saat ini. Salah satunya aplikasi *chatting* yang berperan penting terhadap komunikasi di kehidupan masyarakat. Aplikasi *chatting* sebagai alat komunikasi yang dapat membantu berkomunikasi tanpa ada batasan waktu dan bebas dari hambatan. Adapun masalah dalam berkomunikasi melalui aplikasi *chatting* disebabkan ketidak leluasanya atau sebebas dalam berkomunikasi pada umumnya. Dimana aplikasi *chatting* memiliki jumlah bit yang sudah ditetapkan, sehingga dalam pengiriman file pada aplikasi *chatting* tidak dapat sekaligus atau dalam sekali dalam pengiriman melainkan dalam beberapa kali pengiriman.

Dalam melakukan pengiriman *file video* pada aplikasi *chatting* sering terjadi masalah yaitu ukuran video yang lumayan besar sehingga *file video* yang akan dikirimkan tidak dapat dikirim si penerima. Untuk mengatasi masalah tersebut dapat menggunakan teknik pemampatan untuk pengurangan ukurannya. Salah satu kegunaan teknik pemampatan ialah untuk memperkecil kapasitas ruang dalam media penyimpanan. Agar data didalam ruang penyimpanan tersebut dapat ditata dengan baik. Adapun solusi pada masalah tersebut dengan melakukan proses teknik pemampatan *file video* yang besar menjadi ukuran yang lebih kecil atau bisa mengurangi ukuran bit yang terdapat pada setiap *file video*, akan tetapi tidak mempengaruhi data informasi didalamnya sehingga dapat menghemat wadah penyimpanan dan mempercepat proses pengiriman *file video* melalui aplikasi *chatting*.

Pada penelitian sebelumnya yang dilakukan oleh Rizky Syahputra pada tahun 2016 yang berjudul kompresi *file video mp4* dengan menggunakan metode *discrete cosine tranform* telah menyimpulkan bahwa penelitian terhadap *file video* di perlukan untuk memperkecil ukuran *file video* sehingga mempercepat proses penyimpanan dan pengiriman *file video* melalui aplikasi *chatting* [1].

Adapun penelitian yang dilakukan oleh Hikka Sartika dan Taronisokhi Zebua pada tahun 2019 terhadap algoritma *elias gamma code* yang berjudul perancangan dan implementasi algoritma *elias gamma code* untuk mengkompresi *record database* pada aplikasi rangkuman pengetahuan umum lengkap. Berdasarkan hasil pengujian yang dilakukan, maka disimpulkan bahwa algoritma *elias gamma code* mampu berdampak besar dalam menurunkan ukuran data dengan rata-rata compresi ratio 60 % [2].

Elias gamma code merupakan salah satu contoh *universal code* yang dipelopori oleh *peter elias*, yang mana hanya dapat mengkodekan bilangan bulat positif. *Elias gamma code* banyak digunakan ketika pengkodean *integer* yang mempunyai *upper bound* tidak dapat diperiksa sebelumnya. Dengan menerapkan algoritma *elias gamma code* diharapkan dapat digunakan untuk pemampatan *file video*, sehingga *file video* yang berukuran besar akan dikompresi menjadi ukuran yang lebih kecil dan proses pengiriman *file video* pada aplikasi *chatting* dapat dilakukan lebih cepat.

2. METODOLOGI PENELITIAN

2.1 Kompresi

Kompresi adalah proses mengkodekan informasi menggunakan bit atau *information-bearing* unit yang lain yang lebih rendah dari pada representasi data yang tidak terkodekan dengan suatu sistem *encoding* tertentu. Menurut Ida Mengyi Pu, kompresi data adalah ilmu atau seni mempresentasikan informasi dalam bentuk yang lebih *compact* [4]. Sedangkan menurut David Salomon, kompresi data adalah proses mengkonversikan sebuah input data *stream* (*stream* sumber atau data mentah asli) menjadi data *Stream* lainnya (*bitstream* hasil atau *stream* yang telah terkompresi) yang berukuran lebih kecil [5].

