

## APLIKASI KOMPRESI CITRA BERBASIS ROUGH FUZZY SET

Anny Yuniarti<sup>1</sup>, Nadya Anisa Syafa<sup>2</sup>, Handayani Tjandrasa<sup>3</sup>

<sup>1,2,3</sup>Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember (ITS) Surabaya  
Surabaya 60111 Telp (031)- 5939214  
e-mail: [anny@if.its.ac.id](mailto:anny@if.its.ac.id)

### Abstrak

Perkembangan teknologi informasi yang pesat telah memberi peran yang sangat penting untuk menjalin pertukaran informasi yang cepat. Kecepatan pengiriman informasi akan menjadi bagian utama dalam pengiriman informasi masa mendatang. Salah satu cara untuk mempersingkat waktu dan memperkecil biaya pengiriman adalah dengan melakukan pemampatan data teks, suara dan citra sebelum ditransmisikan. Dalam makalah ini, diperkenalkan suatu metode kompresi citra baru berbasis rough fuzzy set yang menggunakan metode kuantisasi vektor. Metode ini dinamakan Rough Fuzzy Vector Quantization (RFVQ). Dengan menggunakan RFVQ diperoleh hasil kompresi yang baik karena citra yang dihasilkan serupa dengan citra aslinya. Metode ini disusun dari tiga fase yaitu perancangan codebook, fase coding, dan fase decoding. Uji coba dilakukan pada citra abu-abu 8 bit per piksel berukuran 256 x 256. Hasil ujicoba ditunjukkan dengan nilai Peak Signal to Noise Ratio (PSNR) dengan blok dimensi dan tingkat kompresi tertentu. Tingkat kompresi yang bagus untuk menghasilkan citra kompresi dengan metode ini adalah tingkat kompresi 0.25.

**Kata kunci:** Rough fuzzy set, Block Coding, Image Coding, Kompresi Citra, Multimedia

### 1. PENDAHULUAN

Saat ini pengiriman informasi secara *real-time* masih mengalami kendala. Salah satu kendalanya adalah besar jumlah data yang harus dikirim melebihi kecepatan transmisi yang dimiliki oleh perangkat keras yang ada. Hal ini menyebabkan masih terdapat waktu *delay* yang relatif besar. Salah satu solusi untuk mempersingkat waktu dan memperkecil biaya pengiriman data tersebut adalah dengan melakukan pemampatan data teks, suara dan citra (*image*) sebelum ditransmisikan.

Di sisi lain, kuantisasi vektor telah banyak diteliti untuk mengurangi tingkat transmisi bit atau mengurangi tempat penyimpanan sinyal suara dan sinyal citra serta untuk proses *coding* citra. Pada proses *coding* citra, citra dipartisi menjadi blok-blok persegi kecil dan fitur-fitur setiap blok diekstraksi menjadi sebuah vektor tunggal. Setiap vektor tunggal tersebut selanjutnya dibandingkan dengan sekumpulan prototipe vektor standar yang disimpan dalam sebuah *codebook*. Proses perbandingan ini mengembalikan sebuah indeks pada *codebook* yang merepresentasikan *codeword* yang paling cocok dengan vektor input. Nilai-nilai indeks inilah yang nantinya ditransmisikan menggantikan data citra yang ukurannya jauh lebih besar daripada nilai-nilai indeks tersebut. Selanjutnya pihak penerima yang menerima data transmisi berupa data indeks akan merekonstruksi citra dengan menggunakan *codeword* sesuai indeks yang ditransmisikan.

Oleh karena pemrosesan setiap blok dilakukan secara independen, proses rekonstruksi citra dapat menghasilkan diskontinuitas pada citra hasil. Untuk beberapa aplikasi, misalnya aplikasi citra medis, hal ini dapat menyebabkan informasi penting pada citra yang dihasilkan hilang. Penelitian ini bertujuan untuk menerapkan salah satu metode untuk mengatasi efek *blocking* tersebut melalui pendekatan metode *block coding* yang memanfaatkan informasi antar blok. Dengan metode tersebut diharapkan citra yang dihasilkan dari proses *coding* tidak kehilangan informasi yang penting. Salah satu aplikasi dari metode yang berbasis *rough fuzzy set* ini adalah untuk aplikasi kompresi citra.

