

KOMPRESI CITRA DENGAN METODE SCAN

Riko Arlando Saragih¹, Roy Rikki Hutahean²

¹Jurusan Teknik Elektro, Fakultas Teknik, Universitas Kristen Maranatha
Jl. Surya Sumantri 65 Bandung 40164
Telp. (022) 2012186 ext. 247

²Jurusan Teknik Elektro, Fakultas Teknik, Universitas Kristen Maranatha
Jl. Surya Sumantri 65 Bandung 40164
E-mail: riko_saragih@yahoo.com

ABSTRAK

Kompresi citra merupakan sebuah proses yang berkaitan dengan pengurangan alokasi bit yang diperlukan untuk merepresentasikan sebuah citra digital secara efisien. Hal ini dilakukan dengan menghilangkan bit redundansi (ekstra) yang terdapat pada citra digital tersebut, khususnya redundansi spasial. Dalam tulisan ini metode SCAN digunakan karena metode ini mampu memanfaatkan sifat redundansi spasial dengan memperhatikan pola SCAN. Hasil simulasi memperlihatkan bahwa rasio kompresi meningkat sebanding dengan ukuran citra yang akan dikompresi. Hal ini terjadi karena semakin besar ukuran citra yang akan dikompresi, maka semakin besar jumlah bit redundansi yang dapat dihilangkan (kami menggunakan citra grayscale dalam pengujian).

Kata Kunci: kompresi, metode SCAN, rasio kompresi, citra

1. PENDAHULUAN

Citra ialah fungsi intensitas cahaya dua-dimensi $f(x,y)$, dimana x dan y merupakan koordinat spasial (bidang), dan nilai f pada suatu titik (x,y) sebanding dengan kecerahan (brightness) dari citra di titik tersebut. Kecerahan yang dimaksud berasal dari sumber cahaya, dan cahaya adalah suatu bentuk energi. Citra dapat dilihat karena adanya berkas cahaya yang dipantulkan oleh benda-benda sekitarnya. Sumber cahaya menerangi sebuah obyek, lalu obyek tersebut memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh alat-alat optik, misalnya mata pada manusia, kamera, pemindai (*scanner*) dan sebagainya, sehingga bayangan objek yang disebut citra tersebut terekam. Citra sebagai keluaran dari suatu sistem perekam data dapat bersifat optik berupa foto, analog berupa sinyal video seperti gambar pada monitor televisi, atau digital yang dapat langsung disimpan pada suatu pita magnetik. Citra yang dimaksudkan di dalam tulisan ini adalah citra biasa (*still image*).

Setiap citra mempunyai karakteristik tersendiri, sehingga fitur tidak dapat bersifat general tetapi sangat tergantung pada model dan obyek gambar yang digunakan. Fitur dasar yang bisa diambil adalah warna, bentuk, dan tekstur. Fitur yang lebih kompleks menggunakan segmentasi, *clustering* dan *motion estimation*. Pada dasarnya, ada beberapa fitur dasar yang membentuk sebuah citra, di antaranya kecerahan dan kontras, *acuicity*, kontur, warna, bentuk, dan tekstur. (Wijaya, Marvin Ch & Agus Prijono, 2007)

Citra digital adalah citra dengan nilai $f(x,y)$ -nya didigitalisasikan (dibuat diskrit), baik untuk koordinat spasialnya maupun untuk kecerahannya. Digitalisasi dari koordinat spasial citra disebut

dengan *image sampling*, sedangkan digitalisasi dari derajat kecerahan citra disebut dengan *gray-level quantization*. Sebuah citra digital dapat dianggap sebagai suatu matriks, dimana baris dan kolomnya menunjukkan suatu titik di dalam citra, dan nilai elemen matriks tersebut menunjukkan derajat kecerahan di titik tersebut. Elemen-elemen dari citra digital tersebut biasanya disebut dengan *pixel* (piksel), yang merupakan singkatan dari *picture element*. Alokasi jumlah bit yang seragam (sama) untuk tiap piksel mengakibatkan adanya bit-bit ekstra (redundancy) yang sebenarnya bisa dihilangkan tetapi tidak menurunkan kualitas citra, sehingga memori (bandwidth) untuk menyimpan (mengirimkan) citra ini dapat dikurangi.

