

Perbandingan Algoritma Elias Delta Code Dan Unary Coding Dalam Kompresi Citra Forensik

Michael Simangunsong¹

¹ Fakultas Ilmu Komputer dan Teknologi Informasi, Program Studi Teknik Informatika, Universitas Budi Darma, Medan, Indonesia
Email : ¹michaelsimangun@gmail.com

Abstrak—Kebutuhan manusia pada suatu citra sejak dahulu merupakan kebutuhan yang tidak bisa diabaikan begitu saja. Sejak ditemukannya alat-lat untuk menangkap suatu citra seperti kamera. Perkembangan teknologi informasi yang sangat cepat telah menjadi peran yang sangat penting untuk pertukaran informasi. Kecepatan pengiriman ini sangat bergantung pada ukuran dari informasi tersebut. Salah satu untuk mengatasi masalah di atas adalah dengan melakukan kompresi data sebelum ditransmisikan dan kemudian penerima akan melakukan dekompresi yaitu merekonstruksinya kembali menjadi data asli. Masalah lain adalah ketika kita ingin menyimpan file tersebut. Dengan ukuran yang besar maka file citra tersebut juga memerlukan media penyimpanan yang cukup besar. Dalam prakteknya, teknik kompresi terbagi menjadi dua, antara lain kompresi lossy dan kompresi lossless. Elias Delta Code merupakan terobosan yang dibuat oleh Piter Elias sebagai algoritma yang canggih yang menerapkan Elias Gamma Code sebagai dasar algoritmanya, sedangkan algoritma Unary Coding merupakan metode yang diinspirasi dari Neuroscience, di mana data yang ada biasanya dikodekan secara spasial, sehingga lokasi menunjukkan letak data tersebut. Hal ini dilakukan dengan menandai slot tertentu. Dengan adanya proses kompresi, sangat membantu manusia dalam hal pengiriman data, penghematan media penyimpanan dari besarnya kapasitas data yang di simpan.

Kata Kunci: Kompresi, Lossless, Algoritma Elias Delta Code, Unary Coding.

Abstract—Human needs in an image since long time ago is a need that cannot be ignored. Since the invention of the lat tool to capture an image like a camera. The rapid development of information technology has become a very important role for information exchange. This delivery speed is very dependent on the size of the information. One of the solutions to the above problem is to compress the data before it is transmitted and then the recipient will decompress that is reconstructing it back into the original data. Another problem is when we want to save the file. With a large size, the image file also requires a fairly large storage media In practice, compression techniques are divided into two, including lossy compression and lossless compression. Elias Delta Code is a breakthrough made by Piter Elias as a sophisticated algorithm that applies the Elias Gamma Code as the basis of the algorithm, while the Unary Coding algorithm is a method inspired by Neuroscience, where existing data is usually spatially coded, so the location shows where the data is located . This is done by marking certain slots. With the compression process, it is very helpful for humans in terms of sending data, saving storage media from the amount of data capacity stored.

Keywords: Compression, Lossless, Elias Delta Code Algorithm, Unary Coding.

1. PENDAHULUAN

Kecepatan waktu proses merupakan hal yang sangat dibutuhkan saat ini dengan alasan efisiensi waktu. Sebagai ilmu yang terus berkembang, ilmu komputer hadir dengan berbagai inovasi yang membantu kebutuhan masyarakat akan efisiensi waktu tersebut. salah satu inovasi tersebut adalah teknologi kompresi di mana kompresi bekerja dengan memperkecil ukuran data. Sebagaimana kita ketahui besar kecilnya suatu data dapat mempengaruhi waktu saat proses kerja.

Kompresi ialah sebuah proses di mana *file* (citra atau gambar, teks, audio maupun video) dapat ditransformasikan ke *file* lain (terkompresi), sehingga *file* yang terkompresi dapat sepenuhnya kembali seperti *file* aslinya tanpa kehilangan informasi yang sebenarnya[1]. Dalam kompresi, bit data dikompres untuk memperkecil ukuran data yang berulang. Dengan adanya kompresi, akan mudah untuk menyimpan lebih banyak data bila dibandingkan dengan data yang tidak dikompres serta data yang telah dikompres dapat ditransfer lebih cepat karena ada jumlah bit yang lebih sedikit dari aslinya. Proses ini akan sangat berguna jika kita ingin menghemat ruang penyimpanan.

