

FRAKTAL DAN INVERS FRAKTAL

Beni Utomo

*Laboratorium Matematika Komputasi
Jurusan Informatika, Sekolah Tinggi Teknologi Bontang*

Abstract: Fraktal sebagai sebuah geometri bisa digambarkan pada ruang dimensi dua dengan syarat yang harus dipenuhi yaitu bisa didefinisikan pada ruang metrik lengkap. Untuk mengkonstruksi fraktal secara geometri tersebut diperlukan suatu himpunan fungsi yang dikenal dengan nama IFS dan jika IFS tersebut diiterasikan maka akan terbentuk suatu citra yang dikenal sebagai citra fraktal, artinya setiap ada IFS maka selalu bisa dikonstruksi citra fraktal. Invers fraktal merupakan balikan dari proses tersebut, yaitu jika diberikan sebarang citra fraktal maka bisakah ditemukan IFSnya. Jaminan adanya IFS tersebut dikenal dengan nama Teorema Collage.

Keywords: ruang fraktal, fraktal, IFS, invers, teorema collage.

1. Pendahuluan

Konstruksi geometri fraktal secara matematis antara lain terdapat dalam buku Barnsley [1] yang secara singkat bisa diuraikan sebagai berikut. Diberikan ruang metrik kompak (X, d) dan didefinisikan $H(X) = \{A \subset X | A \text{ kompak atau } A = \emptyset\}$. Didefinisikan fungsi $h : H(X) \times H(X) \rightarrow R^+$ dengan $h(A, B) = \text{Maks}(d(A, B), d(B, A))$ dan $d(A, B) = \text{maks}\{\text{Min}\{d(x, y) : y \in B\} : x \in A\}$ untuk setiap $A, B \in H(X)$. Fungsi h merupakan metrik yang disebut metrik Hausdorff dan $(H(X), h)$ merupakan ruang metrik lengkap dan sering disebut ruang fraktal.

Sistem Fungsi Iterasi atau Iterated Function System (IFS) didefinisikan sebagai suatu sistem yang terdiri dari ruang metrik lengkap (X, d) dan pemetaan-pemetaan kontraksi $w_n : X \rightarrow X$ dengan faktor kontraksi $s_n, 0 < s_n < 1$ untuk $n = 1, 2, 3, \dots, N$. Dalam hal ini IFS diberi notasi $\{X; w_1, w_2, \dots, w_n\}$ dengan faktor kontraksi $s = \text{Maks}\{s_n : n = 1, 2, 3, \dots, N\}$. Akibatnya terdapat dengan tunggal titik tetap $A \in H(X)$ dengan $A = W(A) = \bigcup_{n=1}^N w_n(A)$ sehingga A dapat dicari secara iteratif oleh $A = \lim_{n \rightarrow \infty} W^n(B)$ untuk setiap $B \in H(X)$. Titik tetap ini disebut atraktor (*attractor*) IFS. Untuk $X = R^2$, maka sudah ditemukan 3 algoritma untuk mencari atraktor tersebut, yaitu algoritma deterministik, algoritma iterasi random, dan algoritma *cat game* atau *chaos game* (Widodo [20]). Pada saat mengkonstruksi geometri fraktal, terlebih dahulu dimiliki IFS, dan jika IFS tersebut diiterasikan maka akan terbentuk suatu citra geometri yang disebut citra fraktal. Dengan bantuan komputer grafik, atraktor dapat digambarkan dengan baik dan ternyata memberikan keindahan fraktal yang sangat menakjubkan seperti bentuk *Fern*, Segitiga Sierpinski, *Sponge* dan sebagainya. Akibatnya bisa dikatakan bahwa jika dimiliki sebarang IFS maka citra fraktal bisa dibentuk. Pertanyaan selanjutnya adalah bagaimana jika kejadian tersebut dibalik. Jika dimiliki sebarang citra awal, apakah mungkin ditemukan suatu IFS yang jika IFS tersebut diiterasikan akan membentuk suatu citra fraktal, dengan sifat citra fraktal tersebut mirip dengan citra awal. Teorema Collage (Barnsley [1]) menjamin bahwa IFS dengan sifat demikian mungkin untuk dibentuk.

2. Tinjauan Teori

Teori dasar yang digunakan makalah ini adalah ruang metrik dan fraktal. Ruang metrik memberikan dasar konstruksi fraktal yang dikonstruksi melalui proses iteratif.

