



**PEMANFAATAN *ITERATED FUNCTION SYSTEM* (IFS) UNTUK
MEMBANGKITKAN MOTIF ANYAMAN
UKURAN $n \times n$**

SKRIPSI

Oleh

**Ingka Maris
NIM 151810101064**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2019**



**PEMANFAATAN *ITERATED FUNCTION SYSTEM* (IFS) UNTUK
MEMBANGKITKAN MOTIF ANYAMAN
UKURAN $n \times n$**

SKRIPSI

diajukan guna melengkapi tugas akhir dan memenuhi salah satu syarat
untuk menyelesaikan Program Studi Matematika (S1)
dan mencapai gelar Sarjana Sains

Oleh

**Ingka Maris
NIM 151810101064**

**JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2019**

PERSEMBAHAN

Puji syukur dengan menyebut nama Allah SWT yang Maha Pengasih dan Maha Penyayang serta Sholawat atas Nabi Muhammad SAW sehingga terbentuklah skripsi ini dan saya persembahkan untuk:

1. kedua orang tua saya yang tercinta yaitu Bapak Slamet, S.Pd., M.Pd dan Ibu Tutik Nurhayati, kedua kakak saya Arya dan Ratna serta seluruh keluarga yang telah memberikan do'a, semangat dan perhatian terhadap saya;
2. seluruh dosen dan guru yang telah membimbing dan memberikan ilmunya kepada saya sejak sekolah dasar hingga perguruan tinggi dengan penuh kesabaran dan kasih sayang;
3. Almamater Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember, SMA Negeri 1 Purwoharjo, SMP Negeri 1 Muncar dan SD Negeri 7 Sumberberas;
4. teman-teman SIGMA 15 HIMATIKA Geokompstat dan UKMS TITIK yang selalu memberikan dukungan, bantuan dan do'a;
5. teman-teman kos Melati yaitu Yulia dan Resofi yang telah memberikan dukungan;
6. teman-teman pejuang fraktal yaitu Intan, Ipin, Iza, Mela, Mitha, Nadiya, Novita dan Rozida yang selalu memberikan semangat dan bantuan;
7. ojek online Ilham yang selalu memberikan perhatian, dukungan dan semangat.

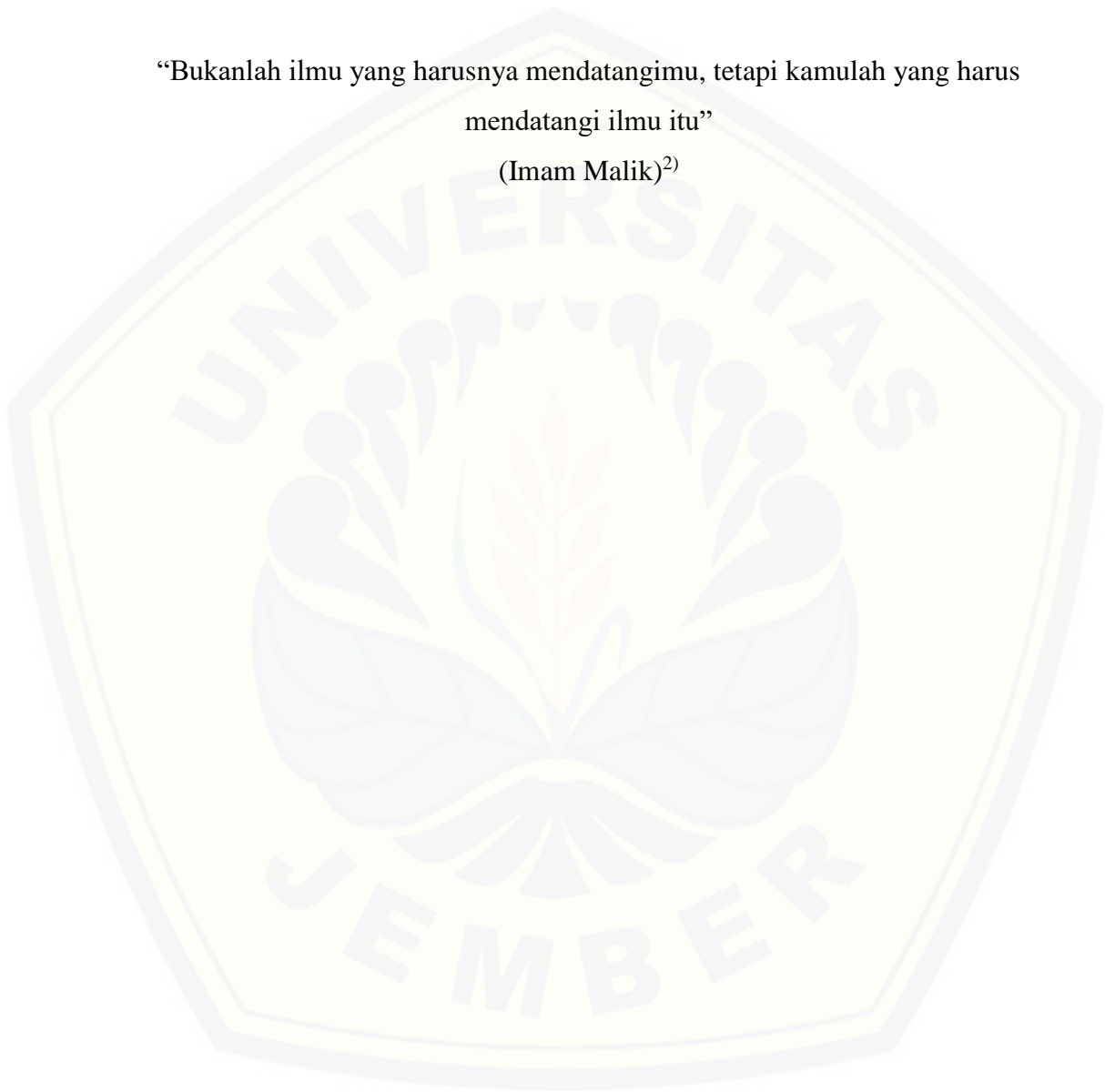
MOTO

“Sesungguhnya sesudah kesulitan itu ada kemudahan”

(QS. Asy Syarh: 5-6)¹⁾

“Bukanlah ilmu yang harusnya mendatangimu, tetapi kamulah yang harus
mendatangi ilmu itu”

(Imam Malik)²⁾



¹⁾Departemen Agama Republik Indonesia. 2005. *Al-Quran dan Terjemahannya*.
Surabaya. CV. Karya Utama

²⁾<https://www.kutipkata.com/kumpulan-motto-hidup-islami/>

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Ingka Maris

NIM : 151810101064

menyatakan dengan sesungguhnya bahwa skripsi yang berjudul “Pemanfaatan *Iterated Function System* (IFS) untuk Membangkitkan Motif Anyaman Ukuran $n \times n$ ” adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggungjawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenar-benarnya tanpa ada tekanan dan paksaan dari pihak manapun dan bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Januari 2019

