

IMPLEMENTASI KOMPRESI CITRA DIGITAL DENGAN MEMBANDINGKAN METODE *LOSSY* DAN *LOSSLESS* COMPRESSION MENGGUNAKAN MATLAB

Witriana Endah Pangesti^[1]; Galih Widagdo^[2]; Dwiza Riana ^[3] Sri Hadiani^[4]

Pascasarjana Magister Ilmu Komputer

Sekolah Tinggi Ilmu Manajemen dan Informatika Nusa Mandiri Jakarta

www.nusamandiri.ac.id

witriana21@gmail.com^[1]; galihwidagdo19@gmail.com^[2]; dwiza@nusamandiri.ac.id^[3];

sri.shv@nusamandiri.ac.id^[4]

Abstract— Current image processing technology has been developed, one of which is compression technology. Digital image compression is an attempt to transform data or symbols, without involving significant changes to digital images for humans who observe them. Image is another term for image as one of the multimedia components that plays a very important role as a form of visual information. Lossy compression and lossless compression where the lossy compression method is a data compression method that removes some of the information as a lossless compression method is a method of data compression with no data information being released or discarded through the compression process. Required after the decompression process the number of bits (bytes) of data or information in the overall data result is the same as the full data (Saragih and Harahap 2019). From previous studies, namely the Implementation of Digital Image Compression By finding Digital Image Quality (Raharja and Harsadi 2018) the authors proceed to use 20 different images compressed with lossless compression methods, given an average as compresses with forty percent percent. Previous research papers that use lossy methods produce an average of sixty percent compress and research papers that the authors do with lossless methods that produce an average compression of forty-nine percent, it can be concluded.

Keywords: Imagery, Lossy compression, Lossless compression, MATLAB, Compressing.

Intisari—Teknologi pengolahan citra saat ini telah banyak berkembang, salah satunya teknologi kompresi. Kompresi citra digital merupakan suatu upaya untuk melakukan tranformasi terhadap data atau simbol, tanpa menimbulkan perubahan yang signifikan atas citra digital bagi mata manusia yang mengamatinnya. Citra merupakan istilah lain untuk gambar sebagai salah satu komponen multimedia memegang peranan sangat penting sebagai bentuk informasi visual. Dalam penelitian

ini penulis menggunakan metode Lossy compression dan Lossless compression dimana metode Lossy compression yaitu suatu metode kompresi data yang menghilangkan sebagian informasi sedangkan metode Lossless compression yaitu suatu metode kompresi data dengan tidak ada informasi data yang hilang atau berkurang jumlahnya selama proses kompresi. Sehingga setelah proses dekompresi jumlah bit (byte) data atau informasi dalam keseluruhan data hasil sama persis dengan data aslinya (Saragih and Harahap 2019). Dari penelitian sebelumnya yaitu Implementasi Kompresi Citra Digital Dengan Mengatur Kualitas Citra Digital (Raharja and Harsadi 2018) penulis melanjutkan dengan melakukan perbandingan dengan menggunakan 20 citra berbeda lalu di kompres dengan metode lossless compression, diketahui rata-rata setelah kompres adalah 49,133005 %. Paper penelitian sebelumnya yang menggunakan metode lossy menghasilkan rata-rata kompres 61,40 dan paper penelitian yang penulis lakukan dengan metode lossless yang menghasilkan rata-rata kompresi 49,133005%, maka dapat di simpulkan bahwa metode lossy lebih baik dibandingkan dengan metode lossless dalam mengkompresi citra.

Kata Kunci: Citra, Lossy compression, Lossless compression, MATLAB, Compressing.

PENDAHULUAN

Teknologi pengolahan citra saat ini telah banyak berkembang, salah satunya teknologi kompresi. Kompresi citra digital merupakan suatu upaya untuk melakukan tranformasi terhadap data atau simbol, tanpa menimbulkan perubahan yang signifikan atas citra digital bagi mata manusia yang mengamatinnya.