### 2. TINJAUAN PUSTAKA

#### 2.1. Rough Set

Teori *rough set* didasarkan pada terminologi yang terdapat pada ruang aproksimasi, yakni sepasang nilai  $\langle X, R \rangle$ , dimana  $X$  adalah sebuah himpunan tidak kosong (himpunan semesta), dan  $R$  adalah relasi ekuivalen pada  $X$ , artinya  $R$  adalah yaitu sebuah relasi refleksif, simetris, dan transitif. Relasi  $R$  mendekomposisi  $X$  menjadi kelas-kelas terpisah sedemikian hingga dua elemen  $x$  dan  $y$  terdapat pada sebuah kelas yang sama jika dan hanya jika  $(x, y) \in R$  atau ekuivalen dengan  $xRy$ .  $X/R$  didefinisikan sebagai himpunan hasil bagi  $X$  terhadap relasi  $R$ , yaitu:

$$X/R = \{T_1, \dots, T_i, \dots, T_p\} \quad (1)$$

dimana  $T_i = [x]_R$  adalah sebuah kelas ekuivalen dari  $R$ ,  $i = 1, \dots, p$ , yakni kelas elemen  $x, y \in X$  sedemikian hingga  $xRy$ . Jika dua elemen  $x$  dan  $y$  didalam  $X$  berada pada kelas ekuivalen  $T_i \in X/R$  yang sama, maka dua elemen tersebut dikatakan *indistinguishable* (Alfredo Petrosino dan Alessio Ferone, 2009).

Teori *rough set* digunakan untuk merepresentasikan himpunan yang batasnya kurang jelas dengan cara memberikan pendekatan bawah dan pendekatan atas dari himpunan tersebut (Arif Rizal, 2008). Himpunan  $T$  dapat dikarakteristikan sebagai sepasang himpunan aproksimasi (Zdzisław Pawlak, 1982):

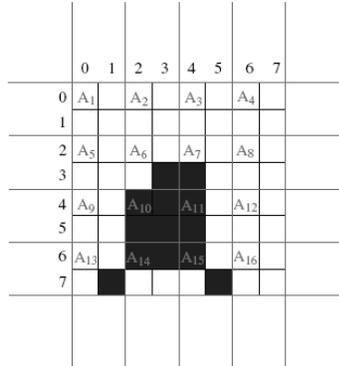
$$RS^*(T) = \{[x]_R \mid [x]_R \cap T \neq \emptyset\} \quad (2)$$

$$RS_*(T) = \{[x]_R \mid [x]_R \subseteq T\} \quad (3)$$

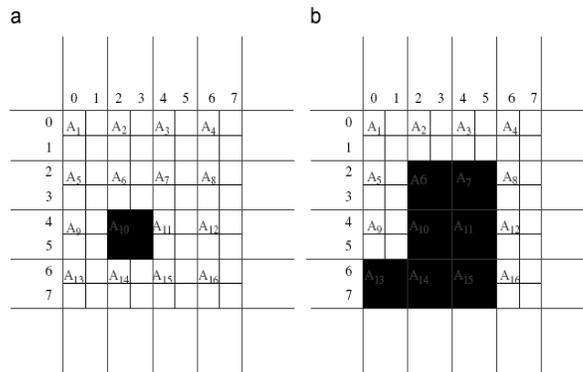
$RS^*(T)$  dan  $RS_*(T)$  merupakan pendekatan atas dan pendekatan bawah nilai  $T$  oleh  $R$ :

$$RS_*(T) \subseteq T \subseteq RS^*(T) \quad (4)$$

Interval  $[RS_*(T), RS^*(T)]$  disebut sebagai *rough set* dari  $T$ . Contoh pendekatan atas dan pendekatan bawah obyek  $T$  pada sebuah citra biner ditunjukkan pada Gambar 1 dan 2.