Definisi 1.1. Diberikan ruang metrik (X, d) dan transformasi $f : X \rightarrow X$. Iterasi maju dari f merupakan transformasi $f^{\circ n}(x) = x$ dengan $f^{\circ 0}(x) = x$, $f^{\circ 1}(x) = f(x)$, $f^{\circ(n+1)}(x) = f \circ f^{\circ n}(x) = f(f^{\circ n}(x))$ untuk $n = 0, 1, 2, \dots$. Jika f invertibel maka bisa didefinisikan iterasi mundur f yaitu $f^{\circ(-m)}(x) : X \rightarrow X$ dengan $f^{\circ(-1)}(x) = f^{-1}(x)$, $f^{\circ(-m)}(x) = (f^{\circ m})^{-1}(x)$ untuk $m = 1, 2, 3, \dots$

Definisi 1.2. Pemetaan Kontraksi (Barnsley [1]). Suatu transformasi $f : X \rightarrow X$ pada ruang metrik (X, d) disebut transformasi kontraktif atau pemetaan kontraksi jika terdapat suatu konstanta s , dengan $0 \leq s < 1$ sehingga berlaku $d(f(x), f(y)) \leq (s) \cdot d(x, y)$ untuk setiap $x, y \in X$.

IFS mengkonstruksi fraktal dengan cara mengulang transformasi berkali-kali untuk sebarang pola awal. Pola awal ditransformasi menjadi suatu pola berulang dengan struktur yang sama pada detail tertentu, hal tersebut merupakan karakteristik dasar dari himpunan fraktal.

Teori transformasi fraktal merupakan teori IFS lokal. Meskipun IFS lokal merupakan teori yang cukup rumit untuk menemukan fraktal, namun pada kenyataannya IFS lokal bisa menyederhanakan proses. Suatu transformasi global pada ruang X merupakan suatu transformasi yang didefinisikan untuk semua titik anggota X ; sementara transformasi lokal merupakan transformasi dengan domainnya adalah himpunan bagian dari X dan transformasi tersebut tidak perlu dikenakan pada semua anggota X , sehingga pengertian IFS lokal merupakan IFS yang dikenakan tidak pada semua domain tetapi hanya dikenakan pada domain yang merupakan himpunan bagian dari ruang yang dipakai.

Ide menemukan fraktal seperti disebutkan sebelumnya adalah menemukan ruang bagian dari ruang fraktal yang bisa diperbaharui menggunakan IFS. Jika mungkin, satu IFS dapat digunakan untuk menggantikan beberapa IFS untuk membuat fraktal yang sama, hal tersebut lebih efisien dan hemat karena jumlah anggota IFS yang lebih kecil. Hal tersebut mirip dengan pernyataan bahwa suatu citra fraktal memerlukan lebih dari satu IFS untuk memproduksinya supaya mirip dengan citra aslinya.

Definisi 1.3. Pemetaan Kontraksi Lokal (Soberano [17]) Diberikan (X, d) ruang metrik kompak dan $S \subset X$ dengan $S \neq \emptyset$. Pemetaan $w : S \rightarrow X$ dikatakan kontraksi lokal pada (X, d) jika terdapat bilangan real s dengan $0 \leq s < 1$ sehingga $d(w(x), w(y)) \leq (s)(d(x, y))$ untuk setiap $x, y \in S$.

Definisi 1.4. IFS Lokal (Soberano [17]) Pemetaan $w_i : S_i \rightarrow X$ pemetaan kontraksi lokal pada ruang metrik (X, d) dengan faktor kontraksi s_i untuk $i = 1, 2, \dots, N$ dan $\bigcup_{i=1}^N S_i = S$. Selanjutnya $\{S : w_i : S_i \rightarrow X, i = 1, 2, \dots, N\}$ disebut IFS lokal.

Bilangan $s = \max\{s_i : i = 1, 2, \dots, N\}$ disebut faktor kontraksi untuk IFS lokal. IFS lokal juga bisa dinyatakan sebagai berikut. Jika G menyatakan himpunan semua subset dari X , maka bisa didefinisikan suatu operator $w_{\text{lokal}} : G \rightarrow G$ dengan $w_{\text{lokal}}(B) = \bigcup_{i=1}^N w_i(S_i \cap B)$ untuk semua $B \in G$

Dengan batasan tersebut, w_{lokal} dapat didefinisikan sebagai pemetaan kontraksi pada himpunan

bagian dari ruang Hausdorff. Jika A adalah himpunan bagian tak kosong dari X , himpunan A disebut atraktor dari IFS lokal jika $w_{\text{lokal}}(A) = A$. Jika himpunan A dan B adalah atraktor maka $A \cup B$ juga atraktor. Jika terdapat suatu atraktor, terdapat atraktor terbesar yang memuat semua atraktor yang lain tersebut. Atraktor terbesar ini berkenaan dengan atraktor dari w_{lokal} dan ditemukan dengan mengambil gabungan dari semua atraktor lain dalam w_{lokal} .