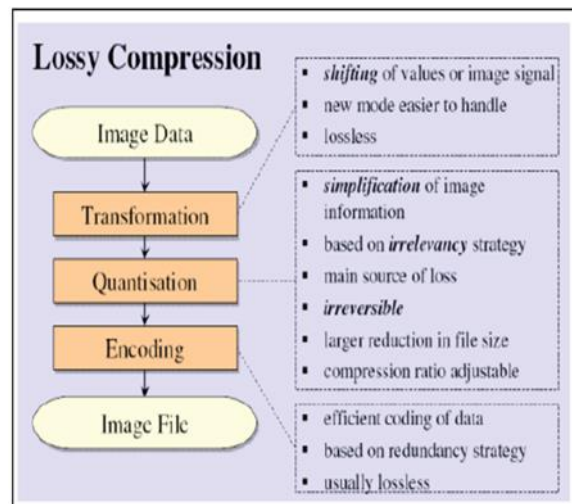
Kompresi data dalam bidang ilmu komputer, ilmu pengetahuan dan seni adalah sebuah penyajian informasi ke dalam bentuk yang lebih sederhana. Kompresi data dapat di artikan juga

sebagai proses yang dapat mengubah sebuah aliran data masukan (sumber atau data asli) ke dalam aliran data yang lain (keluaran atau data yang dimampatkan) yang memiliki ukuran yang lebih kecil (Rahmat et al., 2012).

Teknik kompresi citra merupakan suatu teknik yang digunakan untuk merepresentasikan citra melalui pengurangan kualitas pada citra asli, namun tetap mempertahankan informasi. (Raharja and Harsadi 2018). Supaya Supaya pengguna dapat dengan nyaman menyimpan maupun mentransmisikan citra tanpa ada masalah seperti lambat mentransfer citra maupun error saat mengirim citra. Maka pada penelitian ini bertujuan bagaimana mengkompresi citra dengan membandingkan metode *Lossy compression* pada penelitian sebelumnya dengan metode Lossless compression yang penulis lakukan untuk mengetahui metode mana yang paling baik dalam pengurangan ukuran citra.

BAHAN DAN METODE

Kompresi merupakan proses perubahan sekumpulan data menjadi suatu bentuk kode untuk menghemat kebutuhan tempat penyimpanan dan waktu untuk transmisi data. Metode kompresi data dapat dibagi ke dalam dua kelompok besar yaitu (Saragih & Harahap, 2019) : *Lossy compression* Yaitu suatu metode kompresi data yang menghilangkan sebagian "informasi" dari data asli selama proses kompresi dengan tidak menghilangkan secara signifikan informasi yang ada dalam data secara keseluruhan. Kompresi jenis ini dapat mengurangi ukuran data secara signifikan, dan data yang sudah terkompresi dapat dikompres lagi sampai batas-batas tertentu. *Lossy compression* biasanya digunakan dalam bidang Multimedia karena sering digunakan Untuk mengurangi ukuran data yang bersifat audio-visual. Misalnya untuk memperkecil ukuran file gambar, musik digital, dan untuk enkoding film dari format High Definition seperti Blu-Ray menjadi format MPEG (Moving Picture Experts Group). Contoh: discrete cosine transform, vector quantization, wavelet compression, distributed source coding (DSC). Berikut ini adalah ilustrasi dari kompresi lossy di jelaskan pada gambar 1:



Gambar 1. Lossy compression

Lossless compression Yaitu suatu metode kompresi data dengan tidak ada "informasi" data yang hilang atau berkurang jumlahnya selama proses kompresi. Sehingga setelah proses dekompresi jumlah bit (byte) data atau informasi dalam keseluruhan data hasil sama persis dengan data aslinya. Kompresi jenis ini tidak selalu dapat mengurangi ukuran data secara berarti, karena tidak semua data mengandung informasi yang tidak perlu. Selain itu, data yang telah dikompresi tidak dapat dikompresi lagi. *Lossless compression* biasanya digunakan untuk mengurangi ukuran data-data yang penting, data-data yang isinya tidak boleh berubah sedikitpun. Misalnya data-data yang berbentuk dokumen, teks, spreadsheet, kode sumber (source code), dan data data program. Contoh: Huffman coding, dynamic markov compression (DMC), Lempel-Ziv- Welch (LZW), arithmetic coding, Run-Length encoding. Berikut ini adalah gambar ilustrasi dari kompresi lossless di jelaskan pada gambar 2:



Gambar 2. Lossless compression

HASIL DAN PEMBAHASAN

1. Tampilan Sistem GUI

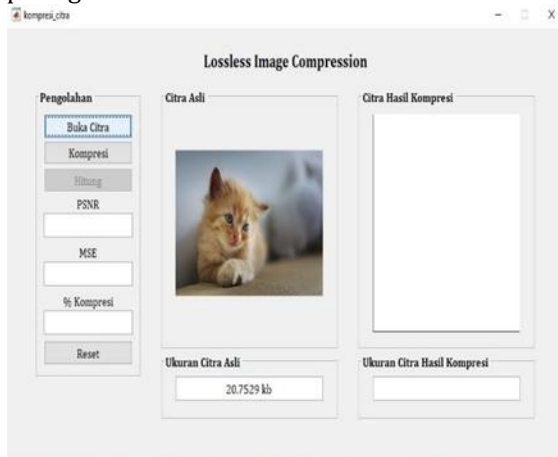
Berikut ini adalah GUI yang di buat menggunakan aplikasi MATLAB dengan metode *Lossless* untuk mengkompresi citra, Objek citra yang digunakan sebagai data uji adalah citra sekunder yang diperoleh dari internet. Citra tersebut akan di kompresi tanpa mengurangi kualitas citra. Tampilan GUI seperti pada gambar 3:



Gambar 3. Tampilan GUI

2. Cara kerja aplikasi

Langkah awal adalah membuka sistem GUI dengan metode *lossless* pada MATLAB seperti pada gambar 4 di bawah ini:



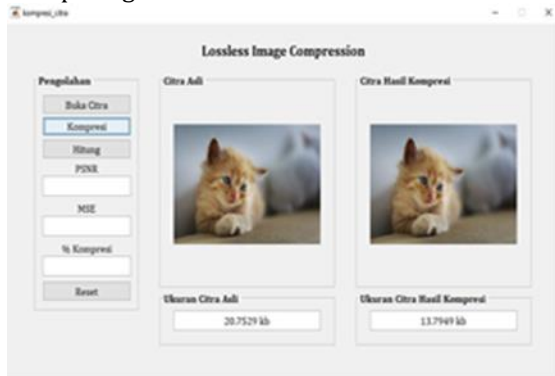
Gambar 4. Langkah Awal Metode Lossless

Lalu klik tombol buka citra untuk memasukan citra yang di pilih untuk diproses. Dengan pseudo-code di bawah ini:

```
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
```

```
% handles structure with handles and user data
(see GUIDATA)
% menampilkan menu browse file
[file, path] =
uigetfile({'*.bmp;*.jpg;*.png;*.tiff;*.tif;*.gif'},...
'Select an Image File');
% jika ada file citra yang dipilih maka akan
mengekseskusi perintah di
% bawahnya
if ~isequal(file,0)
% membaca file citra yang dipilih
Img = im2double(imread(fullfile(path,file)));
% melihat ukuran file citra dalam satuan kb
fileinfo = imfinfo(file);
SIZE = fileinfo.FileSize;
Size = SIZE/1024;
% menampilkan ukuran file citra dalam satuan
kb
set(handles.edit3,'String',[num2str(Size),' kb']);
% menampilkan file citra ke dalam axes
imshow(Img,'Parent',handles.axes1)
% menyimpan variabel Img & fileinfo pada
lokasi handles (lokasi
% penyimpanan variabel dalam MATLAB) agar
dapat dipanggil oleh
% pushbutton yang lain
handles.Img = Img;
handles.fileinfo = fileinfo;
guidata(hObject, handles)
% mereset button2
axes(handles.axes2)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])
set(handles.pushbutton2,'Enable','on')
set(handles.pushbutton3,'Enable','off')
set(handles.edit1,'String',[])
set(handles.edit2,'String',[])
set(handles.edit4,'String',[])
set(handles.edit5,'String',[])
% jika tidak ada file citra yang dipilih maka akan
kembali
else
return
end
```

Berikut ini adalah tampilan proses kompresi pada GUI pada gambar 5:



Gambar 5. Proses Kompresi pada GUI

lalu klik tombol kompresi dengan pseudo_code dibawah ini:

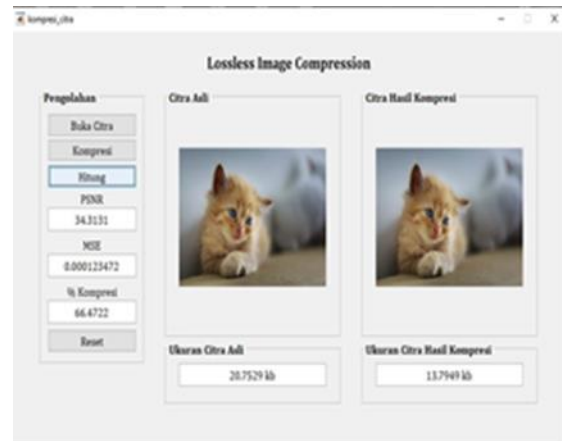
```
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% memanggil variabel Img & fileinfo yang ada pada lokasi handles
Img = handles.Img;
fileinfo = handles.fileinfo;

% melakukan lossless kompresi terhadap citra asli
Img_comp = zeros(size(Img));
for n = 1:3
    I = im2double(Img(:,:,n));
    T = dctmtx(8);
    B = blkproc(I,[8 8],'P1*x*P2',T,T');
    mask = [1 1 1 1 0 0 0 0
            1 1 1 0 0 0 0 0
            1 1 0 0 0 0 0 0
            0 0 0 0 0 0 0 0
            0 0 0 0 0 0 0 0
            0 0 0 0 0 0 0 0
            0 0 0 0 0 0 0 0
            0 0 0 0 0 0 0 0];
    B2 = blkproc(B,[8 8],'P1.*x',mask);
    Img_comp(:,:,n) = blkproc(B2,[8 8],'P1*x*P2',T,T');
end
% menyimpan file citra hasil kompresi
file_comp = ['Hasil Kompresi.',fileinfo.Format];
imwrite(Img_comp,file_comp);
% melihat ukuran file citra hasil kompresi dalam satuan kb
fileinfo_comp = imfinfo(file_comp);
SIZE = fileinfo_comp.FileSize;
Size = SIZE/1024;
```

% menampilkan ukuran file citra hasil kompresi dalam satuan kb

Berikut ini adalah tampilan proses hitung pada GUI pada gambar 6:



Gambar 6 Tampilan Proses Hitung pada GUI

Lalu klik hitung dengan pseudo_code dibawah ini:

```
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton3 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% memanggil variabel Img, fileinfo, Img_comp, dan fileinfo_comp yang ada pada lokasi handles
Img = handles.Img;
fileinfo = handles.fileinfo;
Img_comp = handles.Img_comp;
fileinfo_comp = handles.fileinfo_comp;

% menghitung nilai PSNR, MSE, dan % kompresi
nilai_psnr = psnr(Img,Img_comp);
nilai_mse = mean(mse(Img,Img_comp));
filesize = fileinfo.FileSize;
filesize_comp = fileinfo_comp.FileSize;
persen_kompresi = filesize_comp/filesize*100;
% menampilkan nilai PSNR, MSE, dan % kompresi pada edit text
set(handles.edit1,'String',nilai_psnr)
set(handles.edit2,'String',nilai_mse)
set(handles.edit5,'String',persen_kompresi)
```

Berikut ini adalah tampilan proses reset pada GUI pada gambar 7:



Gambar 7. Tampilan Proses Reset pada GUI

untuk mereset GUI klik tombol reset dengan pseudo_code dibawah ini:

```
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton4 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
```

```
% mereset button2
axes(handles.axes1)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])
axes(handles.axes2)
cla reset
set(gca,'XTick',[])
set(gca,'YTick',[])
set(handles.pushbutton2,'Enable','off')
set(handles.pushbutton3,'Enable','off')
set(handles.edit1,'String','')
set(handles.edit2,'String','')
set(handles.edit3,'String','')
set(handles.edit4,'String','')
set(handles.edit5,'String','')
```

Disini penulis menggunakan 20 data citra yang berbeda baik ukuran file citra maupun informasi citra yang ada. Pada tabel 1 adalah hasil penelitian yang telah dilakukan oleh penulis :

Gambar	Asli(Kb)	Kompresi (Kb)	MSE	PSNR (db)	Persentase Kompresi (Kb)
1	19,7354	9,6592	0,000684196	26,877	48,9435
2	20,7529	13,7949	0,000123472	34,3131	66,4722
3	22,4365	13,0527	0,000216751	31,8692	58,1763
4	25,9414	12,2461	0,000467462	28,5313	47,2067
5	26,9131	11,5781	0,000329655	30,0482	43,0204
6	27,0752	12,5918	0,001399184	23,7929	46,5068
7	33,0264	17,4434	0,000520423	28,0652	52,8165
8	33,3018	19,4834	0,000269811	30,9182	58,5056
9	33,3184	18,6973	0,000379403	29,4378	56,117
10	34,0498	18,96	0,000612279	27,3593	55,683
11	35,1807	18,1348	0,000388653	29,3332	51,5475
12	36,5264	17,417	0,000254758	31,1675	47,6833
13	39,3525	15,1572	0,00130032	24,0883	38,5165
14	40,0078	20,0693	0,000797566	26,2111	50,1635
15	41,082	19,585	0,000502329	28,2189	47,6728
16	41,2393	16,8154	0,00108622	24,8696	40,7753
17	45,0488	20,502	0,00074582	26,5024	45,5105
18	45,4805	17,8672	0,000809432	26,147	39,2854
19	47,9336	23,6865	0,000652997	27,0797	49,4153
20	48,9443	18,9131	0,00162433	23,122	38,642
			0,000658253	27,8976	49,133005