**Gambar 1. Contoh citra dengan obyek  $T$**   
 (Alfredo Petrosino dan Alessio Ferone, 2009)



**Gambar 2. Pendekatan bawah (a) dan atas (b) dari obyek  $T$**   
 (Alfredo Petrosino dan Alessio Ferone, 2009)

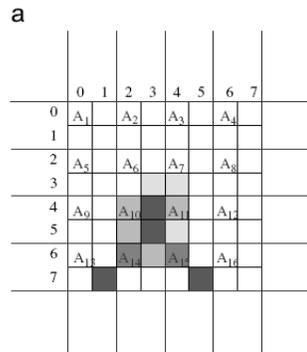
## 2.2. Himpunan Fuzzy (Fuzzy Set)

Teori *fuzzy set* merupakan pengembangan dari teori set (biasa) atau *crisp set*. Perbedaan antara *fuzzy set* dan *crisp set* terletak pada keanggotaan suatu obyek. Pada *crisp set* suatu obyek hanya mempunyai dua kemungkinan keanggotaan yaitu anggota himpunan (1) atau bukan anggota himpunan (0), sedangkan pada *fuzzy set* tingkat keanggotaan elemen berada pada interval  $[0,1]$  (L. A. Zadeh, 1965).

Apabila  $X$  adalah himpunan semesta, *fuzzy set*  $F$  dari  $X$  adalah:

$$\mu_F : X \rightarrow [0, 1]$$

yang menghubungkan setiap  $x \in X$  dengan derajat keanggotaan  $x$  didalam  $F$ . Contoh citra *fuzzy set* dapat dilihat pada Gambar 3 dan Tabel 1.



Gambar 3. Contoh citra fuzzy set objek T

Tabel 1. Nilai tingkat keanggotaan elemen objek T

Nilai Tingkat Keanggotaan Objek T					
x	$\mu_F(x)$	x	$\mu_F(x)$	x	$\mu_F(x)$
(3,3)	0.2	(5,2)	0.3	(6,4)	0.6
(3,4)	0.2	(5,3)	0.7	(7,1)	0.7
(4,2)	0.3	(5,4)	0.2	(7,5)	0.7
(4,3)	0.7	(6,2)	0.6		
(4,4)	0.3	(6,3)	0.3		

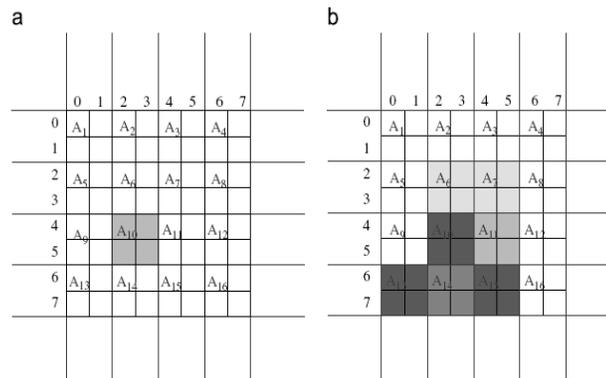
### 2.3. Himpunan Rough Fuzzy (Rough Fuzzy Set)

Teori *rough fuzzy set* merupakan teori gabungan dari *rough set* dan *fuzzy set*. Sehingga citra yang dihasilkan memiliki pendekatan bawah dan atas serta memiliki tingkat keanggotaan elemen antara 0 sampai 1:

$$m_i = \inf \{f(x) \mid x \in X_i\} \quad (5)$$

$$M_i = \sup \{f(x) \mid x \in X_i\} \quad (6)$$

dimana  $m_i$  adalah pendekatan bawah dari citra *fuzzy set*, sedangkan  $M_i$  adalah pendekatan atas dari citra *fuzzy set*. Citra yang dihasilkan dari teori ini dapat dilihat pada Gambar 4.