Diberikan IFS Lokal $\{S; w_i : S_i \rightarrow X, i = 1, 2, \dots, N\}$ dan misalkan himpunan S_i kompak maka dapat didefinisikan suatu barisan himpunan kompak bagian dari X dengan:

$$\begin{aligned} A_0 &= X \\ A_n &= \bigcup_{i=1}^N w_i(S_i \cap A_{n-1}) \end{aligned}$$

untuk $n = 1, 2, \dots, N$

Karena IFS memuat pemetaan-pemetaan kontraksi maka

$$A_0 \subset A_1 \subset A_2 \subset A_3 \subset \dots$$

sehingga $\{A_n\}$ merupakan barisan himpunan kompak yang menurun. Akibatnya ada suatu himpunan kompak $A \subset X$ dengan sifat

$$\lim_{n \rightarrow \infty} A_n = A$$

dan

$$A = \bigcup_{i=1}^N w_i(S_i \cap A) = w_{\text{lokal}}(A)$$

Telah disyaratkan sebelumnya, jika A tak kosong maka merupakan atraktor maksimal untuk IFS lokal. Jika ada himpunan kompak B sehingga $w_{\text{lokal}}(B) \subset B$ maka kemungkinan bahwa A adalah kosong. Himpunan A berkaitan dengan atraktor suatu IFS pada pola penentuan citra fraktal. Koefisien dari pemetaan w_i cukup penting dalam penetapan citra fraktal tersebut.

3. Invers Fraktal

Konstruksi fraktal dimulai dengan dimilikinya suatu IFS yang diiterasikan akan menghasilkan citra fraktal. Bagaimana dengan proses sebaliknya? Jika dimiliki suatu fraktal, bisakah selalu ditemukan suatu IFSnya? Barnsley memberikan jawaban untuk masalah tersebut melalui teorema collage.

Teorema 1.1. Teorema Collage (Barnsley [1]) Diberikan (X, d) ruang metrik lengkap dan $T \in H(X)$. Untuk sebarang bilangan $\varepsilon > 0$, diambil suatu IFS $\{X; (w_0), w_1, w_2, \dots, w_N\}$ dengan faktor kontraksi $0 \leq s < 1$ sehingga

$$h\left(T, \bigcup_{n=1}^N w_n(T)\right) \leq \varepsilon$$

dengan $h(d)$ adalah metrik Hausdorff, maka

$$h(T, A) \leq \frac{\varepsilon}{1-s}$$

dengan A adalah atraktor dari IFS, ekuivalen dengan mengatakan $h(T, A) \leq (1-s)^{-1} h\left(T, \bigcup_{n=1}^N w_n(T)\right)$ untuk semua $T \in H(X)$

Teorema Collage menjamin adanya IFS yang hasil iterasinya merupakan suatu fraktal yang mirip dengan sebarang himpunan bagian kompak yang diberikan. Secara singkat teorema collage bisa dikatakan sebagai berikut. Jika diberikan sebarang $T \in H(X)$ maka bisa dicari suatu IFS, dan jika IFS tersebut diiterasikan sebanyak N kali, maka akan ditemukan suatu himpunan kompak lain yaitu $\bigcup_{n=1}^N w_n(T)$ dengan sifat bahwa $T \in H(X)$ dan $\bigcup_{n=1}^N w_n(T)$ sangat mirip. Sedangkan himpunan A yang memenuhi $A = W(A) = \bigcup_{n=1}^N w_n(A)$ merupakan atraktor dari IFS yaitu $A = \lim_{n \rightarrow \infty} W^{on}(B)$ untuk setiap $B \in H(X)$. Ada hal yang berbeda mengenai arti invers dari fraktal dengan pengertian invers secara umum. Jika invers secara umum menghasilkan sesuatu yang adalah sama sebelum ketika sesuatu tersebut dikenakan fungsi tetapi tidak demikian dengan invers fraktal. Invers fraktal menghasilkan suatu himpunan kompak yang mirip dengan himpunan awal sebelum dicari IFS nya.