Yang menyatakan,

Ingka Maris
NIM 151810101064

SKRIPSI

**PEMANFAATAN *ITERATED FUNCTION SYSTEM* (IFS) UNTUK
MEMBANGKITKAN MOTIF ANYAMAN UKURAN $n \times n$**

Oleh

Ingka Maris
NIM 151810101064

Pembimbing

Dosen Pembimbing Utama : Kosala Dwidja Purnomo, S.Si., M.Si

Dosen Pembimbing Anggota : Bagus Juliyanto, S.Si., M.Si

PENGESAHAN

Skripsi berjudul “Pemanfaatan *Iterated Function System* (IFS) untuk Membangkitkan Motif Anyaman Ukuran $n \times n$ ” karya Ingka Maris telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas
Jember

Tim Penguji:

Ketua,

Anggota I,

Kosala Dwidja Purnomo, S.Si., M.Si.
NIP. 196908281998021001

Bagus Juliyanto, S.Si., M.Si.
NIP. 198007022003121001

Anggota II,

Anggota III,

Dr. Firdaus Ubaidillah, S.Si., M.Si.
NIP. 197006061998031003

Dr. Kristiana Wijaya, S.Si., M.Si.
NIP. 197408132000032004

Mengesahkan
Dekan,

Drs. Sujito, Ph.D.
NIP. 196102041987111001

RINGKASAN

Pemanfaatan *Iterated Function System (IFS)* untuk Membangkitkan Motif Anyaman Ukuran $n \times n$; Ingka Maris, 151810101064; 2019; 42 halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Kesenian merupakan salah satu hal yang sangat dekat dengan kehidupan misalnya kerajinan anyaman. Kerajinan anyaman dibuat dengan tindih-menindih atau silang-menyilang dan memiliki berbagai macam motif. Motif anyaman terbuat dari susunan bangun dua dimensi yaitu persegi.

Anyaman merupakan salah satu bentuk dari penerapan bidang matematika yaitu fraktal. Ciri khas fraktal berupa *Self-similarity* yaitu keadaan objek yang dibangun secara berulang dengan mengganti suatu gambar dengan yang sebangun, tetapi berukuran lebih kecil dari asalnya. Salah satu cara yang dapat digunakan untuk mengkonstruksi fraktal adalah menggunakan *Iterated Function System (IFS)*.

Motif anyaman dapat dibuat dengan membentuk sebuah pola dasar terlebih dahulu. Pola dasar yang dibentuk menggunakan ukuran grid $n \times n$ pada tabel GUI dengan $n > 1$ untuk menghasilkan sebuah motif. Setelah mendapatkan ukuran pola dasar, langkah selanjutnya yaitu memberi pola dengan menginputkan bilangan bulat. Agar mendapatkan motif anyaman yang menarik, diberikan beberapa iterasi pada pola dasar.

Semakin sedikit ukuran $n \times n$ pada pola yang digunakan, maka variasi pola dasar akan semakin sedikit. Sedangkan apabila menggunakan ukuran $n \times n$ semakin banyak, maka variasi pola dasar akan semakin banyak. Setelah menentukan ukuran grid, maka langkah selanjutnya yaitu memasukkan pola pada tabel. Pola dibuat dengan memasukkan bilangan bulat. Bilangan bulat terbesar merupakan tempat pengoperasian IFS.

Motif anyaman yang dibuat menggunakan IFS harus disesuaikan dengan pola dasarnya. Apabila menggunakan pola dasar dengan ukuran grid $n \times n$

semakin sedikit, maka harus diberikan iterasi semakin banyak. Sedangkan apabila menggunakan pola dasar dengan ukuran grid $n \times n$ semakin banyak, maka harus diberikan iterasi semakin sedikit. Jika menggunakan iterasi terlalu banyak, maka gambar motif tidak akan terlalu jelas karena *pixel* yang dihasilkan terlalu banyak sehingga menyebabkan motif anyaman menjadi rumit.



PRAKATA

Puji syukur dengan menyebut nama Allah SWT atas segala rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Pemanfaatan *Iterated Function System* (IFS) untuk Membangkitkan Motif Anyaman Ukuran $n \times n$ ”. Skripsi ini disusun untuk memenuhi salah satu syarat menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Penyusun skripsi ini tidak lepas dari bantuan berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Kosala Dwidja Purnomo, S.Si., M.Si. selaku Dosen Pembimbing Utama dan Bagus Juliyanto, S.Si., M.Si. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
2. Dr. Firdaus Ubaidillah, S.Si., M.Si. dan Dr. Kristiana Wijaya, S.Si., M.Si. selaku Dosen Penguji yang telah memberikan kritik dan saran yang membangun demi kesempurnaan skripsi ini;
3. Drs. Rusli Hidayat, M.Sc. selaku Dosen Pembimbing Akademik yang selalu memberikan motivasi dan pengarahan selama penulis menjadi mahasiswa;
4. seluruh Dosen dan Karyawan Jurusan Matematika Fakultas MIPA Universitas Jember;
5. semua pihak yang tidak dapat disebutkan satu per satu.

Penulis juga menerima segala kritik dan saran dari semua pihak demi kesempurnaan skripsi ini. Akhirnya penulis berharap, semoga skripsi ini dapat bermanfaat.

Jember, Januari 2019

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTO	iii
HALAMAN PERNYATAAN	iv
HALAMAN PEMBIMBINGAN	v
HALAMAN PENGESAHAN	vi
RINGKASAN	vii
PRAKATA	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN	xiv
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan.....	4
1.4 Manfaat	4
BAB 2. TINJAUAN PUSTAKA	5
2.1 Penelitian Terdahulu	5
2.2 Anyaman.....	6
2.3 Fraktal	8
2.4 Karpets Sierpinski	8
2.5 Koordinat dalam Grid	9
2.6 GUI pada Matlab.....	10
2.7 Iterated Function System (IFS)	11
BAB 3. METODE PENELITIAN	13
3.1 Pembentukan Pola Anyaman pada GUI	13

3.2 Motif Anyaman Dibangkitkan pada GUI Menggunakan Metode <i>Iterated Function System (IFS)</i>	14
3.3 Simulasi Program	15
BAB 4. HASIL DAN PEMBAHASAN	17
4.1 Hasil Penelitian	17
4.1.1 Pola Dasar Motif Anyaman	17
4.1.2 Pembangkitan Motif Anyaman pada GUI Menggunakan IFS ...	21
4.2 Pembahasan	26
4.2.1 Pola Dasar Anyaman	26
4.2.2 Pembangkitan Motif Anyaman Menggunakan IFS	28
BAB 5. KESIMPULAN DAN SARAN	31
5.1 Kesimpulan.....	31
5.2 Saran	31
DAFTAR PUSTAKA	33
LAMPIRAN	35