2.2 Algoritma Elias Gamma Code

Algoritma *elias gamma code* dibuat oleh Peter Elias. Tabel kode *elias gamma*, *elias* dibuat dengan menambah panjang kode dalam *Unary* (u). Kode berikutnya, E_γ ditambahkan pada panjang kode (M) dalam biner (β). Dengan demikian, *elias gamma code*, yang juga untuk bilangan bulat positif, sedikit lebih kompleks untuk dibangun. Aturan untuk mengkodekan sebuah bilangan dengan menggunakan *Elias Gamma* adalah sebagai berikut [8]

1. Rubah bilangan kode dalam bentuk biner.
2. Kurangkan 1 dari jumlah bit yang dipilih pada langkah pertama dan tambahkan sesuai dengan banyaknya bilangan nol.

Proses yang ekuivalen untuk menyatakan proses yang pada point nomor dua adalah sebagai berikut :

- a. Pisahkan *integer* menjadi pangkat 2 tertinggi (2) yang dapat dan ditampungnya sisakan digit biner N dari *integer* tersebut.
- b. Kodekan N dalam bentuk unary, jika N adalah nol maka diikuti oleh satu.
- c. Tambahkan sisa digit biner N yang telah dihasilkan.

Proses kompresi atau *encoding* suatu *integer* dengan berdasarkan *elias gamma* dilakukan dengan cara :

1. Tentukan nilai N untuk pangkat yang paling mendekati nilai n yang dituliskan sebagai $\beta(n)$. Nilai ini disebut sebagai unary code, dimana jumlah nilai N ditulis menjadi angka 0 dan diakhiri dengan angka 1
2. Dapatkan nilai L dengan mengurangi nilai n dengan nilai 2, nilai yang didapati diubah menjadi bilangan biner. Pembentukan kode *elias gamma code* dapat diambil sebuah contoh $n = 9$ Lalu temukan bilangan bulat N terbesar sehingga $2N \leq n < 2N + 1 = 23 \leq 9 < 23 + 1$ setelah ditemukan kode N terbesar lalu rubah nilai n menjadi biner lalu hilangkan 1 bit paling kiri $9 = 1001 \rightarrow 001$, kodekan dalam bentuk unary N sebagai 0 diikuti oleh 1 sehingga dihasilkan unary 3 $\rightarrow 0001$, lalu tambahkan sisa digit biner n dibelakang kode unary yang telah dihasilkan 0001001. Agar dapat lebih jelas berikut hasil pengkodean *elias gamma code* dapat dilihat pada tabel 1.

Tabel 1. Kode Elias Gamma

Index	Proses	Code Elias Gamma
1	$2^0 + 0$	1
2	$2^1 + 0$	010
3	$2^1 + 1$	011
4	$2^2 + 0$	00100
5	$2^2 + 1$	00101
6	$2^2 + 2 + 2$	00111
7	$2^3 + 3$	00111
8	$2^3 + 0$	0001000
9	$2^3 + 1$	0001001
10	$2^3 + 2$	0001010
11	$2^3 + 3$	0001011
12	$2^3 + 4$	0001100
13	$2^3 + 5$	0001101
14	$2^3 + 6$	0001110
15	$2^3 + 7$	001111
16	$2^4 + 0$	000010000
17	$2^4 + 1$	000010001
18	$2^4 + 2$	000010010

Sumber: D. Salomon, G. Motta, *Handbook of Data Compression*, 2010 [8]

Langkah berikutnya adalah mengkonversi karakter sesuai dengan kode *Elias Gamma*, sebelum dikonversi terlebih dahulu dilakukan pemeriksaan terhadap panjang *string bit* dengan langkah sebagai berikut :

1. Jika sisa bagi panjang *string bit* terhadap 8 adalah 0 maka tambahkan 00000001. Nyatakan dengan bit akhir.
2. Jika sisa bagi panjang *string bit* terhadap 8 adalah n (1, 2, 3, 4, 5, 6, 7) maka tambahkan 0 sebanyak $7 - n + "1"$ di akhir *string bit*. Nyatakan dengan L, lalu tambahkan bilangan biner dari $9 - n$. Nyatakan dengan bit akhir.