Salah satu aplikasi dari invers fraktal yang dikonstruksi melalui teorema collage adalah citra fraktal yang mirip dengan sebarang citra asli yang diberikan. Pandang himpunan R^2 . Diambil sebarang $S \subset R^2$ dengan S himpunan bagian kompak. Dipihak lain, jika diambil sebarang citra nyata berukuran $n \times m$ dengan $n, m \in R$ maka citra tersebut merupakan himpunan kompak. Dengan kata lain jika diambil sebarang $S \subset R^2$ dengan S himpunan kompak maka himpunan S bisa dipandang sebagai citra nyata.

Dengan demikian Teorema Collage di R^2 bisa dipandang sebagai berikut. Diberikan ruang metrik lengkap (R^2, d) dengan d adalah metrik biasa, dan sebarang citra nyata $T \in H(R^2)$. Untuk sebarang bilangan $\varepsilon > 0$, bisa ditemukan suatu IFS $\{X; (w_0), w_1, w_2, \dots, w_n\}$ dengan faktor kontraksi $0 \leq s < 1$ sehingga citra hasil iterasi dari IFS menjadi mirip dengan citra nyata T .

Pada perkembangannya IFS masih bisa bekerja jika transformasi affine kontraksi dibatasi hanya bekerja pada bagian-bagian dari keseluruhan citra. Ide pengganti untuk menemukan transformasi affin yang memetakan keseluruhan citra menjadi bagian-bagian dari citra itu sendiri adalah mencari suatu transformasi affine yang memetakan bagian citra ke bagian citra yang lain.

Metode yang dipakai untuk mengkonstruksi citra fraktal adalah metode pencocokan bagian-bagian citra (*matching*) antara bagian citra asli dengan citra yang nantinya akan dikonstruksi menjadi citra fraktal yang mirip dengan citra asli. Awalnya citra asli dan citra awal dibagi menjadi potongan-potongan citra. Selanjutnya potongan-potongan tersebut dilakukan pencocokan dengan menggunakan transformasi affine. Akibatnya transformasi affine yang diperlukan lebih dari satu dan menjadi himpunan transformasi affine. Pencocokan yang sedemikian rupa, yang terjadi antar potongan-potongan citra menjadikan himpunan transformasi affine tersebut tidak lagi IFS tetapi menjadi Partitioned Iterated Function System atau PIFS.

4. Cara Penelitian

Dalam penelitian ini digunakan tiga bahan utama untuk penelitian yaitu buku referensi dan jurnal, citra, dan program komputasi. Buku dan jurnal dipakai untuk memberikan dasar dan kajian teoritis serta contoh-contoh untuk membuat desain program. Citra yang dipilih adalah citra skala abu-abu dan bahasa pemrograman yang dipilih adalah MATLAB. Program yang dibangun terdiri dari dua bagian, bagian pertama program berfungsi untuk *encoding* atau pengkodean citra sedangkan bagian kedua berfungsi untuk *decoding* atau pembuatan citra fraktal.

Percobaan-percobaan meliputi beberapa tahapan. Tahapan pertama merupakan percobaan untuk mengetahui pengaruh transformasi yang dikenakan pada citra, meliputi dilatasi, rotasi, pen-

skalaan, dan transformasi miring. Tahapan kedua merupakan percobaan untuk menyusun citra fraktal berdasarkan citra Lena. Percobaan dilakukan dengan partisi ukuran pixel yang berbeda-beda dan selanjutnya hasil yang diperoleh akan dibandingkan.

5. Hasil dan Pembahasan

Ada dua proses inti yang dilakukan untuk menyelesaikan permasalahan yaitu proses menemukan IFS yang disebut encoding dan proses mengiterasikan IFS untuk membentuk fraktal yang disebut decoding. Untuk menyelesaikan proses tersebut secara manual tidaklah mungkin, karena selain melibatkan sejumlah besar potongan citra juga ukuran citra yang sangat kecil sehingga dengan keterbatasan tersebut maka dipilih cara komputasi untuk menyelesaikannya dan dalam hal ini dipilih Matlab untuk menyelesaikannya.

Menggunakan Matlab untuk menyelesaikan berarti harus memandang setiap citra selain sebagai kumpulan pixel juga citra dipandang sebagai matriks. Matlab bisa mengidentifikasi setiap citra sebagai matriks terbatas pada ekstensi tertentu yaitu setiap citra dengan ekstensi berikut.