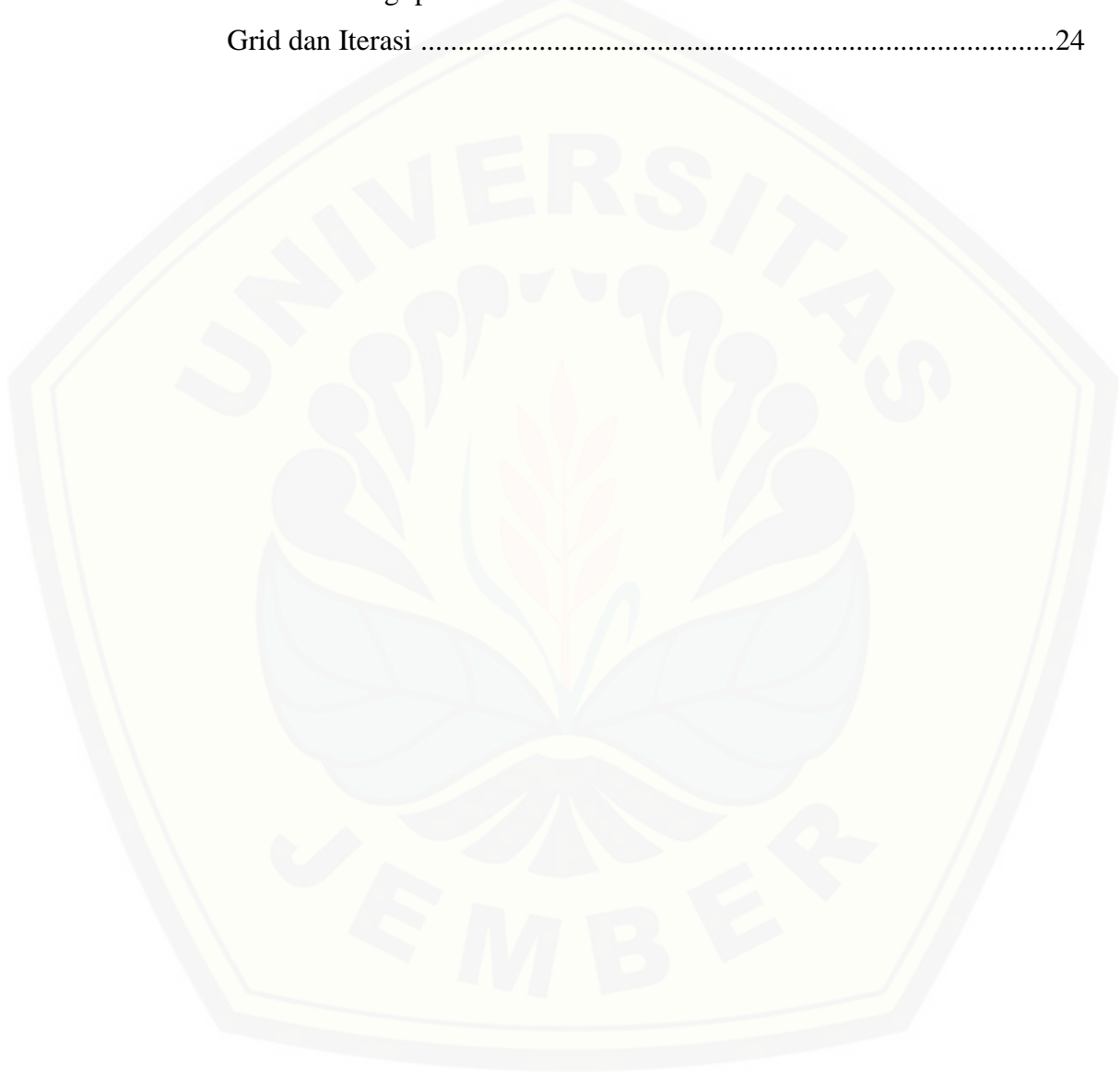
DAFTAR GAMBAR

Gambar 1.1 Macam-Macam Jenis Fraktal	2
Gambar 2.1 Motif Anyaman dari Bujursangkar dan Persegi Panjang dengan Dilasi dan Translasi dalam IFS	5
Gambar 2.2 Contoh Variasi Motif Anyaman	6
Gambar 2.3 Macam-Macam Anyaman	7
Gambar 2.4 Karpets Sierpinski	9
Gambar 2.5 Pembagian Bidang	9
Gambar 3.1 Diagram Alir Skema Penelitian	13
Gambar 3.2 Variasi Motif Anyaman	15
Gambar 4.1 Pola Anyaman Ukuran Grid 1×1	17
Gambar 4.2 Pola Dasar Anyaman	18
Gambar 4.3 Pola Dasar Anyaman Ukuran Grid 7×7	19
Gambar 4.4 Pola Dasar Menggunakan Tanda Bilangan Bulat 2, 3, 4, 5, 6, 7, 8, 9 dan 10	20
Gambar 4.5 Pola Anyaman Menggunakan Dua Bilangan Bulat pada Satu Baris atau Satu Kolom	21
Gambar 4.6 Variasi Motif Anyaman	22
Gambar 4.7 Variasi Motif Anyaman	23
Gambar 4.8 Tombol <i>Color Map</i>	24
Gambar 4.9 Variasi Warna Motif Anyaman	25
Gambar 4.10 Pola Dasar Anyaman	26
Gambar 4.11 Variasi Pola Anyaman	27
Gambar 4.12 Pengembangan Pola Dasar Anyaman	27
Gambar 4.13 Variasi Motif Anyaman dengan Iterasi Maksimal	29

DAFTAR TABEL

Tabel 4.1 Jumlah Persegi pada Satu Baris atau Satu Kolom Berdasarkan Ukuran
Grid 3×3 dan Iterasi22

Tabel 4.2 Jumlah Persegi pada Satu Baris atau Satu Kolom Berdasarkan Ukuran
Grid dan Iterasi24



DAFTAR LAMPIRAN

Lampiran 4.1 Script Program Matlab	35
Lampiran 4.2 Motif Anyaman	41



BAB 1. PENDAHULUAN

1.1 Latar Belakang

Kesenian merupakan suatu hal yang sangat dekat dengan kehidupan. Pengertian seni menurut Aristoteles adalah bentuk yang pengungkapannya dan penampilannya tidak pernah menyimpang dari kenyataan dan seni itu adalah meniru alam. Terdapat macam-macam seni, salah satu diantaranya adalah seni kerajinan. Kerajinan merupakan sebuah benda yang dibuat dengan tangan dan menggunakan berbagai macam bahan dari alam. Salah satu kerajinan tangan tersebut adalah anyaman. Kerajinan anyaman merupakan kerajinan tangan yang masih sering dibuat dan dapat dijumpai sampai saat ini. Anyam adalah mengatur, tindih-menindih dan silang-menyilang, melakukan pekerjaan menganyam (Sugiono, 2008). Anyaman dapat dibuat menggunakan berbagai macam bahan sehingga mengakibatkan terbentuknya berbagai motif anyaman.