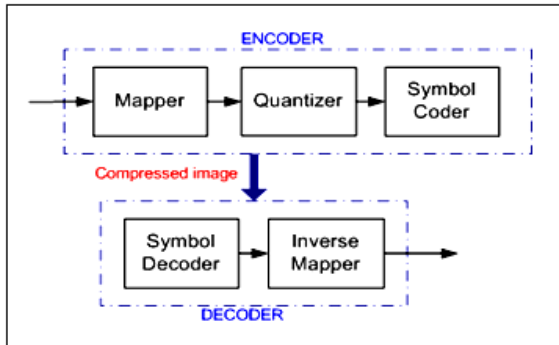
2. METODE SCAN

Kompresi citra berhubungan dengan proses pengurangan sejumlah bit yang dibutuhkan untuk merepresentasikan sebuah citra digital. Kompresi citra dilakukan dengan mengurangi jumlah bit yang dibutuhkan untuk merepresentasikan suatu citra dengan mengambil keuntungan dari bit-bit ekstra (*redundancy*) yang terdapat pada citra digital. Proses ini bertujuan untuk menghasilkan representasi citra yang membutuhkan memori yang kecil. Kompresi dicapai dengan penghilangan satu atau lebih dari tiga *basic redundancies data* yang terdapat pada citra digital, antara lain:

1. *Coding Redudancy* : timbul akibat kurang optimalnya pengkodean yang digunakan.
2. *Interpixel Redudancy* : timbul akibat korelasi di antara pixel-pixel dari citra.
3. *Psychovisual Redundancy*: timbul akibat adanya data yang tidak dapat diterima oleh sistem visual manusia.

Proses kebalikan dari proses kompresi ialah proses dekompresi (decoding), yang digunakan untuk merekonstruksi ulang data yang telah dikompresi.

Kompresi citra disusun oleh dua struktur blok yang berbeda, yaitu: Encoder dan Decoder, seperti diperlihatkan pada Gambar 1.



Gambar 1. Diagram Blok Sistem Kompresi Citra. Encoder bertanggung jawab mengurangi *interpixel redundancy* dan *psychovisual redundancy* dari citra masukan. Pada tahap pertama, *mapper* mengubah citra input ke dalam format/domain yang baru untuk mengurangi *interpixel redundancy*. Pada tahap kedua, blok *quantizer* mengurangi akurasi dari keluaran *mapper* sesuai dengan kriteria yang telah ditentukan sebelumnya. Pada tahap terakhir, *symbol decoder* menghasilkan kode untuk keluaran *quantizer* dan memetakan output tersebut sesuai dengan kode. Cara kerja *decoder* merupakan kebalikan dari blok-blok pada *encoder*.

Citra $f(x,y)$ menjadi masukan dari *encoder*, yang menghasilkan kumpulan simbol-simbol dari data input dan menggunakannya untuk merepresentasikan citra tersebut. Jika dimisalkan n_1 dan n_2 , dimana n_1 adalah sejumlah informasi (biasanya bit-bit) asli (masukan) dan n_2 sebagai citra yang telah dienkodkan, maka kompresi yang dapat dicapai, dapat dinilai secara angka melalui rasio kompresi sebagai berikut :

$$C_R = \frac{n_1}{n_2} \quad (1)$$

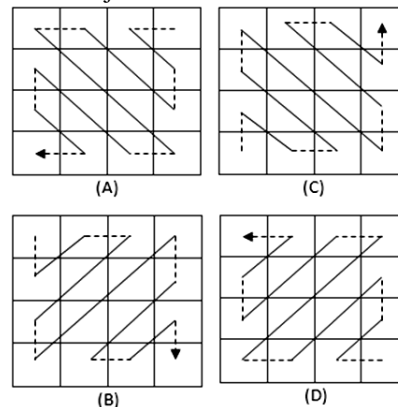
Metode kompresi citra dalam tulisan ini merupakan kompresi *lossless*. Artinya citra hasil dekompresi sama dengan citra semula sebelum dikompresi (citra masukan). (Mannicam, S.S. & N. Bourbakis, 2004). Menurut pendapat Chen, Chao S. dan Chen, Rong J. (2006), secara umum, proses kompresi dengan metode ini dilakukan melalui 3 tahap, yaitu :

1. Scanning and prediction

Pada tahap ini, citra masukan dibagi-bagi ke dalam blok-blok dengan ukuran $2^k \times 2^k$, $k \geq 2$. Penentuan ukuran blok sangat mempengaruhi kualitas kompresi. Ukuran blok yang lebih kecil mengurangi ratio kompresi karena ukuran blok yang lebih kecil menggunakan *scan path* yang yang lebih banyak sehingga membutuhkan bit-bit yang lebih banyak untuk mengenkodkan seluruh *scan path* tersebut.