Format	Deskripsi	Ekstensi
TIFF	Tagged Image File Format	.tiff , .tif
JPEG	Joint Photographic Expert Group	.jpeg , .jpg
GIF	Graphics Interchange Format	.giv
BMP	Windows Bitmap	.bmp
PNG	Portable Network Graphics	.png
XWD	X Window Dump	.xwd

Suatu citra digital yang dinyatakan oleh Matlab mempunyai representasi matriks sebagai berikut:

$$\begin{pmatrix} f(1,1) & f(1,2) & \cdots & f(1,N) \\ f(2,1) & f(2,2) & \cdots & f(2,n) \\ \cdots & \cdots & \cdots & \cdots \\ f(N,1) & f(N,2) & \cdots & f(N,N) \end{pmatrix}$$

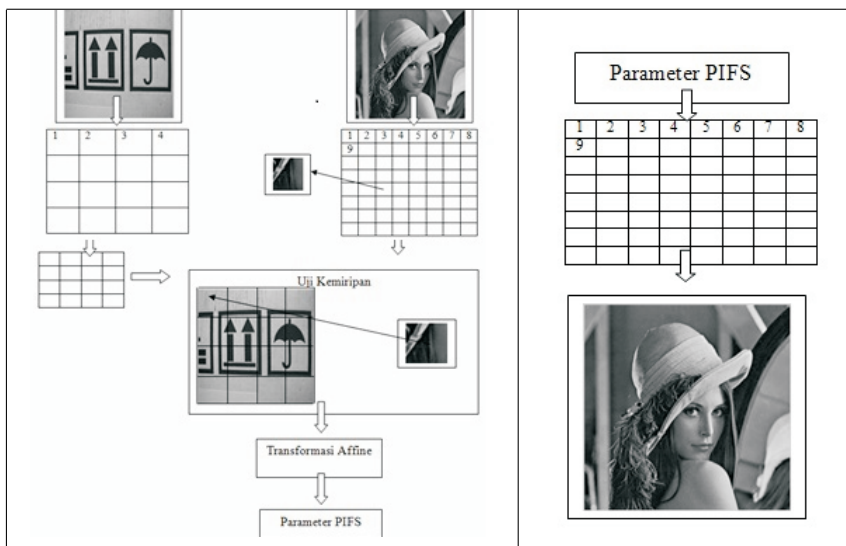
Setiap elemen pada matriks tersebut disebut elemen citra atau pixel. Elemen $f(i, j)$ menyatakan pewarnaan (tingkat kecerahan) pada elemen baris i kolom j . Demikian seterusnya. Pada citra skala abu-abu, besarnya elemen pada baris ke i kolom ke j tergantung dari format citra yang bersangkutan. Meskipun citra yang dinyatakan sebagai matriks posisinya koordinatnya dinyatakan oleh bilangan bulat tetapi nilai setiap koordinat tidak terbatas pada bilangan bulat tersebut. Terdapat beberapa kelas data pada Matlab yang bisa menyatakan suatu citra. Misalkan kelas uint8 dengan range $[0, 255]$, kelas uint16 dengan range $[0, 65535]$, kelas uint32 dengan range $[0, 4294967295]$ dst.