Proses Dekompresi/*decoding* dalam kode *elias gamma code* dapat dilakukan langkah-langkah, adalah:

1. Lakukan pembacaan pada 8 bit terakhir, hasil pembacaan berupa bilangan *decimal*. Nyatakan hasil pembacaan dengan n.
2. Hilangkan bit pada bagian akhir sebanyak $7 + n$.

Berikut ini adalah algoritma dekomposisi *Elias Gamma Code*, adalah:

1. Lakukan pembacaan *string bit* dari awal sampai angka 1 ditemukan. Catat posisi angka 1 dan nyatakan sebagai p. Nyatakan jumlah angka 0 dengan n.
2. Lakukan pembacaan *string bit* setelah angka 1 sebanyak n.
3. Ganti kode hasil pembacaan dengan karakter sesuai berdasarkan tabel *Elias Gamma Code*.

3. HASIL DAN PEMBAHASAN

3.1 Analisa Masalah

Komunikasi saat ini adalah sarana penting bagi kehidupan manusia dimana mereka saling bertukar informasi dengan penyampaian pesan antar sesama. Pada saat ini banyak sekali orang yang saling mengirim *file* MP4 dengan mengambil dari internet tanpa melihat seberapa besar kapasitas *file* MP4 tersebut dan tidak peduli seberapa besar media penyimpanan yang dimiliki. Selain itu masyarakat perlu menunggu waktu yang lama untuk menampilkan MP4 akibat terlalu besarnya kapasitas *file* MP4 tersebut. Sehingga *user* terkadang perlu melakukan pemampatan sebuah *file* MP4 yang dimilikinya agar mudah dalam pengiriman MP4 ataupun mengurangi ukurannya sehingga tidak memenuhi ruangan penyimpanan.

Pada penelitian ini akan dilakukan analisa dan perancangan perangkat lunak pemampatan *file* MP4 dengan menggunakan algoritma *Elias gamma code* pada aplikasi *chatting* yang berguna untuk mempercepat proses pengiriman *file*. Algoritma *Elias gamma code* merupakan jenis kompresi *Lossless* dimana dekomposisi dari *file* yang terkompresi sama dengan *file* aslinya, tidak ada informasi yang hilang. Tahapan analisa terhadap suatu sistem dilakukan sebelum tahapan perancangan dilakukan. Adapun tujuan dilakukan analisa terhadap suatu sistem adalah untuk mengetahui alasan mengapa sistem tersebut diperlukan, yaitu dengan merumuskan kebutuhan-kebutuhan dari sistem tersebut untuk meminimalisir sumber yang berlebih serta membantu merencanakan pembentukan sistem.

3.1.1 Penerapan Algoritma Elias Gamma Code

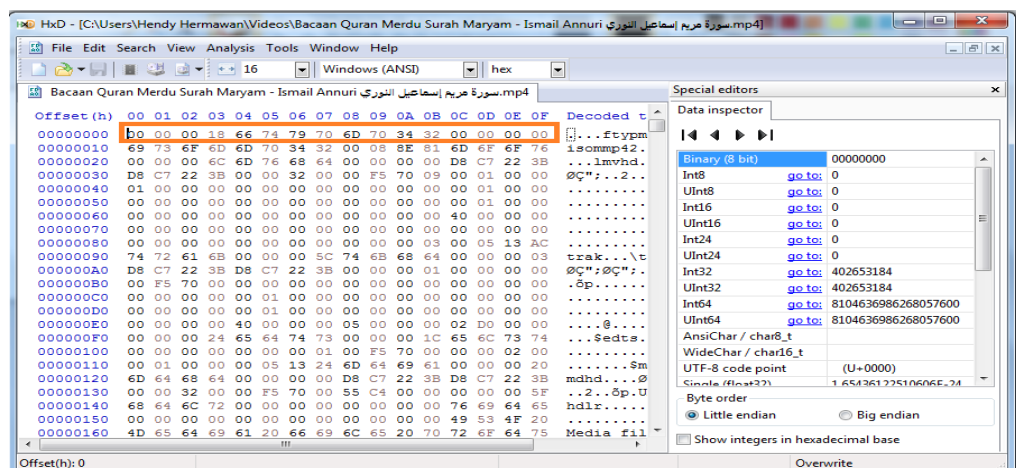
Dalam melakukan kompresi *file* MP4 sebelumnya harus dilakukan analisa terhadap *file* MP4 yang akan dikompresi. *File* MP4 merupakan *file* yang sangat dikenal dan paling populer diantara *file* yang lainnya. Dalam menganalisa *file* MP4 yang harus dilakukan adalah mengambil *sample file* MP4 dengan melakukan pembacaan *file* MP4. Pembacaan *file* MP4 dilakukan untuk mendapatkan nilai dari data pada sebuah *file* MP4 yang berupa bilangan *hexadesimal*. Dalam proses hitungan manual nilai *sample file* MP4 diambil menggunakan *software* *HxD Hex Editor*, kemudian data nilai *hexadesimal* MP4 di kompresi menggunakan algoritma *Elias Gamma Code*. Berikut informasi objek *file* MP4 yang akan diambil *sample*nya sebelum dilakukan kompresi.