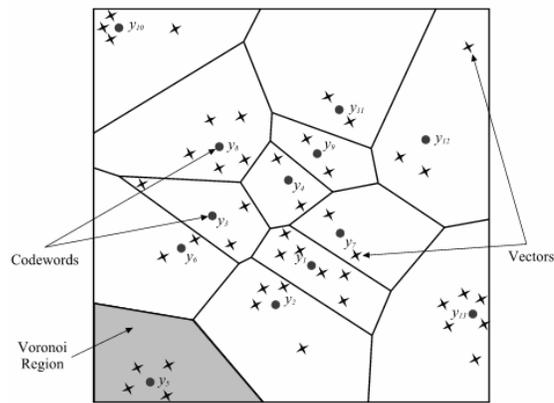


Gambar 4. Pendekatan bawah (a) dan atas (b) dari objek citra fuzzy set T  
 (Alfredo Petrosino dan Alessio Ferone, 2009)

### 2.4. Kuantisasi Vektor

Kuantisasi vektor adalah metode kompresi data *lossy* berdasarkan prinsip blok pengkodean. Sebuah kuantisasi vektor memetakan vektor k-dimensi pada ruang vektor  $R^k$  ke dalam vektor himpunan terbatas  $Y = \{y_i; i = 1, 2, \dots, N\}$ . Kuantisasi vektor dapat dipresentasikan dengan persamaan (7) dan diilustrasikan dalam Gambar 5 (Mohamed Qasem, 2010).

$$V_i = \{x \in R^k \mid \|x - y_i\| \leq \|x - y_j\|, j \neq i\} \quad (7)$$



Gambar 5. Ilustrasi kuantisasi vektor

Dalam kuantisasi vektor, setiap vektor  $y_i$  disebut vektor kode atau *codeword* dan himpunan dari semua *codeword* disebut *codebook*. Setiap *codeword* yang saling berhubungan dengan daerah tetangga terdekat disebut daerah *voronoi*. Sebuah kuantisasi vektor memiliki dua operasi: *encoder* dan *decoder*. *Encoder* mengambil indeks sebuah *codeword* dengan cara mengevaluasi jarak terdekat antara vektor masukan dengan masing-masing *codeword* di *codebook*. Setelah *codeword* terdekat ditemukan, indeks dari *codeword* dikirimkan. *Decoder* bertugas menerima sandi indeks dari *codeword*, kemudian sandi tersebut digantikan dengan *codeword* terkait.

### 3. METODE PENELITIAN

Secara umum metode *Rough Fuzzy Vector Quantization* (RFVQ) tersusun dari tiga fase, yaitu perancangan *codebook*, *coding*, dan *decoding*. Fase perancangan *codebook* membutuhkan citra sejumlah  $N$  untuk menghasilkan sebuah *codebook* yang digunakan untuk data masukan dari proses *coding* dan *decoding*. Sedangkan fase *coding* hanya membutuhkan sebuah citra dan *codebook* sebagai data masukan untuk menghasilkan citra berkode yang akan digunakan sebagai data masukan fase *decoding*. Dari sebuah *codebook* dan citra berkode yang dihasilkan dari dua fase sebelumnya, fase *decoding* menghasilkan citra yang sudah terkompresi. Citra inilah yang menjadi luaran dari sistem ini. Proses diagram alir sistem secara umum dapat dilihat pada Gambar 6.

#### 3.1. Perancangan Codebook

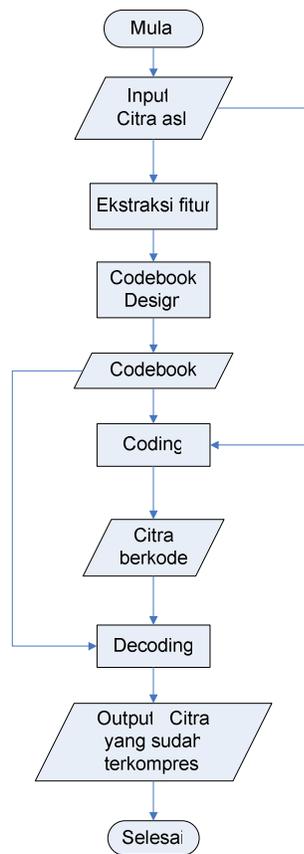
Fase perancangan *codebook* membutuhkan citra sejumlah  $N$  yang berukuran  $H \times W$  sebagai data masukan. Kemudian nilai tingkat keabuan dikelompokkan ke dalam beberapa tingkat keanggotaan *fuzzy set*. Lalu ditentukan nilai-nilai pendekatan batas bawah dan atas tiap piksel dari blok dimensi  $H_B \times W_B$ . Kemudian nilai-nilai pendekatan batas bawah dan atas tersebut disimpan ke dalam vektor *array* 1 dimensi. Setelah itu dijalankan algoritma *K-Means clustering* untuk menghasilkan sebuah *codebook* yang merupakan luaran dari fase ini. Seluruh proses pada fase ini terdapat pada Gambar 7.