5.1. Proses Encoding

Proses *encoding* dilakukan dengan cara melakukan partisi pada citra asli menjadi blok-blok berbentuk persegi berukuran tertentu, blok-blok tersebut disebut blok range. Ada beberapa metode untuk melakukan partisi citra seperti metode *fixed square* dan metode *quadtree*. Metode *quadtree* bekerja dengan cara membagi suatu citra menjadi empat subcitra. Untuk subcitra yang telah memenuhi suatu tingkat kehomogenan tertentu maka subcitra tersebut tidak akan dibagi lagi. Sebaliknya jika subcitra tidak memenuhi suatu kriteria kehomogenan tertentu maka subcitra tersebut akan dibagi lagi menjadi empat dan setiap subcitra tersebut akan diuji tingkat kehomogenannya, jika telah memenuhi kriteria maka subcitra tidak akan dibagi lagi tetapi jika subcitra tersebut belum memenuhi kriteria tertentu maka subcitra akan dibagi empat lagi dan demikian seterusnya hingga memungkinkan suatu subcitra berukuran paling kecil dan pasti memenuhi suatu kriteria kehomogenan tertentu. Sedangkan metode *fixed square* diawali dengan mengambil dua sebarang citra abu-abu berukuran sama, satu sebagai citra yang akan ditemukan citra fraktalnya dan yang satu adalah citra awal. Citra awal berguna sebagai citra yang akan dikonstruksi menjadi citra fraktal yang mirip dengan sebarang citra asli. Citra asli merupakan citra yang akan ditemukan padanannya melalui proses *encoding*. Bagi citra awal menjadi subcitra-subcitra berukuran secara tumpang tindih (*overlapping*), artinya untuk setiap $D_i, D_j \in D \subset R^2$, $D_i \cap D_j \neq \emptyset$, untuk $i \neq j$. Subcitra tersebut disebut dengan sel-sel domain. Selanjutnya citra asli dipartisi menjadi subcitra-subcitra yang tidak saling tumpang tindih dengan ukuran $n \times n$ pixel, artinya untuk setiap $R_k, R_l \in R^2$, $R_k \cap R_l = \emptyset$ untuk $k \neq l$. Subcitra-subcitra tersebut selanjutnya disebut sel-sel range. Proses utama dari encoding adalah menemukan sel domain yang mirip dengan sel range untuk semua sel range pada setiap sel domain. Pencocokan sel range dengan sel domain hanya mungkin jika ukuran tiap sel range dan sel domain sama. Untuk memenuhi syarat tersebut maka ukuran sel-sel domain diperbaharui dari berukuran $2n \times 2n$ menjadi sama dengan ukuran sel range yaitu $n \times n$. Proses ini dilakukan dengan cara mengambil rata-rata dari setiap bagian sel domain berukuran 2×2 pixel diartikan sebagai 1 pixel.

Untuk setiap sel range akan dicari sel domain yang paling mirip. Proses pencarian sel domain tersebut menggunakan transformasi affine. Diambil sebarang sel range $R_{k,l}$ yaitu sel range yang terletak pada baris ke k dan kolom ke l . Pencarian sel domain bisa dilakukan dengan aturan tertentu atau bisa dilakukan urut berdasarkan baris dan kolom. Untuk setiap sel domain $D_{i,j}$ uji kemiripannya dengan $R_{k,l}$. Uji kemiripan pertama kali dilakukan secara langsung atau menggunakan transformasi identitas, selanjutnya hasil tersebut dibandingkan kemiripannya setelah $D_{i,j}$ dikenakan transformasi antara lain rotasi 90° , rotasi 180° , rotasi 270° , putar poros horisontal dan putar poros vertikal. Hasil setiap perbandingan dicatat dan pilih $D_{i,j}$ yang paling mirip $R_{k,l}$. Proses dilanjutkan untuk sel domain yang lain dan untuk sel range yang lain.

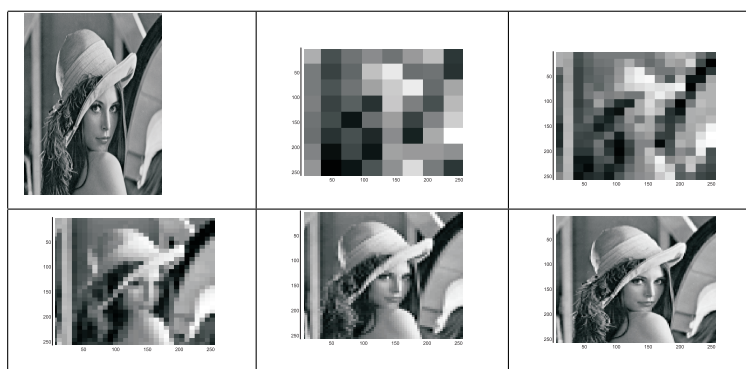
Proses pencarian dan pencocokan bagian-bagian citra dari sel range dan sel domain merupakan bagian yang sangat rumit terutama dari waktu yang dibutuhkan. Pengembangan suatu algoritma untuk proses *encoding* bisa dilakukan untuk meminimalkan waktu tersebut. Sifat kesebangunan diri lokal (*local self-similarity*) bisa digunakan untuk mengembangkan efisiensi mekanisme pencarian domain.