Dalam tulisan ini menggunakan blok 32×32 . Setiap blok di-*scan* dengan 4 buah *scan path* (D_0 , D_1 , D_2 , dan D_3) dan proses ini dilakukan sambil menghitung nilai-nilai kesalahan prediksi dari piksel-piksel pada masing-masing blok, yang diprediksi menggunakan prediktor sesuai dengan jenis *scan path*-nya. Setelah itu, pada setiap *scan path* dihitung jumlah nilai absolut dari kesalahan-kesalahan prediksinya dan jumlah bit yang dibutuhkan untuk mengenkodkan *scan path* tersebut, sehingga setiap blok dapat dipilih satu buah *scan path* sebagai *scan path* terbaik, dimana *scan path* terbaik ialah *scan path* yang memperkecil nilai kesalahan prediksi dan jumlah untuk meng-*encode scan path* tersebut.

Seluruh jenis *scan path* yang digunakan dalam tulisan ini dapat dilihat pada Gambar 2 dan pediktor-prediktor yang digunakan pada keempat *scan path* tersebut disajikan dalam Tabel 1.

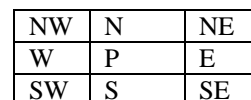


Gambar 2. Jenis-jenis Scan Path

Tabel 1. Prediktor-prediktor

No.	Predictor	Scan Path	Predictor Neighbors	Predictor rule
(A)	UR	D_0	{N,E}	$(N+E)/2$
(B)	UL	D_1	{N,W}	$(N+W)/2$
(C)	BL	D_2	{S,W}	$(S+W)/2$
(D)	BR	D_3	{S,E}	$(S+E)/2$

Pada perhitungan kesalahan prediksi, digunakan notasi-notasi untuk menamai piksel-piksel yang berdekatan seperti diperlihatkan pada Gambar 3.



Gambar 3. Notasi Piksel-piksel Tetangga p

Ada beberapa penamaan untuk piksel-piksel tertentu ketika dilakukan proses perhitungan kesalahan prediksi, antara lain:

- a) Piksel yang sedang di-*scan* (*current pixel*) disimbolkan dengan "p".

b) Piksel yang berfungsi sebagai *predictor neighbors* dan telah di-*scan* disimbolkan dengan “*q*” dan “*r*”.

c) Piksel yang telah di-*scan* sebelum piksel *p* disimbolkan dengan “*s*”.

Ada 2 pilihan cara yang akan dilakukan ketika menghitung nilai kesalahan prediksi pada setiap piksel *p*, yaitu:

a) Jika kedua *predictor neighbors*, *q* dan *r* ada, sehingga nilai kesalahan prediksi pada piksel *p* ialah:

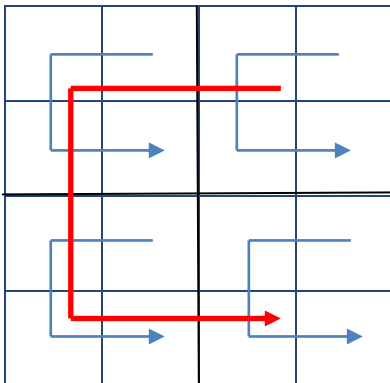
$$e = p - \frac{q+r}{2} \quad (2)$$

b) Jika kedua atau salah satu *predictor neighbors*, *q* dan *r* tidak ada, maka nilai kesalahan prediksi pada piksel *p* ialah:

$$e = p - s \quad (3)$$

c) Jika piksel *p* merupakan pixel pertama yang di-*scan* untuk keseluruhan citra masukan, maka nilai kesalahan prediksinya adalah nol.