#### 3.2. Coding

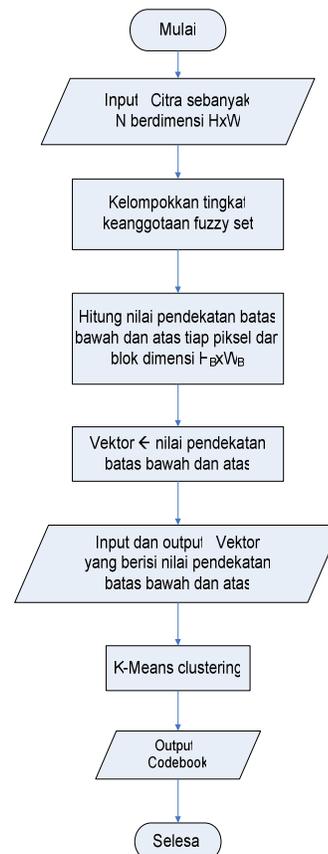
Fase *coding* membutuhkan sebuah citra berukuran  $H \times W$  dan *codebook* sebagai data masukan. Citra masukan diproses seperti pada fase perancangan *codebook* sampai dihasilkan sebuah vektor *array* 1 dimensi berisi nilai pendekatan batas bawah dan atas. Setelah itu dicari *codeword* terdekat dengan vektor yang terdapat pada *codebook* menggunakan rumus jarak *Euclidean*. Hasil pencarian *codeword* terdekat disimpan di tiap blok dimensi sehingga dihasilkan sebuah citra berkode yang merupakan luaran dari fase ini. Seluruh proses pada fase ini terdapat pada Gambar 8.

#### 3.3. Decoding

Fase *decoding* membutuhkan sebuah citra berkode dan *codebook* sebagai data masukan. Fase ini bertujuan untuk merekonstruksi kembali citra berkode dengan mencocokkan nomor indeks *codeword* pada *codebook* sehingga didapatkan kembali citra yang serupa dengan citra aslinya. Proses pertama dari fase ini adalah mencocokkan kode dari citra berkode dengan *codeword* dari *codebook* di tiap blok dimensi citra. Kemudian mendapatkan nilai vektor yang berisi pendekatan batas bawah dan atas lalu menyimpannya. Setelah itu mengganti nilai piksel sisi pojok kanan atas dan kiri bawah dengan rata-rata dari pojok kiri atas dan kanan bawah untuk setiap nilai vektor. Lalu mengambil nilai piksel yang paling tinggi diantara blok dimensi vektor pendekatan batas bawah dan mengambil nilai piksel yang paling rendah diantara blok dimensi vektor pendekatan batas atas. Terakhir adalah mengganti nilai piksel dengan nilai rata-rata vektor pendekatan batas bawah dan atas. Luaran dari fase ini adalah sebuah citra yang sudah terkompresi. Seluruh proses pada fase ini diilustrasikan pada Gambar 9.