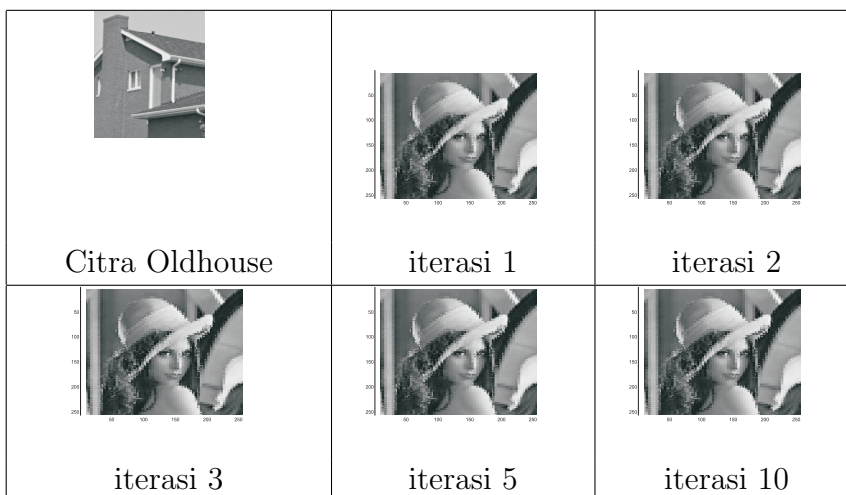
Gambar 1 Skema Encoding dan Decoding

5.2. Proses Decoding

PIFS yang diperoleh dari proses encoding merupakan dasar dari proses decoding. Untuk memulai proses decoding, informasi yang tersimpan dalam PIFS diambil dan selanjutnya siap untuk diiterasikan.



Gambar 2 Citra Fraktal Lena



Gambar 2 Citra Fraktal Lena Ukuran Blok Range 4×4 Pixel

Gambar 3 merupakan citra fraktal Lena dengan citra awal Oldhouse. Dari hasil tersebut terlihat bahwa untuk ukuran partisi dengan ukuran blok range semakin kecil maka hasil yang diperoleh lebih halus dan lebih mencerminkan citra aslinya. Meskipun demikian, waktu yang diperlukan untuk memperoleh citra fraktal dengan partisi pixel yang mengecil memerlukan waktu lebih lama.

6. Kesimpulan dan Saran

Hasil dari inves fraktal pada citra tersebut merupakan IFS yang disusun dari 8 transformasi dengan menyertakan alamat pasangan blok range dengan blok domain beserta ukuran mengkerutnya. Berdasarkan proses dan hasil yang diperoleh, ada beberapa hal yang menjadi kelebihan dan permasalahan yang muncul pada saat melakukan proses iterasi untuk mendapatkan citra fraktal dari sebarang citra awal sehingga kedua citra tersebut mirip. Masalah pokok yang muncul adalah rumitnya proses *encoding* dengan tingkat kompleksitas komputasi yang cukup tinggi sehingga waktu yang diperlukan untuk melakukan proses encoding sangat lama, tidak sebanding dengan waktu yang diperlukan untuk proses *decoding*. Hal tersebut dapat dilakukan untuk penelitian lebih lanjut untuk lebih memperpendek waktu *encoding* misalkan dilakukan pemrograman paralel. Permasalahan yang kedua adalah citra yang digunakan masih merupakan citra dalam skala abu-abu, hal tersebut dapat diperluas untuk sebarang citra berwarna serta dari segi ukuran mungkin bisa dihilangkan syarat ukuran citra yang masih mengharuskan berbentuk persegi.

Proses pencocokan blok range dan blok domain pada bahasan diatas dikerjakan dengan bahasa Matlab. Selanjutnya bisa dikembangkan untuk menggunakan bahasa pemrograman lain seperti bahasa C, Pascal dan lainnya yang memiliki *toolbox-toolbox* yang berbeda sehingga bisa diharapkan memberikan hasil yang lebih baik. Pengembangan tetap bisa dilakukan pada penyelesaian masalah waktu *encoding* yang lebih singkat. Hal lain yang masih bisa dikerjakan adalah pembuatan blok range dan blok domain yang mungkin bisa tidak berbentuk persegi.