Gambar 8. Hasil Penelitian

Dari 20 citra yang di kompres, diketahui rata-rata setelah kompres adalah 49,133005 %. Dengan 20 citra yang berbeda ukuran file dan informasi citra didalamnya dari paper penelitian sebelumnya yang menggunakan metode lossy menghasilkan rata-rata kompres 61,40% dan paper penelitian yang penulis lakukan dengan metode lossless menghasilkan rata-rata kompresi 49,133005%, maka dapat di simpulkan bahwa metode lossy lebih baik dibandingkan dengan metode lossless dalam mengkompresi citra.

KESIMPULAN

Paper penelitian sebelumnya yang menggunakan metode lossy menurunkan ukuran citra dengan besar tetapi tingkat kualitas citra rendah sedangkan pada penelitian yang penulis lakukan dengan menggunakan metode lossless menghasilkan kualitas gambar baik tetapi untuk ukuran citra tidak terlalu berkurang pesat, maka dapat di simpulkan bahwa metode lossy lebih baik dibandingkan dengan metode lossless dalam mengkompresi byte dalam citra

Penulis menyarankan untuk penelitian selanjutnya bisa dikembangkan dengan menambahi metode atau algoritma yang berbeda

untuk lebih memperkecil ukuran file dan kualitas citra yang di kompresi lebih baik.

REFERENSI

- Rahmat, H., Damiri, D. J., & Susanto, A. (2012). Implementasi Kompresi Data Pada Jaringan Komputer Menggunakan Algoritma Zlib. *Jurnal Algoritma*, 9(1), 119–124. <https://doi.org/10.33364/algoritma/v.9-1.119>
- Saragih, N. E., & Harahap, F. (2019). *Perancangan Aplikasi Kompresi SMS dengan Algoritma Dynamic Markov Compression pada Android*. 7(2548–3528), 1–6.
- Hidayatullah, Muhammad Taufiq, Efri Suhartono, and Irma Safitri s t. 2019. "Kompresi Arithmetic Coding menggunakan teknik cs dan arithmetic compression coding using cs and combined technique of dct and svd method." *E-proceeding of engineering* 6 (1): 503–10.
- Ikhsan, Muhammad. 2016. "Implementasi Kompresi Citra Digital" 1 (979-458–924): 258–66.
- Pandi, Nauli Simangunsong, Hery Sunandar, Metode Deflate, and Hery Sunandar. 2015. "Kompresi Citra Berwarna Menggunakan Metode Deflate" 10 (2301–9425): 1–5.
- Pardosi, Irpan Adiputra, and Ali Akbar Lubis. 2019. "Analisis Kualitas Citra Hasil Reduksi Noise Menggunakan Spatial Median Filter Dan Adaptive Fuzzy Filter Terhadap Variasi Kedalaman Citra." *Indonesian Journal of Information Systems* 1 (2): 78. <https://doi.org/10.24002/ijis.v1i2.1939>.
- Raharja, Bayu Dwi, and Paulus Harsadi. 2018. "Implementasi Kompresi Citra Digital Dengan Mengatur Kualitas Citra Digital." *Jurnal Ilmiah SINUS* 16 (2): 71–77. <https://doi.org/10.30646/sinus.v16i2.363>.
- Rozi, Apri, Nelly Astuti Hasibuan, and Imam Saputra. 2019. "PERBANDINGAN METODE HIGH-BOOST FILTERING DAN." *Jurnal Pelita Informatika* 18 (2301–9425): 7–12.
- Saragih, Nidia Enjelita, and Fitriana Harahap. 2019. "Perancangan Aplikasi Kompresi SMS Dengan Algoritma Dynamic Markov Compression Pada Android" 7 (2548–3528): 1–6.
- Syahputra, Rahmad Eko, Deteksi Tepi, Metode Edge Linking, and Operator Sobel. 2019. "Dalam Citra Digital Dengan Metode." *Informatika, Jurnal Pelita* 18 (2301–9425): 62–68.
- Wahyuni, Meri Sri. 2019. "Analisa membandingkan hasil kompresi file mpeg-4 dan flv menggunakan algoritma huffman dan lz77." *Saintek itm* 32 (1): 40–47.