Tabel 2. Informasi File Mp4 Sample

Keterangan	
Jenis file	.MP4
Judul	Bacaan Quran Merdu Surah Maryam - Ismail Annuri
Ukuran	163 MB
Durasi	20.56 Menit

1. Masukkan *file*.

Dari *sample* MP4 di dapat nilai *hexadesimal* menggunakan bantuan *software* *HxD Hex editor* seperti pada gambar 1.



Gambar 1. Nilai Hexadesimal File Mp4 Sample

Berdasarkan pada gambar diatas didapati *sample* nilai *hexadesimal file* MP4. Untuk keperluan hitungan manual hanya diambil dari sisi kiri atas sampai dengan bilangan ke 16.

2. Melakukan pembacaan isi *file*

Adapun bilangan *hexadesimal file* MP4 *sample* tersebut 00, 00, 00, 18, 66, 74, 79, 70, 6D, 70, 34, 32, 00, 00, 00, 00. Mengurutkan dari karakter yang memiliki *frekwensi* terbesar (banyak nilai yang sama) ke terkecil . urutan bilangan *hexadesimal* dapat dilihat pada tabel dibawah ini :

Tabel 3. Nilai Bit File MP4 Sample

Bilangan hexa	Frekuensi	Ascii decimal	Ascii binary	Bit	Bit x Frekuensi
00	7	00	00000000	8	56
18	1	24	00011000	8	8
66	1	102	01100110	8	8
74	1	116	01110100	8	8
79	1	121	01111001	8	8
70	2	112	01110000	8	16
6D	1	109	01101101	8	1
34	1	52	00110100	8	1
32	1	50	00110010	8	1
Jumlah Bit x frekuensi				107 bit	

3. Selanjutnya, melakukan proses pengurutan Bilangan berdasarkan frekuensi kemunculannya yang diurutkan dari frekuensi kemunculan terbesar ke kemunculan terkecil. Jika terdapat lebih dari satu bilangan *hexadesimal* dengan frekuensi kemunculan yang sama. Maka diurutkan berdasarkan abjad. Untuk hasil pengurutan bilangan dan frekuensi bilangan dapat dilihat sebagai berikut :

Tabel 4. Hasil Pengurutan Bilangan Set Kompresi *Elias Gamma Code* Berdasarkan Frekuensi Kemunculan.

No	Bilangan hexa	Frekuensi
1	00	7
2	70	2
3	66	1
4	74	1
5	79	1
6	18	1
7	6D	1
8	34	1
9	32	1
Jumlah		16 bit

4. Langkah selanjutnya adalah masing-masing bilangan *hexa* pada tabel 4. diatas diganti kode yang terdapat pada tabel kode *elias gamma* (tabel 2.1). Setelah diganti, hitung jumlah bit untuk tiap bilangan.

Tabel 5. Hasil Pergantian Nilai Frekuensi *Hexa* Berdasarkan Kode *Elias Gamma Code*.