Seluruh blok-blok dari citra masukan di-*scan* dengan urutan yang telah ditentukan (diperlihatkan pada Gambar 4), dimana setiap *region* dibagi ke dalam empat buah *subregion*. Urutan-urutan ini juga berlaku sama ketika proses dekompresi dilakukan, sehingga dapat diperoleh citra yang sama dengan citra masukan.



Gambar 4. Urutan blok-blok untuk seluruh *region* dan *subregion*

2. Scan Path Encoding

Pada tahap ini, setiap *scan path* terbaik dari setiap blok dari citra masukan, baik *region* maupun *subregion*, dikodekan ke biner. Jumlah bit yang dihasilkan pada tahap ini ditentukan jumlah *scan path* yang digunakan pada keseluruhan citra yang akan dikompresi.

3. Arithmetic Coding

Dalam tulisan ini menggunakan *Arithmetic Coding* yang dikembangkan oleh Eric Bodden, Malte Clasen, dan Joachim Kneis (2004). Metode tersebut menggunakan penskalaan dengan *range* terbatas, dimana pada setiap satu interval dibagi ke dalam 3 skala, yaitu skala E_1 , E_2 dan E_3 . Nilai E_1 terletak pada setengah bawah dari interval dan E_2 terletak pada setengah atas dari interval.

Apabila suatu simbol memiliki batas bawah (*lower*) dan batas atasnya (*upper*) terletak pada

kedua skala E_1 dan E_2 , maka batas-batas tersebut terletak pada skala E_3 . Pada ketiga skala tersebut memiliki aturan masing-masing, antara lain:

➤ Pada E_1 :- biner = ‘0’

- *lower* = 2 x *lower*

- *upper* = 2 x *upper*

➤ Pada E_2 :- biner = ‘1’

- *lower* = 2 x (*lower* - 0,5)

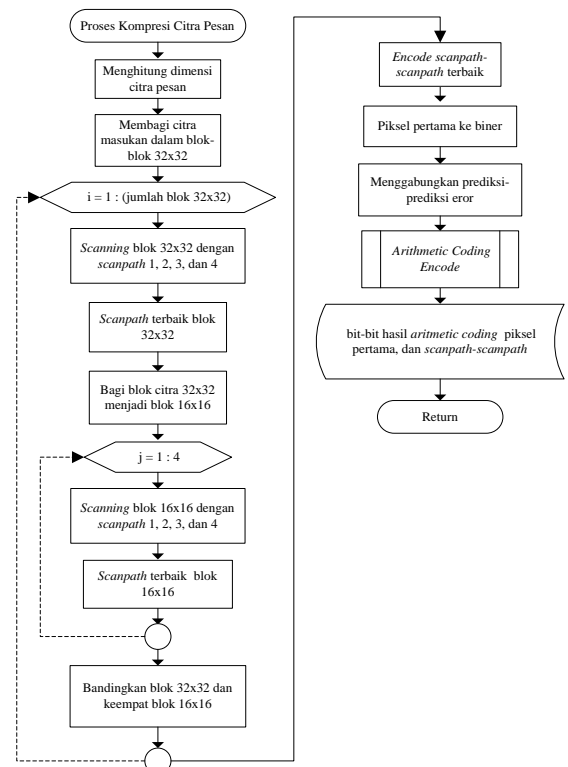
- *upper* = 2 x (*upper* - 0,5)

➤ Pada E_3 :

- *lower* = 2 x (*lower* - 0,5) x *prob* (simbol)

- *upper* = 2 x (*upper* - 0,5) x *prob* (simbol)

Diagram alir dari proses kompresi dengan menggunakan metode SCAN ini dapat dilihat pada Gambar 5 berikut :



Gambar 5. Diagram alir proses kompresi dengan metode SCAN

3. PEMBAHASAN

Lima buah citra grayscale 8 bit, format “*.bmp” dan ukurannya yang berbeda-beda digunakan sebagai citra masukan untuk pengujian metode kompresi ini. Kualitas kompresi citra dinilai dengan menghitung rasio kompresi pada masing-masing citra masukan. Karakteristik dari citra masukan yang akan diujikan secara umum sama, yaitu masing-masing berdimensi baik 64 x 64 piksel, 96 x 96 piksel, 128 x 128 piksel, dan 160 x 160 pikse serta alokasi bit per piksel adalah 8 bit. Pemilihan kelima citra ini untuk pengujian adalah karena kelima citra ini merupakan citra yang umum digunakan sebagai sampel untuk proses kompresi.