Teknik kompresi terbagi atas dua bagian secara umum yaitu *lossy* dan *lossless*. Kompresi *lossy* adalah kompresi citra yang akan mengalami perubahan data setelah selesai dimampatkan, ini berarti data asli tidak dapat diregenerasi dari data yang terkompresi. Dalam kompresi *lossless* data yang diterima atau yang diproses akan mengembalikkan data persis seperti data asli. Pencitraan medis memberikan tantangan besar yang memiliki algoritma kompresi untuk mengurangi hilangnya ketelitian suatu citra sebanyak mungkin agar tidak menimbulkan kesalahan diagnostik yang belum memiliki tingkat kompresi yang tinggi untuk kapasitas penyimpanan yang sedikit dan waktu transmisi yang cepat.

Elias Delta Code merupakan terobosan yang dibuat oleh Piter Elias sebagai algoritma yang canggih yang menerapkan Elias Gamma Code sebagai dasar algoritmanya, sedangkan algoritma *Unary Coding* merupakan metode yang diinspirasi dari *Neuroscience*, di mana data yang ada biasanya dikodekan secara spasial, sehingga lokasi menunjukkan letak data tersebut. Hal ini dilakukan dengan menandai slot tertentu.

Pada penelitian sebelumnya “Perancangan Aplikasi Kompresi Menggunakan Metode Shannon Fano dan Unary Coding Pada File Teks” oleh Mhd. Rajani Pane dalam jurnalnya pada majalah Ilmiah Inti, Volume 12, Nomor 3, September 2017 mengatakan kedua jenis algoritma tersebut dapat digunakan pada proses kompresi teks[2]. Serta pada penelitian sebelumnya oleh Dian Rachmawati, Mohammad Andri Budiman, Cut Amalia Saffiera dalam jurnalnya yang berjudul “An Implementation Of Elias Delta Code And ElGamal Algorithm In Image Compression And Security”[3]. berdasarkan pada penelitian sebelumnya tersebut, penulis membuat konsep kompresi data *lossless* dengan membandingkan performa dalam melakukan kompresi dengan parameter rasio kompresi (RC dan CR), *Redudancy Data*

(RD) dan *Space Saving* (SS) antara algoritma *Elias Delta Code* dan *Unary Coding* dengan tujuan untuk mengetahui algoritma yang cocok dalam mengkompresi citra forensik.

2. METODOLOGI PENELITIAN

2.1 Kompresi

Kompresi adalah sebuah proses di mana *file* (citra atau gambar, teks, audio maupun video) dapat ditransformasikan ke *file* lain (terkompresi), sehingga *file* yang terkompresi dapat sepenuhnya kembali seperti *file* aslinya tanpa kehilangan informasi yang sebenarnya[1]. Kompresi pada citra bertujuan untuk meminimalkan kebutuhan ruang penyimpanan dalam merepresentasikan citra digital dengan mengurangi duplikasi data di dalam citra tersebut, sehingga ruang penyimpanan yang dibutuhkan lebih sedikit daripada citra sebelumnya. Rasio citra kompresi adalah ukuran persentase citra yang telah berhasil dimampatkan.

2.2 Citra

Citra adalah suatu gambaran, kemiripan atau imitasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optik berupa foto, bersifat sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada media penyimpanan[1].

2.3 Algoritma Elias Delta Code

Algoritma Elias Delta Code merupakan terobosan yang dibuat oleh Piter Elias sebagai algoritma yang canggih yang menerapkan Elias Gamma Code sebagai dasar algoritmanya[5]. Langkah-langkah *encoding* pada algoritma Elias Delta Code adalah sebagai berikut:

1. Tulis bilangan dalam bentuk biner.
 2. Buang bit paling kiri pada bentuk biner bilangan.
 3. Hitung sisa bit pada langkah 2 ditambah 1 dan tuliskan hasilnya dalam bentuk biner di depan bilangan pada langkah 2.
 4. Hitung jumlah bit pada langkah 3 dikurang 1 dan tuliskan jumlahnya dalam bentuk biner 0 lalu sisipkan bilangan tersebut pada awalan bilangan pada langkah 3.
- Sebagai contoh kasus pada bilangan 40
1. $40 \rightarrow 100000$
 2. Buang bit paling kiri, maka $\rightarrow 00000$
 3. Sisa bit pada langkah 2 ditambah dengan 1 $\rightarrow 5+1=6$ dalam bentuk biner ditulis 110, maka bilangan sementara adalah 110 | 00000
 4. Jumlah bit pada bilangan 110 adalah 3, maka $3-1=2$, maka bilangan biner dari langkah ini adalah 10 dimasukkan kedalam bilangan pada langkah 3, sehingga didapat 00 | 110 | 00000 atau 001100000.

Langkah untuk mendecoding algoritma Elias Delta Code adalah sebagai berikut:

1. Baca bilangan biner 0 sampai di temukan angka 1.
2. Jumlahkan jumlah biner 0 dengan 1 atau $n'+1$.
3. Baca bit bilangan setelah biner 0 sesuai hasil yang didapatkan pada langkah 2. Lalu kurangkan dengan 1 sehingga didapatkan n .
4. Dapatkan bilangan enkodenya dengan menjumlahkan $2^n + x$, x merupakan bilangan yang sisa.

2.4 Algoritma Unary Coding

Unary Coding merupakan metode yang diinspirasi dari *Neuroscience*, di mana data yang ada biasanya dikodekan secara spasial, sehingga lokasi menunjukkan letak data tersebut. Hal ini dilakukan dengan menandai slot tertentu. Metode Unary Coding umumnya direpresentasikan dalam sebuah string dari n dan bit 1 diikuti dengan satu bit 0 yang mengakhiri yang didefinisikan sebagai $n-1$ bit 1 diikuti satu bit 0, atau sebaliknya sebagai alternatif dapat juga secara ekuivalen dimulai dari n bit 0 diikuti dengan bit 1 yang mengakhiri yang didefinisikan sebagai $n-1$ bit 0 diikuti dengan satu bit 1. Pada metode Unary Coding tidak terdapat pembagian frekuensi simbol-simbol yang ada pada string. Berikut ini adalah langkah-langkah algoritma kompresi Unary Coding.

1. $Low \leftarrow 0.0$
2. $High \leftarrow 1.0$
3. **While** (simbol-simbol masih ada) **do**
4. Ambil simbol *input*
5. $Code\ Range \leftarrow high - low$
6. $High \leftarrow low + Code\ Range \times high_range(s)$
7. $Low \leftarrow low + Code\ Range \times low_range(s)$
8. **and while**
9. **output low**

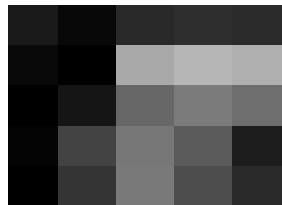
Tabel 1. Tabel Unary Coding

N	Kode	Kode Alternatif
1	0	1
2	10	01
3	110	001
4	1110	0001
5	11110	00001
6	111110	000001
7	1111110	00000001
8	11111110	000000001
9	111111110	0000000001
10	1111111110	00000000001

3. HASIL DAN PEMBAHASAN

3.1 Analisa Masalah

Pada penelitian ini akan diuraikan bagaimana proses kompresi untuk memperkecil ukuran citra dilakukan dengan menerapkan saat dilakukannya proses kompresi dan untuk mengembalikan data citra tersebut keukuran aslinya dilakukan proses dekompresi. Berikut contoh sebuah citra *grayscale* yang berukuran 5x5 *pixel* yang akan dikompres dengan acuan nilai *file byte_array* dari citra tersebut. Dimana ukuran citra resolusi 256 x 256*pixel* diperkecil dengan menggunakan *Microsoft office Picture Manager* menjadi resolusi 5 x 5 *pixel*. Berikut gambar citra *grayscale* forensik sidik jari dengan ukuran 256 x 256 *pixel*.