No	Bilangan hexa	Frek hexa	Elias gamma code	Bit	Bit X Frek
1	00	7	1	1	7
2	70	2	010	3	6
3	66	1	011	3	3
4	74	1	00100	5	5
5	79	1	00101	5	5
6	18	1	00111	5	5
7	6D	1	00111	5	5
8	34	1	0001000	7	7
9	32	1	0001001	7	7
Jumlah bit X frek				50	

5. Selanjutnya adalah menyusun kembali kode-kode yang telah dibuat tabel 3.4 diatas sesuai dengan posisi bilangan pada *string file* video. Bilangan yang dimampatkan 00,00,00,70,66,74,79,18,6D,34,32,00,00,00,00. maka susunan kode-kode nya adalah :

00	00	00	18	66	74				
1	1	1	00110	011	00100				
79	70	6D	70	34	32	00	00	00	00
00101	010	00111	010	0001000	0001001	1	1	1	1

Dengan total bit yaitu 50 bit.

Karena 50 tidak habis dibagi 8 dan menyisakan 2, atau dengan kata lain:

$50 \bmod 8 = 2$, nyatakan hasil bagi tersebut dengan “n”, lalu masukkan rumus untuk penambahan *padding* :

$7 - n + “1”$

$7 - 2 + “1” = 5$, dari bit menjadi 000001

Sedangkan untuk menambahkan *flagging* dapat menggunakan rumus berikut :

$9 - n$

$9 - 2 = 7$, nilai binernya menjadi 00000111

Jadi penambahan *padding* dan *flagging string* bit yang dihasilkan jadi seperti ini berikut :

11100110 01100100 00101010 00111010 00010000 00100111 11000001 00000111

Total bit sebelumnya setelah penambahan pada *string bit* kode *elias gamma code* adalah $50 + 2 + 12 = 64$.

6. Langkah berikutnya bagi *string* bit menjadi per 8 bit, lalu merubahnya menjadi karakter.

11100110=230= æ

01100100=100= d

00101010=42= *

00111010=58= :

00010000=16=

00100111=39= ‘

11000001=193= Á

00000111=7=

Hasil dari proses pemampatan menghasilkan karakter berikut :

æ d * : ‘ Á

Berdasarkan hasil kompresi, maka diperoleh total bit data sebelum dikompresi adalah sebanyak 107 bit dan ukuran setelah dikompresi menjadi 64 bit, maka kinerja dari kompresi berdasarkan algoritma *elias gamma code* adalah sebagai berikut :

$$\text{Compression Ration (Cr)} = \frac{\text{ukuran data setelah dikompresi}}{\text{ukuran data sebelum dikompresi}} \times 100 \%$$

$$= \frac{64}{107} \times 100 \% = 59,8 \%$$

Proses *decoding* dilakukan untuk mengembalikan karakter yang dihasilkan dari proses *encoding* ke string bit.

- Langkah pertama yang dilakukan adalah mengkonversikan karakter kode-kode tersebut ke bilangan biner ,sehingga dihasilkan 64 bit. Æ = 230 = 11100110
d = 100 = 01100100
* = 42 = 00101010
: = 58 = 00111010
=16 = 00010000
‘ = 39 = 00100111
Á = 193 =11000001
• = 7 = 00000111Maka dapatlah string bit dari proses dekompresi :
11100110 01100100 00101010 00111010 00010000 00100111 11000001 00000111.
- Setelah merubah karakter menjadi string bit, langkah berikutnya yaitu menghilangkan *padding* dan *flagging* dengan cara mengambil 8 bit terakhir dan merubahnya menjadi bilangan *desimal*.
- kemudian nyatakan dengan “n” .
n = 00000111 = 7, karena nilai “n” adalah 7, berarti ada *padding* yang digunakan, lalu gunakan rumus “7 + n” seperti berikut :
7 + 7 = 14, maka hilangkan *string* bit sebanyak 14 bit terakhir.
11100110 01100100 00101010 00111010 00010000 00100111 11000001 00000111. Maka hasil menghilangkan *padding* dan *flagging* menjadi
11100110 011001000010101000111010000100000010011111.
- Lalu baca *string* bit dari kiri ke kanan dan sesuaikan dengan tabel 3.4 sampai menemukan sebuah bilang *hexadesimal*.
11100110011001000010101000111010000100000010011111
Maka dari proses dekompresi akan menghasilkan bilangan *hexadesimal* yaitu :
00,00,00,18,66,74,79,70,6D,70,34,32,00,00,00,00.