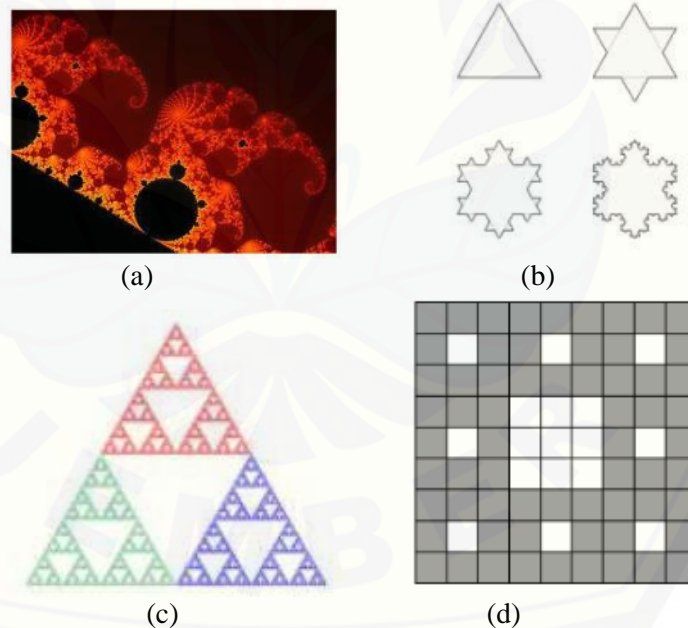
Kerajinan anyaman memiliki berbagai macam variasi motif. Hal ini bertujuan agar tidak terlihat monoton dan menambah unsur keindahan pada anyaman tersebut. Terdapat beberapa contoh motif anyaman yaitu motif Mata Itik, motif Mata Walik, motif Sasag dan motif Kepang. Terdapat pula motif yang terdiri atas bangun-bangun dua dimensi yaitu bangun persegi dan bangun persegi panjang. Bangun dua dimensi tersebut disusun untuk membuat sebuah motif yang teratur.

Bidang matematika dapat diaplikasikan pada teknologi komputer. Teknologi ini sudah berkembang pesat di Indonesia. Proses perhitungan matematika maupun perhitungan lainnya bisa didapat dengan mudah dan cepat apabila memanfaatkan komputer. Terdapat salah satu bidang matematika yang dapat diaplikasikan pada teknologi komputer yaitu fraktal. Fraktal berasal dari bahasa latin yaitu *frengere* merupakan kata kerja yang berarti membelah atau *fractal* merupakan kata sifat yang berarti tidak teratur atau terfragmentasi (Mandelbrot, 1983). Fraktal digunakan pertama kali untuk menunjukkan kurva-kurva yang memiliki sifat *self-similarity*. Ciri khas fraktal berupa *Self-similarity* adalah keadaan objek yang dibangun secara berulang dengan mengganti suatu gambar dengan yang sebangun, tetapi berukuran lebih kecil dari asalnya. Selain *self-similarity*, fraktal juga

memiliki ciri khas lain yaitu *infinite detail* merupakan objek fraktal yang memiliki bentuk dasar yang seakan-akan tidak habis-habis apabila diperhatikan (Santoso, 1994).

Fraktal didapat dengan cara mengulang suatu pola yang dimiliki sehingga menghasilkan struktur atau bentuk yang serupa dengan bentuk semula pada setiap bagiannya. Salah satu cara yang dapat digunakan untuk mengkonstruksi fraktal adalah menggunakan *Iterated Function System* (IFS). IFS adalah suatu metode yang terasosiasi dengan fungsi iterasi untuk membentuk suatu fraktal.

Fraktal memiliki berbagai macam jenis antara lain: *Koch Snowflake*, segitiga Sierpinski, karpet Sierpinski, kurva Hilbert, himpunan Cantor, himpunan Mandelbrot dan himpunan Julia (Mandelbrot, 1983). Macam-macam jenis fraktal tersebut memiliki bentuk yang berbeda-beda dan dapat dilihat pada Gambar 1.1.



Gambar 1.1 Macam-Macam Jenis Fraktal (a) Himpunan Mandelbrot (b) *Koch Snowflake* (c) Segitiga Sierpinski (d) Karpet Sierpinski (Sumber: Hasang, 2012)

Jenis-jenis fraktal dapat diaplikasikan terhadap benda-benda yang ada di dalam kehidupan, misalnya anyaman. Anyaman dapat dibuat menggunakan salah satu jenis fraktal yaitu karpet Sierpinski. Suria, *dkk* (2014) menyatakan bahwa

motif anyaman yang dihasilkannya menggunakan fraktal karpet Sierpinski dengan objek penyusunan berupa persegi yang disusun dengan menggunakan metode *Iterated Function System* (IFS). Penelitian Suria mengacu pada penelitian-penelitian sebelumnya dan menciptakan motif anyaman baru menggunakan metode yaitu IFS dengan fraktal karpet Sierpinski yang diaplikasikan menggunakan bahasa pemrograman C++. Aplikasi ini dapat digunakan untuk membuat motif-motif anyaman baru menggunakan karpet Sierpinski sesuai dengan pola yang ingin dibuat. Penelitian yang dilakukan oleh Suria menggunakan ukuran 5×5 grid. Selain itu, dalam penelitian Situngkir (2013) terdapat aplikasi yang dibangun dengan melakukan pengubahan suatu citra menjadi grid anyaman sehingga diperoleh suatu metode yang dapat digunakan untuk membuat pola anyaman. *Cellular-Automata* adalah metode yang digunakan dalam membuat inovasi tersebut. Adaptasi dari modul-modul *Cellular-Automata* diterapkan pada aplikasi komputasional untuk membentuk motif-motif baru pada anyaman di Indonesia dengan menggunakan teknologi modern.

Berdasarkan uraian yang telah disampaikan di atas, penulis ingin mengembangkan motif-motif anyaman dengan memanfaatkan IFS yang menyerupai karpet Sierpinski. Membentuk motif-motif anyaman tersebut dapat dibuat menggunakan aplikasi komputasi dengan membangun citra yang dibagi menjadi beberapa grid anyaman sehingga terbentuklah sebuah motif anyaman baru. Penulis ingin menggunakan salah satu aplikasi komputasi yaitu GUI pada Matlab.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, penelitian ini akan membahas bagaimana membuat motif anyaman ukuran grid $n \times n$ dengan memanfaatkan *Iterated Function System* (IFS) melalui GUI pada Matlab.

1.3 Tujuan

Berdasarkan rumusan masalah di atas, tujuan dari penulisan yaitu mendapatkan bermacam-macam motif anyaman ukuran grid $n \times n$ dengan memanfaatkan *Iterated Function System* (IFS) melalui GUI pada Matlab.

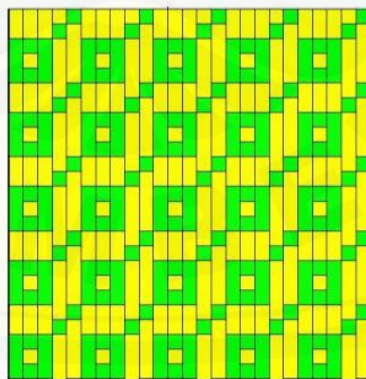
1.4 Manfaat

Manfaat dari penulisan ini adalah dapat membuat motif-motif anyaman ukuran grid $n \times n$ dengan memanfaatkan karpet Sierpinski dan menggunakan metode *Iterated Function System* (IFS) yang dibuat dengan aplikasi komputasi GUI pada Matlab. Selain itu, dengan menggunakan fraktal terutama karpet Sierpinski, pengrajin anyaman dapat lebih mudah untuk mengembangkan motif-motif baru pada anyaman.