Gambar 1. Citra *Grayscale* Forensik Sidik Jari resolusi 5 x 5

3.1.1 Penerapan Algoritma Elias Delta Code

1. Diberikan *input* data nilai *pixel* citra sebagai berikut:

Tabel 1. Matriks Citra *Grayscale* 5x5

203	192	147	203	198
185	137	103	117	144
129	107	115	127	147
89	97	113	139	143
254	252	247	254	253

2. Dari *input* data tersebut, setiap karakter kemudian dipetakan kedalam tabel frekuensi kemunculan data yang digambarkan dalam tabel 2. berikut ini:

Tabel 2. Frekuensi Data Dengan Binary Code

n	Pixel	Frekuensi	Ascii Binary Code	Jumlah Bit	Total Bit
1	89	1	01101011	8	8
2	97	1	01101111	8	8
3	103	1	01110000	8	8
4	107	1	10000111	8	8
5	113	1	10001111	8	8
6	115	1	10010001	8	8
7	117	1	10101000	8	8
8	127	1	10101011	8	8
9	129	1	10101110	8	8
10	137	1	10101111	8	8
11	139	1	11000000	8	8
12	143	1	11000100	8	8
13	144	1	11010000	8	8
14	147	2	11010011	8	16

n	Pixel	Frekuensi	Ascii Binary Code	Jumlah Bit	Total Bit
15	185	1	11010101	8	8
16	192	1	11011101	8	8
17	198	1	11100001	8	8
18	203	2	11101111	8	16
19	247	1	11110010	8	8
20	252	1	11110101	8	8
21	253	1	11111101	8	8
22	254	2	11111111	8	16
Total		25		176	200

3. Langkah selanjutnya adalah mengurutkan nilai frekuensi secara *descending order* dan mengkonversi nilai *Elias Delta Code* yang ditunjukkan pada tabel 3.

Tabel 3. Pengurutan Frekuensi dan Konversi Elias Delta Code

n	Pixel	Frekuensi	Elias Delta Code	Jumlah Bit	Total Bit
1	254	2	1	1	2
2	203	2	0100	4	8
3	147	2	0101	4	8
4	89	1	01100	5	5
5	97	1	01101	5	5
6	103	1	01110	5	5
7	107	1	01111	5	5
8	113	1	00100000	8	8
9	115	1	00100001	8	8
10	117	1	00100010	8	8
11	127	1	00100011	8	8
12	129	1	00100100	8	8
13	137	1	00100101	8	8
14	139	1	00100110	8	8
15	143	1	00100111	8	8
16	144	1	001010000	9	9
17	185	1	001010001	9	9
18	192	1	001010010	9	9
19	198	1	001010011	9	9
20	247	1	001010100	9	9
21	252	1	001010101	9	9
22	253	1	001010110	9	9
Total Bit					165

Berdasarkan pada tabel 3. di atas dapat dibentuk nilai bit baru hasil kompresi dari susunan nilai decimal *pixel* citra sampel awal sebelum kompresi yaitu 203, 192, 147, 203, 198, 185, 137, 103, 117, 144, 129, 107, 115, 127, 147, 89, 97, 113, 139, 143, 254, 252, 247, 254, 253. (tanpa tanda koma dan spasi) menjadi nilai ke biner.

0010101100010101000010010101111001010110001010000001010101100011100100001000010010100000101001000100000011011010100100100100100011001001110100.

Dari hasil kompresi dengan menggunakan *Elias Delta Code* di atas, dapat dihitung kinerja kompresinya sebagai berikut:

- a. *Ratio of Compression* (R_c)

$$R_c = \frac{\text{ukuran data sebelum dikompresi}}{\text{ukuran data setelah dikompresi}}$$

$$R_c = \frac{200}{165}$$

$$R_c = 1.212$$

- b. *Compression Ratio* (CR)

$$CR = \frac{\text{ukuran data setelah dikompresi}}{\text{ukuran data sebelum dikompresi}} \times 100\%$$

$$CR = \frac{165}{200} \times 100\% = 82\%$$

- c. *Space Saving*

$$SS = 100\% - CR$$

$$SS = 100\% - 82\%$$

$$SS = 18\%$$

Pada proses kompresi sebelumnya telah dihasilkan data berupa kode biner seperti berikut ini: 001010110001010100001001010111001010110001010000001010101100011100100001000010010100000101001000100000011011010100100100100100011001001110100.