Rasio kompresi untuk citra masukan dengan dimensi 64 x 64 piksel, 96 x 96 piksel, 128 x 128 piksel, dan 160 x 160 piksel dapat dilihat berturut-turut dalam Tabel 2, Tabel 3, Tabel 4, dan Tabel 5.

Tabel 2. Rasio kompresi untuk citra masukan dengan dimensi 64 x 64 piksel

Citra Masukan	Rasio Kompresi
Lena.bmp	1,22
Pepper.bmp	1,17
Pirate.bmp	1,20
Walkbridge.bmp	1,16
Woman_blonde.bmp	1,24

Tabel 3. Rasio kompresi untuk citra masukan dengan dimensi 96 x 96 piksel

Citra Masukan	Rasio Kompresi
Lena.bmp	1,37
Pepper.bmp	1,34
Pirate.bmp	1,32
Walkbridge.bmp	1,23
Woman_blonde.bmp	1,40

Tabel 4. Rasio kompresi untuk citra masukan dengan dimensi 128 x 128 piksel

Citra Masukan	Rasio Kompresi
Lena.bmp	1,48
Pepper.bmp	1,47
Pirate.bmp	1,38
Walkbridge.bmp	1,27
Woman_blonde.bmp	1,49

Tabel 5. Rasio kompresi untuk citra masukan dengan dimensi 160 x 160 piksel

Citra Masukan	Rasio Kompresi
Lena.bmp	1,54
Pepper.bmp	1,56
Pirate.bmp	1,42
Walkbridge.bmp	1,28
Woman_blonde.bmp	1,54

Berdasarkan Tabel 2, 3, 4, dan 5 dapat dilihat bahwa rasio kompresi yang dihasilkan relatif meningkat sebanding dengan bertambahnya dimensi citra masukan yang digunakan. Hal ini terjadi karena semakin besar dimensi citra yang digunakan, maka jumlah bit ekstra yang dapat dihilangkan semakin banyak. Hal ini juga menunjukkan bahwa secara spasial nilai piksel antar piksel yang bertetangga sama (hampir sama).

Pengujian secara obyektif juga dilakukan terhadap hasil dekompresi dari kelima gambar masukan yang sudah dikompresi sebelumnya dengan mengedarkan angket terhadap beberapa orang sebagai responden dan dihitung nilai *Mean Opinion Score* (MOS) terhadap hasil angket. Hasilnya dapat dilihat dalam Tabel 6.

Tabel 6. Nilai MOS terhadap hasil dekompresi

Citra Masukan	MOS
Lena.bmp	4,625
Pepper.bmp	4,75
Pirate.bmp	4,75
Walkbridge.bmp	5
Woman_blonde.bmp	4,875

Nilai MOS yang hampir mendekati 5 (excellent) memperlihatkan bahwa secara kualitatif citra hasil kompresi tidak terlihat berbeda dengan citra masukan.

4. KESIMPULAN

Metode SCAN terbukti dapat melakukan kompresi citra karena mampu menghilangkan beberapa bit ekstra yang terdapat pada citra sehingga ukuran citra yang sudah dikompresi dapat direduksi (rasio kompresi lebih besar daripada satu). Untuk pengembangan lebih lanjut dapat diujikan mungkin metode kompresi ini dapat diujikan untuk citra berwarna dan masukan berupa video.

PUSTAKA

Bodden, Eric., Clasen, Malte., dan Kneis, Joachim. (2004). *Arithmetic Coding Revealed*.

Chen, Chao S. & Chen, Rong J. (2006). *Image Encryption and Decryption Using SCAN Methodology*, IEEE.

Mannicam, S.S. & N. Bourbakis. (2004). *Lossless Compression and Hiding Image*. *Pattern Recognition Letters* 37 (2004) 475 - 486.

Wijaya, Marvin Ch & Agus Prijono. (2007). *Pengolahan Citra Digital Menggunakan MatLAB*. Bandung: Informatika.