BAB 2. TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

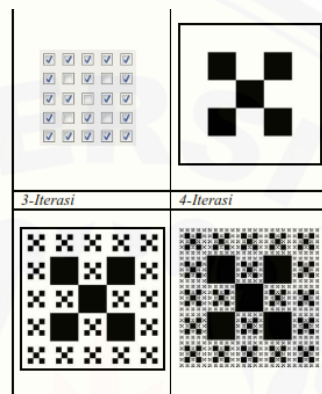
Penelitian yang dibuat oleh Isyana Prasasti pada awal tahun 2018 membahas tentang pengembangan motif anyaman dengan transformasi afin yaitu translasi, dilasi dan translasi serta rotasi dan translasi secara IFS. Motif anyaman yang ingin dibuat memanfaatkan objek geometri berupa persegi panjang atau bujursangkar. Setelah menentukan posisi awal objek, selanjutnya dilakukan transformasi afin dengan mengubah nilai arah pergeseran searah sumbu x maupun sumbu y atau kedua-duanya dan dilakukan translasi secara berulang-ulang. Agar didapatkan motif anyaman, maka dilakukan iterasi pada objek geometri yang telah diberikan transformasi afin. Namun, terdapat kelemahan dari program yang telah dibuat pada penelitian ini yaitu apabila program sudah dijalankan dan di tengah-tengah proses terdapat kesalahan dalam menginputkan parameter nilai, maka terjadi kesalahan pada gambar motif anyaman. Oleh karena itu langkah selanjutnya yang dilakukan yaitu gambar yang sudah terbentuk sebelumnya harus *direset* dan memulainya dengan memasukkan beberapa parameter nilai dari awal atau langkah pertama. Gambar 2.1 merupakan contoh motif anyaman yang telah dibuat pada penelitian Prasasti.



Gambar 2.1 Motif anyaman dari bujursangkar dan persegi panjang dengan dilasi dan translasi dalam IFS
(Sumber: Prasasti, 2018)

Suria, *dkk* juga melakukan penelitian mengenai pengembangan motif anyaman pada tahun 2014. Penelitiannya mengenai pemanfaatan fraktal karpet

Sierpinski untuk membuat variasi motif anyaman. Metode yang dibuat dalam membuat motif anyaman yaitu menggunakan IFS. Metode tersebut diterapkan pada pemrograman C++ sehingga menghasilkan variasi motif anyaman. Pola dasar yang dibuat pada penelitian ini menggunakan ukuran grid 5×5 dengan memanfaatkan *checkbox*. Contoh motif anyaman pada penelitian Suria dapat dilihat pada Gambar 2.2.

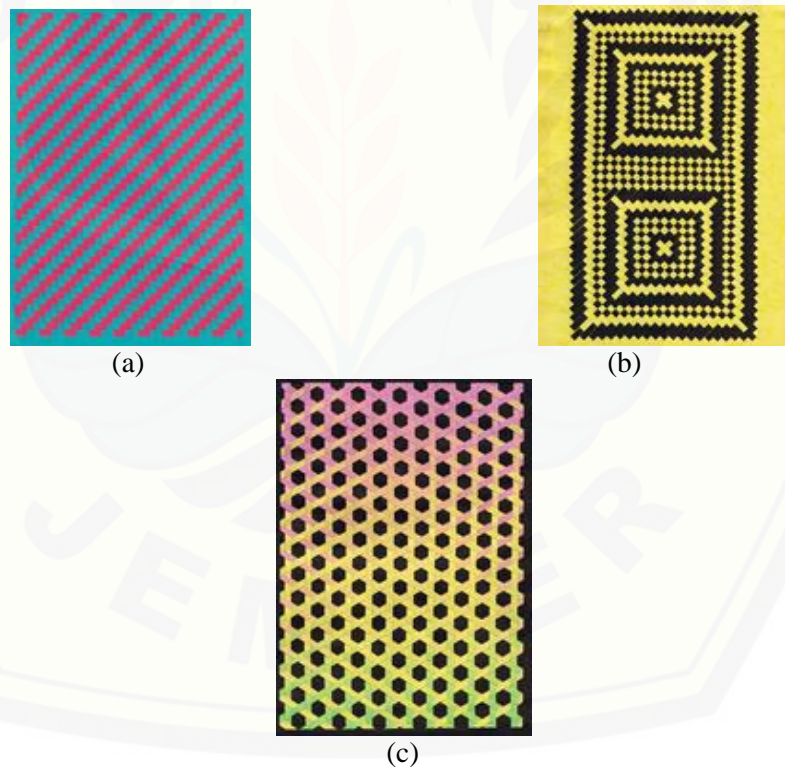


Gambar 2.2 Contoh variasi motif anyaman
(Sumber: Suria, *dkk*, 2014)

2.2 Anyaman

Kerajinan anyaman merupakan kerajinan tangan yang masih sering dibuat dan dapat dijumpai sampai saat ini. Kerajinan anyaman merupakan salah satu bentuk kerajinan yang terus dihasilkan oleh sebagian masyarakat Indonesia dengan ciri yaitu bentuk dan ornamen beragam dengan menggunakan bahan yang tersedia di alam, baik bambu, pandan, rotan dan mendong (Patria dan Mutmainah, 2015). Anyaman diartikan sebagai menganyam, mengatur (bilah, daun pandan dan sebagainya) tindih-menindih dan silang-menyilang (seperti pembuatan tikar dan bakul) (Sugiono, 2008). Anyaman dapat dibuat menggunakan berbagai macam bahan sehingga mengakibatkan terbentuknya berbagai motif anyaman. Bentuk anyaman dapat dibagi menjadi dua, yaitu anyaman dua dimensi dan anyaman tiga dimensi. Anyaman dua dimensi merupakan anyaman yang hanya memiliki ukuran lebar dan panjang saja, walaupun seandainya memiliki ketebalan, ketebalan tersebut tidak terlalu diperhitungkan. Sedangkan anyaman tiga dimensi, yaitu anyaman yang memiliki ukuran panjang, lebar, dan tinggi (Dekranas, 2011:136).

Anyaman dibagi menjadi tiga berdasarkan cara membuatnya, yaitu anyaman datar (sasak), anyaman miring (serong) dan anyaman persegi (truntum). Anyaman datar (sasak) merupakan anyaman yang dibuat datar, lebar, dan pipih. Kerajinan jenis ini banyak digunakan untuk tikar, dinding rumah tradisional, dan pembatas ruangan. Anyaman miring (serong) merupakan anyaman yang dibuat miring, biasanya berbentuk dua dimensi atau tiga dimensi. Jenis kerajinan ini banyak digunakan untuk tempat tape, keranjang dan lain sebagainya. Sedangkan anyaman persegi (truntum) merupakan anyaman yang dibuat dengan motif persegi, bisa segi tiga, segi empat, segi delapan dan seterusnya (Mutmainah, 2014). Macam-macam anyaman tersebut dapat dilihat pada Gambar 2.3 dengan motif yang berbeda-beda seperti berikut ini:



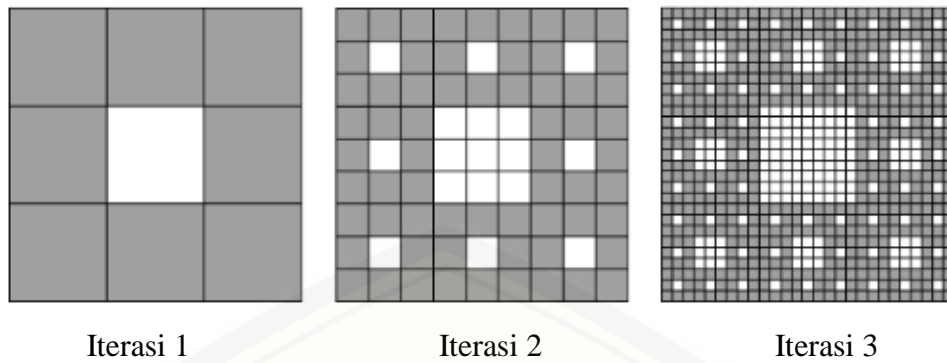
Gambar 2.3 Macam-Macam Anyaman (a) Anyaman Datar (Sasak)
(b) Anyaman Miring (Serong) (c) Anyaman Truntum
(Sumber: <https://rozisenirupa.blogspot.com/2016/02/anyaman-indonesia.html>, 2018)

2.3 Fraktal

Fraktal merupakan kata sifat yang berarti tidak teratur atau berasal dari bahasa latin yaitu *frengere* merupakan kata kerja yang berarti membelah. Fraktal merupakan objek geometris yang kasar pada segala skala, terlihat dapat dibagi-bagi dan didapatkan melalui proses iteratif serta mempunyai sifat *self-similarity* (keserupaan diri) (Mandelbrot, 1983). Selain *self-similarity*, keteraturan tertentu lainnya juga dimiliki oleh fraktal, yaitu *self-squaring*, *self-inverse* dan *self-affinity*. *Self-squaring* dapat diartikan yaitu suatu bentuk geometri fraktal yang merupakan peningkatan kerumitan dari bagian sebelumnya. *Self-inverse* artinya yaitu terdapat suatu bagian dalam geometri fraktal yang merupakan susunan yang terbalik dari susunan lainnya. Sedangkan *self-affinity* menggambarkan bahwa fraktal disusun atas bagian-bagian geometri yang saling terangkai satu sama lain (Baragar, 2001). Beberapa fraktal dapat dipecah atau dibagi-bagi menjadi beberapa bagian yang semuanya dapat terlihat identik dengan objek itu sendiri bila dilihat secara keseluruhan. Struktur fraktal dibentuk secara alami dengan adanya replikasi yang bekerja dalam diri objek tersebut pada skala yang berbeda (Aschwanden, 2011).

2.4 Karpas Sierpinski

Menurut Gough (2012), Sierpinski adalah sebuah kurva yang mengisi ruang kurva itu sendiri. Objek pada Sierpinski ini membagi dirinya sendiri dan masing-masing bagian mengisi kembali ke bagian persegi tersebut pada skala yang berbeda. Karpas Sierpinski mulai dibangun dengan persegi. Persegi dibagi menjadi sembilan kotak yang kongruen dalam 3 grid. Kemudian prosedur yang sama diterapkan secara rekursif ke dalam 8 kotak yang tersisa. Contoh Karpas Sierpinski dapat dilihat pada Gambar 2.4.

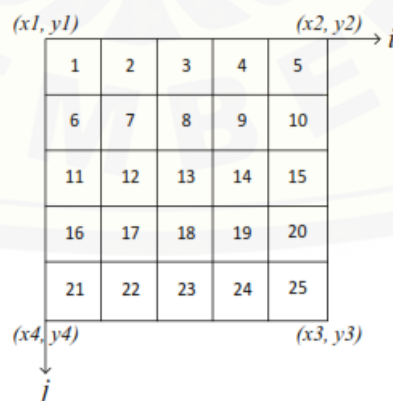


Gambar 2.4 Karpets Sierpinski
(Sumber: Hulse, 2010)

Pembangkitan karpets Sierpinski secara umum dapat dirumuskan dengan melihat jumlah iterasinya. Apabila memiliki n iterasi maka jumlah persegi berwarna yaitu 8^n , ukuran persegi $\frac{1}{9^n}$ dan total area $\frac{8^n}{9^n}$.

2.5 Koordinat dalam Grid

Motif anyaman yang dibentuk menyerupai fraktal karpets Sierpinski, membutuhkan rancangan pola dasar. Pola yang dibentuk memanfaatkan nilai koordinat-koordinat 2 dimensi. Setiap bagian dari motif anyaman yang akan dibuat terdiri dari fraktal karpets Sierpinski berbentuk persegi, oleh karena itu pembentukan tersebut dilakukan dengan cara membagi bidang sebanyak n bagian baris dan kolom dengan jarak yang sama. Pada Gambar 2.5 ditunjukkan pembagian bidang sebanyak 5 bagian.



Gambar 2.5 Pembagian Bidang
(Sumber: Suria, dkk, 2014)

Menurut Suria, *dkk* (2014), persegi memiliki 4 buah titik utama sebagai penyusunnya, yaitu koordinat (x_1, y_1) yaitu $(0,0)$ terletak pada sisi kiri atas, koordinat (x_2, y_2) yaitu $(max-width,0)$ yang terletak di sisi kanan atas dengan $max-width$ adalah lebar maksimal persegi, koordinat (x_3, y_3) yaitu $(max-width, max-lenght)$ yang terletak di sisi kanan bawah dengan $max-height$ adalah panjang maksimal persegi, dan koordinat (x_4, y_4) yaitu $(0, max-lenght)$ yang terletak di sisi kiri bawah. Titik $x_{i,j}$ merupakan simbol dari setiap koordinat titik x , dengan i merupakan baris dan j merupakan kolom. Sedangkan titik $y_{i,j}$ merupakan simbol dari setiap koordinat titik y .

Persamaan berikut dapat digunakan untuk mencari nilai koordinat titik x dan titik y pada kotak atau persegi yang akan dibangkitkan:

$$titik\ x_{i,j} = x_1 + \left(\left(\frac{(x_2 - x_1)}{n} \right) i \right), x_1 + \left(\left(\frac{(x_2 - x_1)}{n} \right) j \right)$$

$$titik\ y_{i,j} = y_1 + \left(\left(\frac{(y_4 - y_1)}{n} \right) i \right), y_1 + \left(\left(\frac{(y_4 - y_1)}{n} \right) j \right)$$

dengan:

- i = baris kotak yang akan dibangkitkan
- j = kolom kotak yang akan dibangkitkan
- x_1 = lebar minimal
- y_1 = panjang minimal
- x_2 = lebar maksimal
- y_4 = panjang maksimal
- n = ukuran grid

2.6 GUI pada Matlab

Grapichal User Interface (GUI) merupakan pemrograman yang dapat digunakan pada Matlab. Menurut Sugiharto (2006), GUIDE atau GUI builDER merupakan sebuah GUI yang menyediakan media tampilan grafis sebagai pengganti perintah teks untuk berinteraksi antara pengguna dengan program. Hasil

pemrograman yang menggunakan GUI akan jauh lebih menarik dan menjadi lebih interaktif serta penggunaan program menjadi lebih efektif.