Setiap kode biner tersebut akan dilakukan pembacaan karakter, misalnya biner pada index pertama dibaca 0 selanjutnya dilakukan pengecekan kedalam tabel Elias Delta Code untuk mendapatkan karakter yang sesuai. Jika tidak terdapat bit 0 pada tabel maka lanjutkan pengecekan pada biner selanjutnya yaitu 00, dan tidak juga terdapat pada tabel. Lanjutkan pengecekan pada biner selanjutnya yaitu 001, setelah dilakukan pengecekan pada tabel, biner 001 tidak juga terdapat pada kode Elias Delta Code. Lanjutkan pengecekan dengan biner berikutnya hingga didapatkan karakter yang sesuai dengan kode Elias Delta Code. Pada pengecekan data diatas, karakter akan ditemukan setelah dilakukan pengecekan pada biner ke-8. Selanjutnya adalah membangkitkan karakter Ascii maka didapatkan karakter 255 akan dibangkitkan sebagai hasil dekompresi pada file ini. Setelah dilakukan pengecekan pada seluruh bit kode, maka akan membangkitkan seluruh *pixel* dari citra yang semula yaitu:

203, 192, 147, 203, 198, 185, 137, 103, 117, 144, 129, 107, 115, 127, 147, 89, 97, 113, 139, 143, 254, 252, 247, 254, 253. Dan disusun kembali ke dalam bentuk matriks 5 x 5.

203	192	147	203	198
185	137	103	117	144
129	107	115	127	147
89	97	113	139	143
254	252	247	254	253

3.1.2 Penerapan Algoritma Unary Coding

Adapun cara yang dilakukan dalam mengkompresi citra forensik dengan cara atau aturan yang ada pada algoritma unary coding adalah sebagai berikut:

1. Mengurutkan karakter dari yang jumlahnya paling sedikit ke yang paling banyak.
2. Membuat kode biner untuk nilai n.
3. Mengganti nilai n menjadi nilai kode bit.

Setelah matriks citra forensik sudah didapatkan maka langkah selanjutnya adalah mengurutkan data matriks tersebut dan membuat frekuensi dari setiap kemunculan yang ada pada matriks.

Tabel 4. Frekuensi Data Dengan Binary Code

No	Piksel	Frek.	ASCII Binary Code	Jlh. Bit	Total Bit
1	89	1	01101011	8	8
2	97	1	01101111	8	8
3	103	1	01110000	8	8
4	107	1	10000111	8	8
5	113	1	10001111	8	8
6	115	1	10010001	8	8
7	117	1	10101000	8	8
8	127	1	10101011	8	8
9	129	1	10101110	8	8
10	137	1	10101111	8	8
11	139	1	11000000	8	8
12	143	1	11000100	8	8
13	144	1	11010000	8	8
14	147	2	11010011	8	16
15	185	1	11010101	8	8
16	192	1	11011101	8	8
17	198	1	11100001	8	8
18	203	2	11101111	8	16
19	247	1	11110010	8	8
20	252	1	11110101	8	8

Lakukan pengurutan frekuensi dari yang terkecil ke yang terbesar kemudian bangkitkan kode Unary Coding.

Tabel 5. Frekuensi Data Dengan Unary Coding

No	Piksel	Frek	Unary Coding	Jlh. Bit	Total Bit
1	254	2	0	1	2
2	203	2	10	2	4
3	147	2	110	3	6
4	89	1	1110	4	4
5	97	1	11110	5	5

Tabel 6. Perbandingan Rc, Cr, SS pada kedua Algoritma

Algoritma Elias Delta Code		Algoritma Unary Coding
Ratio Of Compressi (Rc)	1.212	0.722
Compression Ratio (Cr)	82 %	1.29 %
Space Saving (SS)	18 %	0.98 %

3.2 Implementasi

Adapun yang menjadi kebutuhan sistem adalah berdasarkan spesifikasi perangkat keras dan perangkat lunak standart program ini dijalankan dengan menggunakan perangkat keras (*Hardware*) yang direkomendasikan adalah sebagai berikut:

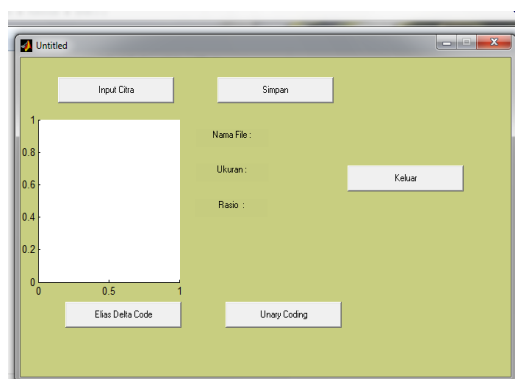
Sistem tidak akan dapat berjalan dengan baik jika tidak didukung oleh *Hardware* dan *Software* yang memadai, sama hal nya dengan sistem yang penulis rancang, berikut adalah spesifikasi *hardware* yang penulis gunakan dalam merancang sistem:

1. *Processor* minimal *Dual Core*
2. *Harddisk* minimal 100 GB
3. *Memori* 1 GB
4. *VGA* 1 GB
5. *LCD Monitor* minimal 10 *inchi*

Sistem dapat berjalan dengan bantuan *Software* yang dapat mendukung *Hardware* diatas perangkat lunak seperti Sistem Operasi dan Program Aplikasi seperti berikut

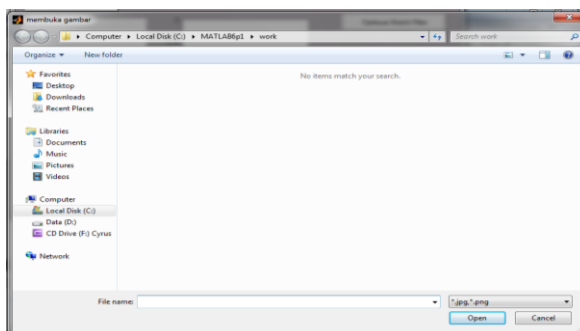
1. *Sistem Operasi Windows 7*
2. *Matlab 6.1*

Pada proses implementasi penerapan metode *Elias Delta Code* dan *Unary Coding* yang dapat memperkecil ukuran pada citra forensik yang telah dianalisis dan telah dirancang sebelumnya dengan menggunakan bahasa *Pemrograman Matlab 6.1* yang merupakan *tools* yang dapat dibuat untuk aplikasi kompresi citra forensik maka akan dilakukan uji coba program ini yang dijalankan dengan menggunakan *Windows 7* yang mana PC tersebut perlu untuk disimulasikan dengan menginstal komponen *windows* serta komponen *Matlab 6.1* Adapun tampilan dari penerapan Algoritma *Elias Delta Code* dan *Unary Coding* untuk kompresi pada citra dapat dilihat dari tampilan berikut ini :



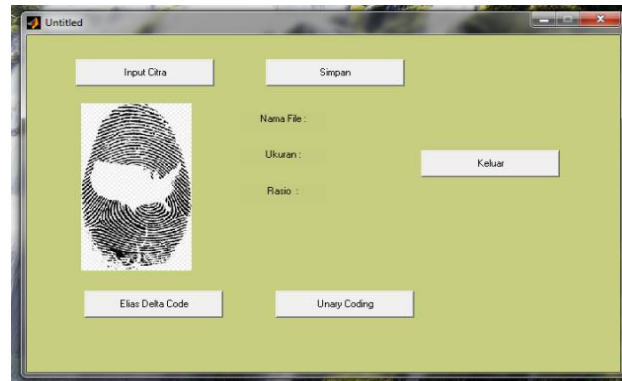
Gambar 2. Tampilan Program Utama

Merupakan tampilan program untuk menginput gambar dari *directori* penyimpanan gambar yang akan di input untuk dapat dikompresi pada citra tersebut.



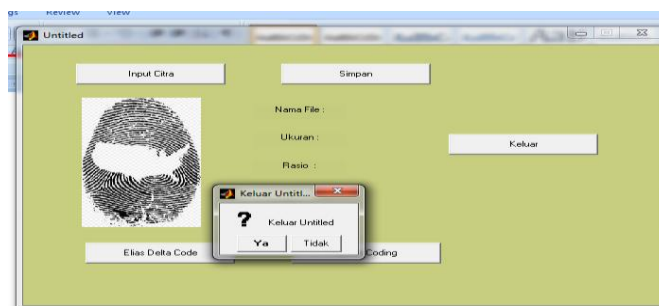
Gambar 3. Tampilan Cari Citra

Setelah tombol input citra diklik maka akan masuk ke *directory* penyimpanan citra, setelah citra tersebut ditemukan maka langkah selanjutnya yang dilakukan oleh *user* adalah klik open maka citra akan muncul.