Gambar 6. Diagram alir model sistem secara umum



Gambar 7. Diagram alir fase perancangan codebook

#### 4. HASIL DAN PEMBAHASAN

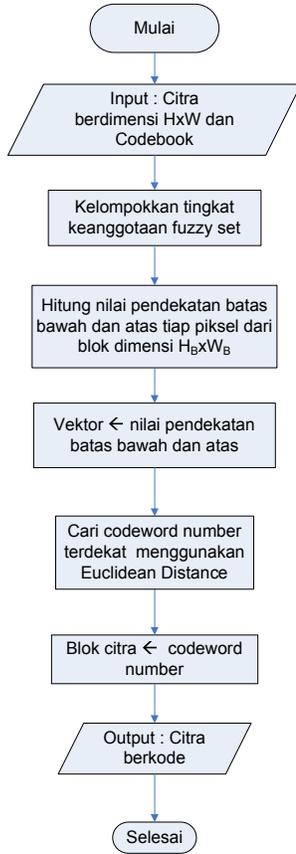
Pada penelitian ini digunakan citra abu-abu 8 bit per piksel berukuran 256 x 256. Untuk perancangan *codebook* digunakan 40 data. Parameter tingkat kompresi didasarkan pada jumlah kluster dan blok dimensi seperti pada Tabel 2.

Tabel 2. Data tingkat kompresi

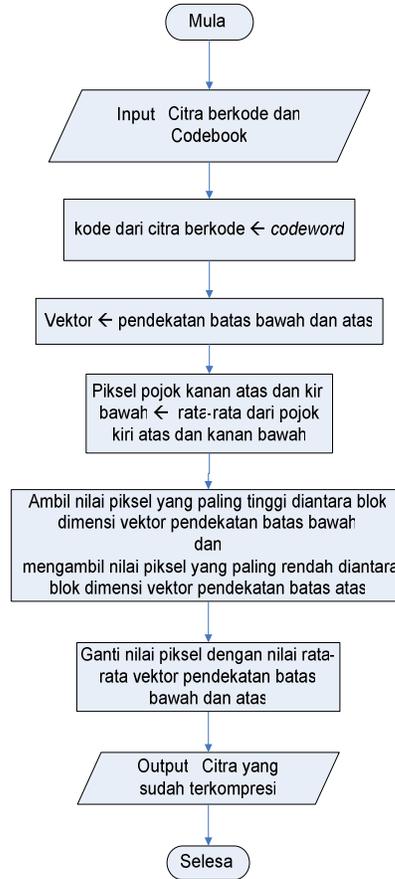
Tingkat Kompresi	Jumlah Kluster	Blok Dimensi
0.03	16	4
0.06	256	4
0.14	32	2
0.25	256	2
0.44	16384	2

Ada tiga uji coba yang dilakukan. Uji coba pertama yaitu membandingkan hasil kompresi terhadap citra pembuat *codebook* dengan jumlah yang berbeda. Kedua adalah membandingkan data uji coba dengan nilai *fuzzy set* yang berbeda-beda. Ketiga adalah membandingkan data uji coba dengan tingkat kompresi yang berbeda-beda. Hasil dari ketiga percobaan tersebut dapat dilihat pada Gambar 10-12.

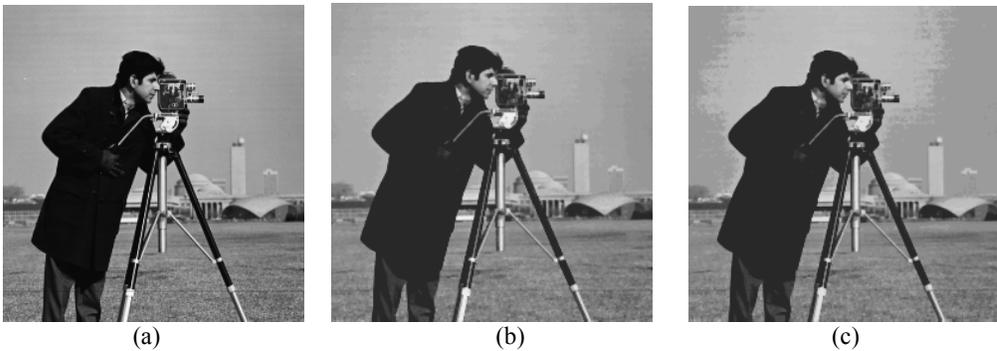
Selain citra hasil kompresi, hasil uji coba juga ditunjukkan dengan nilai PSNR tiap percobaan. Nilai PSNR percobaan 1-3 dapat dilihat pada Tabel 3.