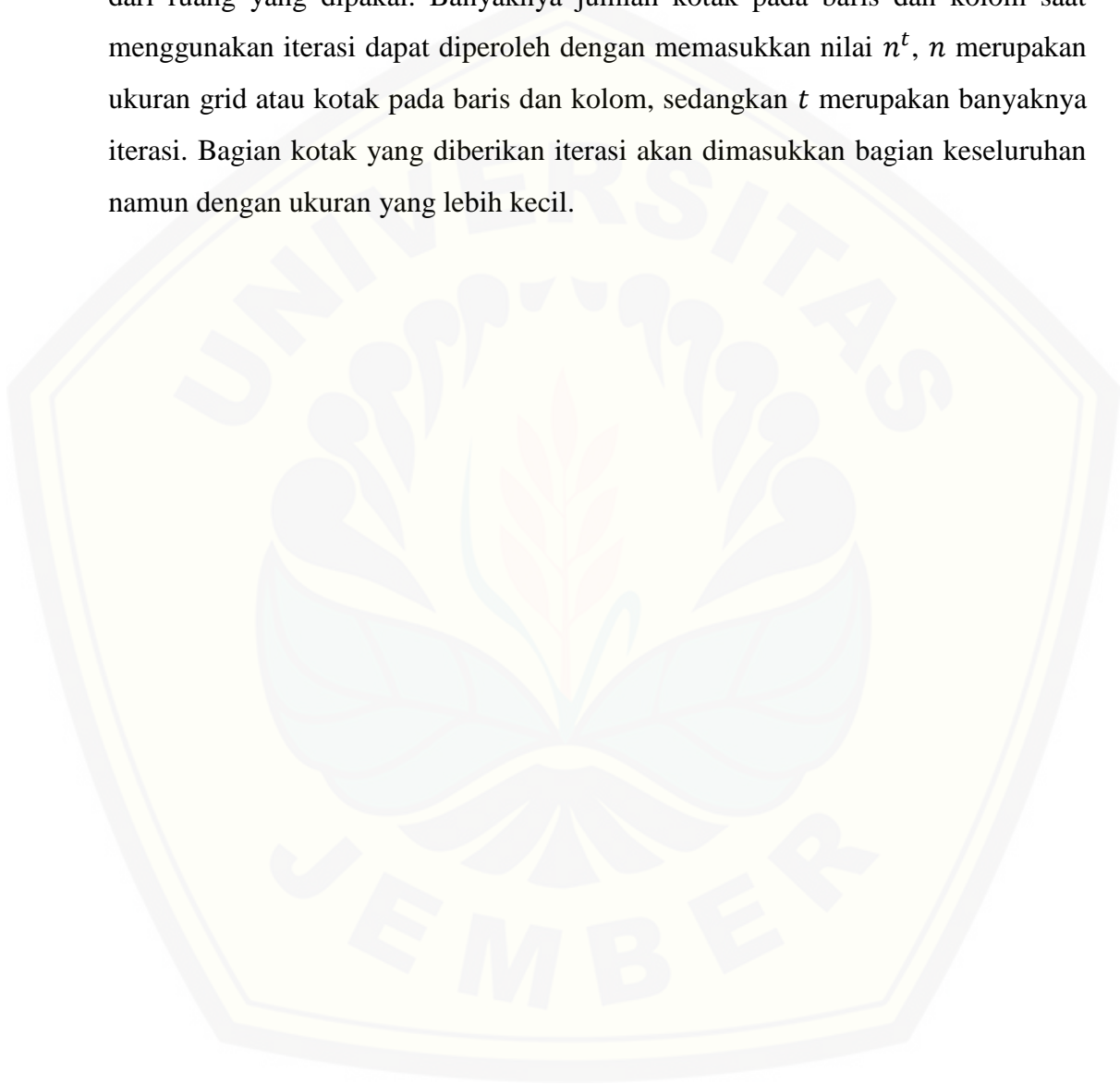
Komponen-komponen standar terdapat pada Matlab untuk keperluan membuat GUI antara lain *edit text*, *static text*, *pushbutton*, *pop-up menu*, *axes*, *uitable*, dll. *Edit text* berguna untuk memasukkan atau memodifikasi suatu teks atau tulisan. *Static text* hanya dapat menampilkan tulisan, sehingga tidak dapat dimodifikasi. *Pushbutton* merupakan jenis kontrol berupa tombol tekan yang akan menghasilkan tindakan jika diklik, misalnya tombol OK, cancel, hitung dan sebagainya. *Pop-up menu* berguna menampilkan daftar pilihan yang terdefiniskan. *Axes* berguna untuk menampilkan sebuah grafik atau gambar. *Uitable* digunakan untuk menampilkan tabel. *Axes* digunakan pada penelitian ini untuk menampilkan gambar anyaman sesuai pola yang diinginkan atau dibentuk. Pola tersebut diinputkan melalui *uitable*.

2.7 Iterated Function System (IFS)

Iterated Function System (IFS) adalah fungsi iterasi yang terdiri dari transformasi suatu citra untuk membentuk sebuah objek fraktal. Metode ini memiliki sifat *self-similarity* sehingga menghasilkan fraktal yang sama dengan objek penyusun aslinya. IFS membangun fraktal dengan cara mengulang transformasi berkali-kali untuk sebarang pola. Pola awal ditransformasi menjadi suatu pola berulang dengan struktur yang sama pada detail tertentu, hal tersebut merupakan karakteristik dasar dari himpunan fraktal (Utomo, 2011).

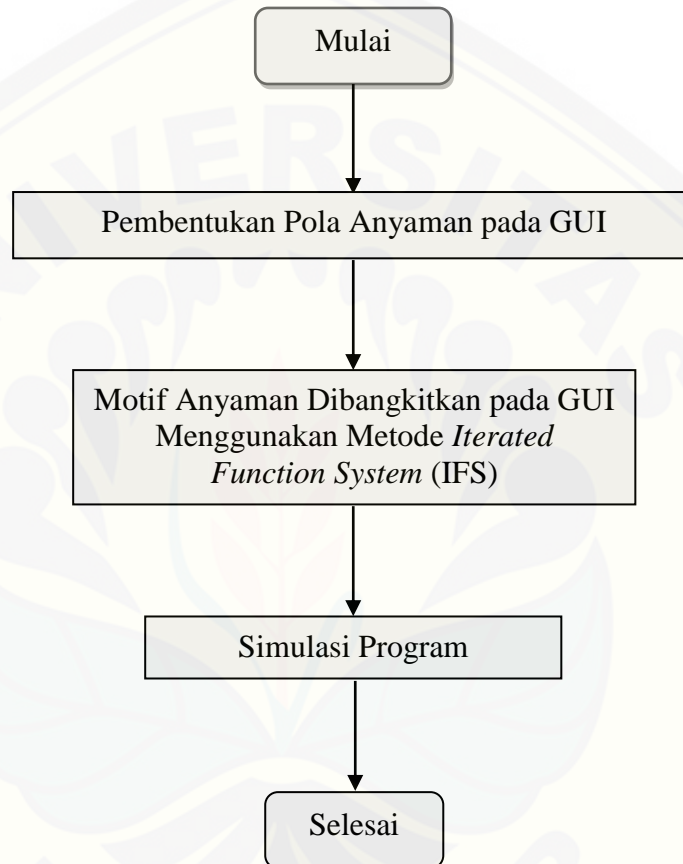
Fungsi iterasi adalah fungsi yang terdiri atas dirinya sendiri dan tak terhingga pengulangan bentuk objeknya, dalam proses ini disebut iterasi. Proses iterasi dimulai dari nilai awal, selanjutnya menentukan hasil yang akan mengakibatkan fungsi sebagai masukan dan mengulanginya beberapa kali (Falconer, 2003). Selain itu, fungsi iterasi juga merupakan *object study* yang mendalam dalam ilmu komputer. Setiap fungsi yang menggunakan konsep IFS tersebut dapat menggunakan bantuan program komputer untuk menghasilkan suatu motif tertentu yang sangat unik (Utomo, 2011).