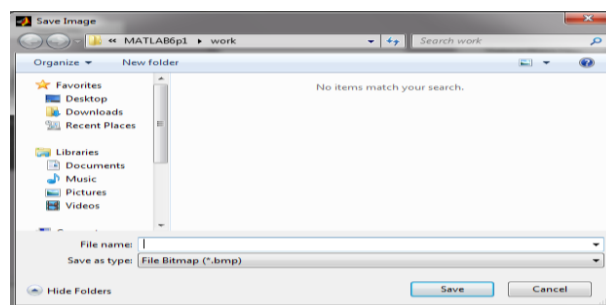


Gambar 4. Tampilan Input Citra Pada Program

Setelah citra di input maka akan muncul citra forensik yang akan dikompresi dan langkah selanjutnya dengan melakukan kompresi pada citra tersebut. Ini merupakan tampilan dari menu program, yang keluar dari aplikasi untuk mengkompresi citra forensik agar ukuran citra dapat lebih kecil dari pada ukuran citra sebelumnya, ketika tombol Keluar diklik maka akan muncul perintah untuk keluar dari aplikasi, jika *user* klik ya maka program akan diakhiri, dan jika *user* klik tidak maka program keluar tidak dilakukan.



Gambar 5. Tampilan Keluar Program



Gambar 6. Tampilan Simpan Citra

4. KESIMPULAN

Berdasarkan analisa dan pembahasan yang dilakukan, maka dapat disimpulkan Algoritma elias delta code dan unary coding adalah algoritma yang dapat mengkompresi citra forensik. Format citra yang paling baik untuk mengkompresi citra forensik adalah format bitmap .bmp yang hasilnya jauh lebih baik dari ekstensi lainnya. Hasil kompresi dapat dilihat berdasarkan rasio yang diperoleh, jika hasil rasio positif maka hasil kompresi baik dan jika rasio negatif maka hasil kompresi buruk. Perancangan aplikasi dengan menggunakan *software* Matlab sangat memudahkan untuk mengolah data citra untuk mengkompresi.

REFERENCES

- [1] T. Sutoyo, E. Mulyanto, V. Suhartono, O. D. Nurhayati, and Wijanarto, Teori Pengolahan Citra Digital, I. Yogyakarta: CV. Andi, 2009.
- [2] I. Pendahuluan et al., "Perancangan aplikasi kompresi menggunakan metode shannon fano dan unary coding pada file teks," vol. 12, no. September, pp. 306–311, 2017.
- [3] D. Rachmawati, M. A. Budiman, and C. A. Saffiera, "An Implementation Of Elias Delta Code And ElGamal Algorithm In Image Compression And Security," IOP Conf. Ser. Mater. Sci. Eng., vol. 300, p. 012040, 2018.
- [4] Darma Putra, Pengolahan Citra Digital, I. Yogyakarta: CV. Andi, 2010.
- [5] Abdul Kadir & Adhi Susanto, Teori dan Aplikasi Pengolahan Citra, I. Yogyakarta: CV. Andi Offset, 2013.

- [6] Rinaldi Munir, Algoritma & Pemrograman Dalam Bahasa Pascal dan C, 3rd ed. Bandung: Informatika Bandung, 2011.
- [7] David Salomon, Data Compression, 4th ed. Springer, 2007.
- [8] "Image Forensic." [Online]. Available: <https://www.rti.org/impact/forensic-technology-center-excellence-ftcoe>.
- [9] Rosa A. S.;M. Shalahuddin, Rekayasa Perangkat Lunak, 4th ed. Bandung: INFORMATIKA Bandung, 2016.
- [10] Budi Santosa, Matlab untuk Statistika & Teknik Optimasi, I. Yogyakarta: Graha Ilmu, 2008.