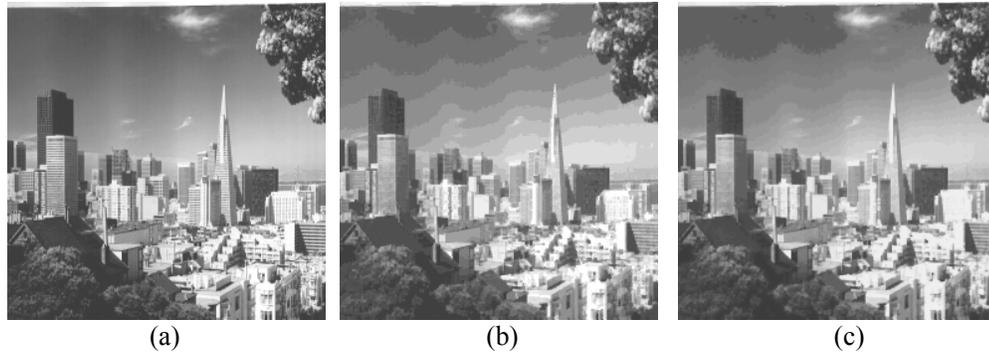
Gambar 8. Diagram alir fase *coding*



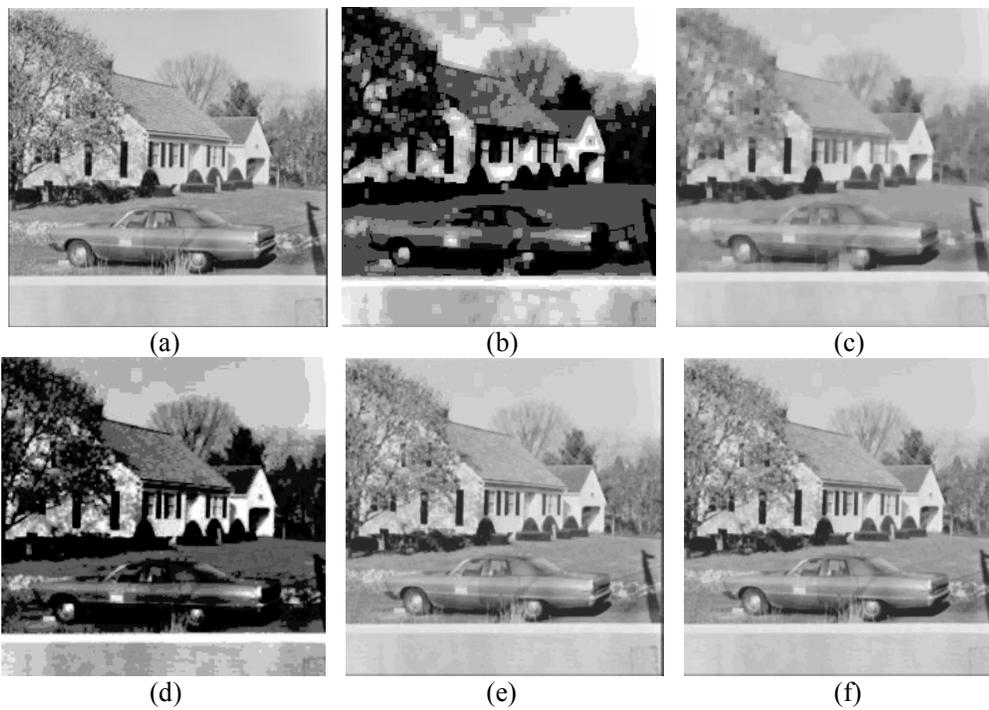
Gambar 9. Diagram alir fase *decoding*



Gambar 10. Uji coba terhadap jumlah citra pembuat *codebook*. (a) citra asli; (b) citra hasil uji coba dengan 40 citra pembuat *codebook*; (c) citra hasil uji coba dengan 10 citra pembuat *codebook*



Gambar 11. Uji coba terhadap nilai *fuzzy set* berbeda (a) citra asli; (b) citra hasil uji coba dengan *fuzzy set* 16; (c) citra hasil uji coba dengan *fuzzy set* 32