4. KESIMPULAN

Berdasarkan dari penelitian yang telah dilakukan, maka hasil akhir dari penelitian tersebut dapat diambil beberapa kesimpulan dari pembahasan sebelumnya. Adapun kesimpulan tersebut Prosedur dalam mengkompresi *file* MP4 dapat dilakukan dengan mencari nilai *hexadecimal file* MP4 dan kemudian diterapkan dengan menggunakan algoritma *elias gamma code*. Proses penerapan *algoritma elias gamma code* dalam peoses pemampatan *file* MP4 menghasilkan

compression ratio sebesar 59,8 % sehingga dapat menghemat ruang dan mempercepat proses pengiriman *file* MP4 yang dihasilkan.

REFERENCES

- [1] Rizky Syahputra, "Kompresi file video mp4 dengan menggunakan metode discrete cosine tranform," Jurnal Riset Komputer (JURKOM), vol. 3, no. 1, pp. 52-57, februari 2016.
- [2] TaroniSokhi Zebua Hikka Sartika, "perancangan dan implementasi algoritma elias gamma code untuk mengkompresi record database pada aplikasi rangkuman pengetahuan umum lengkap," Komik(konferensi nasional teknologi informasi dan komputer), vol. 3, no. 1, pp. 259-265, Oktober 2019.
- [3] M.Arif Dalma. (2020, July) dosenpintar.com. [Online]. <http://dosenpintar.com/pengertia-implementasi>
- [4] Ida Mengyi Pu, Fundamental Data Compression. Burlington: Linacre House,Jordan Hiil,Oxford OX2 8DP 30 Corporate Drive, 2006.
- [5] David Salomon, Data compression the Complete Reference, 175th ed. New York: NY 10010 ,USA, Springer-Verlag New York,inc, 2004.
- [6] S.Si D. T. Sutoyo, Teori pengolah citra digital, Andi, Ed. Yogyakarta: Andi, April 2009. [Online]. https://www.baktikominfo.id/en/informasi/pengetahuan/teknik_kompresi_data_dan_jenis-jenisnya_yang_perlu_anda_ketahui-795
- [7] Santanu Saputra. (2017, May) teknobos.com. [Online]. <https://www.google.com/amp/s/teknobos.com/amp/pengertian-jenis-format-file-video/>
- [8] G.Motta D.Salomon, HANDBOOK OF DATA COMPRESSION. Spinger London Dordrecht Heidelberg New york: 2010, 5 th edition.
- [9] Dosen pendidikan2. (2019, Nov.) DOSEN PENDIDIKAN. [Online]. https://www.dosenpendidikan.co.id/pengertian-chatting-fungsi-manfaat-dampak/#fungsi_dan_manfaat_chatting
- [10] Dani Iqbal, "Implementasi algoritma Levenstein untuk kompresi file video pada aplikasi Chatting berbasis android," KOMIK(Konferensi Nasioanal Teknologi Informasi dan Komputer), vol. 3, no. 1, pp. 266-273, Oktober 2019.
- [11] Rosa A.S and M.Salahuddin, Rekayasa Perangkat Lunak. Yogyakarta: Andi, 2016.
- [12] M.Sc,Eva Maulina Aritonang,S.Kom Alfa Satyaputra, "Java for Beginners with eclipse 4.2 juno," in Java for Beginners with eclipse 4.2 juno. Jakarta: PT.Elex Media Komputindo, 2012, pp. 12-13.
- [13] Candra Adi Putra. (2013, November) Candra.web.id. [Online]. <https://www.google.com/amp/www.candra.WEB.ID/VISUAL-GUIDE-ECLIPSE-IDE-UNTUK-PEMROGRAMAN-ANDROID/>
- [14] Sukiman and T.Chandra, Aplikasi Kompresi File dengan Algoritma Elias Gamma Code," core IT, vol.1, no.1, Teknik Informatka STMIK IBBI. MEDAN: 2013, 2013.