7. Lampiran

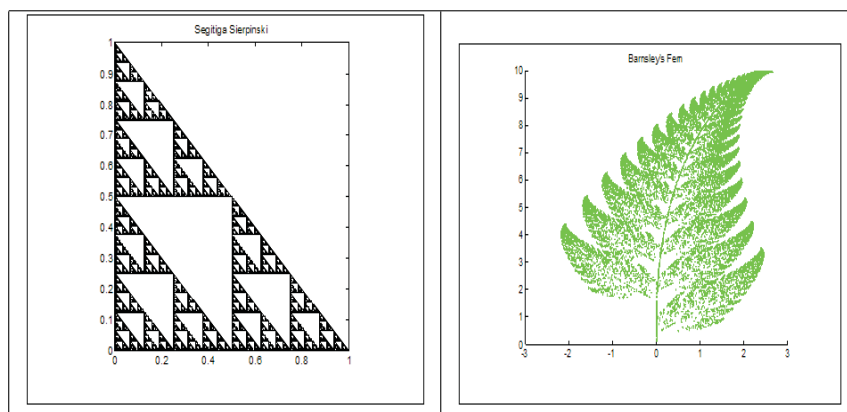
IFS dan Hasil Iterasinya

w	a	b	c	d	e	f	g
1	0,5	0	0	0,5	1	1	0,33
2	0,5	0	0	0,5	1	50	0,33
3	0,5	0	0	0,5	50	50	0,34

Tabel 1 IFS dari Segitiga Sierpinski

w	a	b	c	d	e	f	g
1	0	0	0	0,16	0	0	0,01
2	0,85	0,04	-0,04	0,85	0	1,6	0,85
3	0,2	-0,26	0,23	0,22	0	1,6	0,07
4	-0,15	0,28	0,26	0,24	0	0,44	0,07

Tabel 2 IFS dari Pakis Barnsley



Gambar 4 Hasil Iterasi dari IFS

References

- [1] Barnsley, M. F., 1988, *Fractals Everywhere*, Academic Press, Inc., London
- [2] Becker, Karl-Heinze, 1989, *Dynamical Systems and Fractals : Computer Graphics Experiment in Pascal*, Cambridge University Press, Cambridge.
- [3] Devaney, R., L., 1992, *A First Course In Chaotic Dynamical Systems, Theory And Experiment*, Addison Wesley Publishing Company, Inc., Boston
- [4] Edgar, Gerald, 2007, *Measure, Topology, and Fractal Geometry*, Second Edition, Springer, USA
- [5] Falconer, Kenneth, 2003, *Fractal Geometri, Mathematical Foundation And Application: Second Edition*, John Wiley And Sons, Ltd, England
- [6] Falconer, Kenneth, 1997, *Techniques in Fractal Geometri*, John Wiley And Sons, England
- [7] Fisher, Y, 1995, *Fractal Image Compression : Theory And Appliation* , Springer-Verlag New York, Inc, New York
- [8] Gonzales R.C., 2005, *Digital Image Processing Using Matlab*, Prentice Hall, Upper Saddle River, New Jersey
- [9] Helmbert, Gilbert, 2007, *Getting Acquainted with Fractals*, Walter de Gruyter, GmbH and Co, Berlin, Germany
- [10] Hoggar, S.,G., 2006, *Mathematics of Digital Images: Creation, Compression, Restoration, Recognition*, Cambridge University Press, New York
- [11] Liebovitch, L.,S., 1998, *Fractals and Chaos : Simplified for The Life Sciences*, Oxford University Press, Oxford
- [12] Lowen, S.,G., Teich,M.,C., 2005, *Fractal Base Point Processes*, John Wiley and Sons, Inc, Hoboken, New Jersey
- [13] Mandelbrot, Benoit B., 1982, *The Fractal Geometry of Nature*, WH Freeman and Company, New York
- [14] Nikiel, Slawomir, 2007, *Iterated Function Systems for Real-Time Image Synthesis*, Springer Verlag London Limited, London
- [15] Pietgen,H.O., 2004, Jurgens,H., and Saupe, D., *Chaos and Fractals : New Frontiers of Science*, Second Edition, Springer-Verlag New York, Inc., New York
- [16] Pitas, Ioannis, 1993, *Digital Image Processing Algorithms*, Prentice Hall International (UK) Ltd
- [17] Soberano, Lisa, A., 2000, *The Mathematical Foundation of Image Compression*, University of North Carolina at Wilmington, North Carolina
- [18] Steven, Roger T., 1990, *Advanced Fractal Programming In C*, M &T Publishing, Inc., Redwood City
- [19] Utomo,B,. 2010, *Implementasi Teorema Collage pada Citra*, Tesis, Jurusan Matematika, Universitas Gadjah Mada, Jogjakarta
- [20] Widodo, 2010, *Entropi Sistem Dinamika Diskrit dan Penerapannya Pada Entropi Barisan DNA*, Pidato Pengukuhan Guru Besar dalam Bidang Ilmu Matematika, Fakultas MIPA, Universitas Gadjah Mada, Jogjakarta