Gambar 12. Uji coba terhadap tingkat kompresi berbeda. (a) citra asli; (b) citra hasil uji coba dengan tingkat kompresi 0.03; (c) 0.06; (d) 0.14; (e) 0.25; (f) 0.44

Tabel 3. PSNR RFVQ pada percobaan 1-3

	Percobaan	PSNR
1	40 citra pembuat codebook	19.82
	10 citra pembuat codebook	19.73
2	fuzzy set 16	19.63
	fuzzy set 32	19.94
3	Tingkat kompresi 0.03	16.07
	Tingkat kompresi 0.06	17.75
	Tingkat kompresi 0.14	17.19
	Tingkat kompresi 0.25	21.71
	Tingkat kompresi 0.44	21.69

## 5. KESIMPULAN

Pada pengujian yang dilakukan dengan perangkat lunak yang menggunakan metode *Rough Fuzzy Vector Quantization* (RFVQ), dihasilkan bahwa hal-hal yang dapat mempengaruhi citra hasil kompresi adalah jumlah citra pembuat *codebook*, tingkat keanggotaan *fuzzy set*, dan blok dimensi. Semakin banyak jumlah citra pembuat *codebook*, semakin bagus citra kompresi yang dihasilkan. Semakin tinggi tingkat keanggotaan *fuzzy set*, semakin bagus citra kompresi yang dihasilkan. Kedua hal tersebut disebabkan oleh nilai tingkat keabuan yang mengisi *codebook* menjadi semakin banyak dan beragam. Pada metode RFVQ, semakin kecil blok dimensi, semakin bagus citra kompresi yang dihasilkan. Sedangkan pada metode JPEG, semakin besar blok dimensi, semakin bagus citra kompresi yang dihasilkan. Selain tiga hal tadi, pemilihan *codeword* yang optimal pada *codebook* juga mempengaruhi keberhasilan citra kompresi yang dihasilkan dari metode RFVQ ini. Tingkat kompresi yang bagus untuk menghasilkan citra kompresi hasil metode ini adalah tingkat kompresi 0.25. Dari sini, dapat disimpulkan bahwa semakin kecil blok dimensi dan semakin besar jumlah kluster yang digunakan semakin bagus citra kompresi yang dihasilkan.

Dari sifat-sifat metode RFVQ tadi, dapat disimpulkan bahwa metode ini lebih cocok untuk proses *streaming*. Hal ini disebabkan oleh proses *streaming* membutuhkan kecepatan pengiriman data. Dengan menggunakan metode ini, dapat menghemat waktu pengiriman data. Sebab, proses *coding* dapat dilakukan pada sisi *server*, sedangkan proses *decoding* dilakukan oleh klien. Proses *coding* dilakukan *server* untuk mengkompresi data sebelum dikirimkan ke klien melalui Internet sehingga dihasilkan data berkode, dan *decoding* dilakukan oleh klien untuk merekonstruksi data berkode sehingga ditampilkan data tanpa kompresi. Sehingga hanya kode data saja yang ditransmisikan. Dengan begitu proses pengiriman data akan semakin cepat bila dibandingkan jika mengkompresi data dengan mengirimkan satu persatu isi data tersebut. Hal ini terbukti dari tingkat kompresi yang tinggi, citra yang dihasilkan dari metode ini juga semakin bagus.

## 6. DAFTAR PUSTAKA

- Alfredo Petrosino dan Alessio Ferone, 2009, *Rough fuzzy set-based image compression*, Fuzzy Sets and Systems 160, Elsevier, halaman 1485–1506.
- Arif Rizal, 2008, *Diskritisasi informasi dalam teori rough set menggunakan immune algorithm*, Laporan Tugas Akhir Jurusan Sistem Informasi, ITS, (URL: <http://digilib.its.ac.id/detil.php?id=4105&q=rough%20set>).
- L. A. Zadeh, 1965, *Fuzzy Sets*, Information and Control 8, halaman 338–353.
- Mohamed Qasem, 2010, *Vector Quantization*, (URL: <http://www.mqasem.net/vectorquantization/vq.html>).
- Zdzislaw Pawlak, 1982, *Rough sets*, International Journal of Computer and Information Sciences 11, halaman 341–356.