Fraktal yang dibentuk dari IFS dapat berada pada dimensi spasial mana pun (pada dimensi apa pun). Akan tetapi, biasanya fraktal IFS dihitung dan digambar pada dua dimensi (Wahjono, *dkk*, 2011). Selain itu, IFS tidak digunakan pada semua domain, tetapi hanya pada domain yang termasuk dalam himpunan bagian dari ruang yang dipakai. Banyaknya jumlah kotak pada baris dan kolom saat menggunakan iterasi dapat diperoleh dengan memasukkan nilai n^t , n merupakan ukuran grid atau kotak pada baris dan kolom, sedangkan t merupakan banyaknya iterasi. Bagian kotak yang diberikan iterasi akan dimasukkan bagian keseluruhan namun dengan ukuran yang lebih kecil.



BAB 3. METODE PENELITIAN

Langkah-langkah yang dapat digunakan untuk menyelesaikan penelitian akan dibahas pada bab ini. Gambar 3.1 merupakan skema dari diagram alir penelitian. Langkah-langkah penyelesaian yang akan dilakukan pada penelitian ini yaitu:



Gambar 3.1 Diagram alir skema penelitian

3.1 Pembentukan Pola Anyaman pada GUI

Pola merupakan bentuk dasar atau rangkaian yang digunakan untuk menghasilkan suatu bentuk. Pola anyaman yang dibuat terdiri dari beberapa persegi. Persegi-persegi yang menjadi pola dasar dapat dikembangkan mejadi suatu fraktal. Fraktal karpet Sierpinski memiliki bentuk persegi yang dibagi menjadi beberapa bagian. Objek persegi yang ada pada karpet Sierpinski dapat dibuat sebagai bentuk awal dari motif anyaman. Sebelum membentuk motif

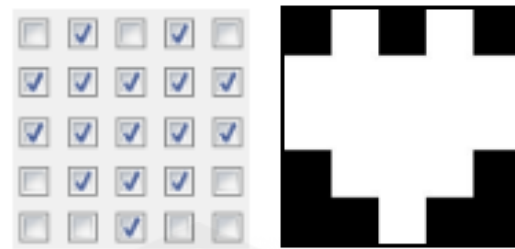
anyaman, terlebih dahulu menentukan pola anyaman. Pola dapat dibuat dengan cara menentukan ukuran grid $n \times n$ sesuai dengan motif yang diinginkan. Pembagian persegi menggunakan $n > 1$ agar dapat membentuk sebuah motif. Apabila n yang digunakan adalah 1, maka motif tidak dapat terbentuk. Setelah menentukan ukuran grid $n \times n$, langkah selanjutnya yaitu memasukkan pola pada tabel GUI. Pola dibuat dengan memasukkan bilangan bulat ke dalam tabel. Bilangan terbesar pada pola merupakan tempat untuk menempatkan IFS. Penelitian ini ingin mengaplikasikan objek karpet Sierpinski pada GUI dengan menggunakan *uitable* dan *axes*.

3.2 Motif Anyaman Dibangkitkan pada GUI Menggunakan Metode *Iterated Function System (IFS)*

Membangkitkan motif anyaman dapat dilakukan setelah terbentuknya sebuah pola ukuran grid $n \times n$. Semakin banyak grid $n \times n$ yang dibuat, maka akan semakin cepat terlihat bentuk motif anyaman dan hanya menggunakan sedikit iterasi untuk menampilkan hasil motif anyaman. Apabila menggunakan jumlah grid $n \times n$ semakin sedikit, maka motif akan lebih sulit terlihat dan membutuhkan iterasi yang semakin banyak.

GUI dapat dimanfaatkan dalam membentuk motif anyaman dengan membuat tabel ukuran grid $n \times n$. Setelah tabel dimasukkan pola anyaman, maka akan dimasukkan jumlah IFS yang diinginkan. IFS dioperasikan pada tabel yang ditandai bilangan bulat terbesar. Selain digunakan untuk tempat IFS, bilangan bulat terbesar juga digunakan untuk menentukan warna pada motif anyaman yang akan dibuat.

Motif anyaman yang ingin dibentuk pada GUI menggunakan metode IFS. Setelah itu, motif anyaman dapat terlihat lebih sempurna dan menarik dengan menggunakan 2 iterasi atau 3 iterasi seperti Gambar 3.2. Pola dasar akan dimasukkan ke dalam kotak-kotak persegi yang kosong pada tabel sehingga membentuk motif yang sempurna. Namun, apabila iterasi yang digunakan semakin banyak, maka motif yang terlihat tidak akan terlalu jelas karena gambar motif akan semakin kecil dan ukuran pixel semakin banyak.



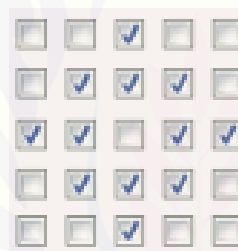
Iterasi 1



Iterasi 2



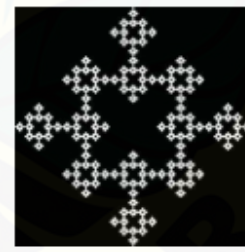
Iterasi 3



Iterasi 1



Iterasi 1



Iterasi 2

Gambar 3.2 Variasi Motif Anyaman
(Sumber: Suria, dkk, 2014)

3.3 Simulasi Program

Pada simulasi program ini digunakan aplikasi GUI Matlab dengan prosedur sebagai berikut:

1. Menentukan ukuran pola $n \times n$, pada penelitian ini $n > 1, n = 2, 3, 4, \dots$;
2. Membuat tabel ukuran $n \times n$ pada *uitable* ;
3. Memasukkan pola pada tabel ;

(Pola berupa bilangan bulat, bilangan bulat terbesar pada pola sebagai *self-similarity*);

4. Menentukan jumlah iterasi ;
5. Membangkitkan karpet Sierpinski sesuai pola dengan iterasi yang telah ditentukan ;
6. Hasil ditampilkan di *axes*.



BAB 5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan maka diperoleh kesimpulan sebagai berikut:

- a. Pola dasar anyaman yang digunakan untuk membuat motif anyaman dipengaruhi oleh ukuran grid n baris dan n kolom. Semakin sedikit grid $n \times n$ yang digunakan, maka motif anyaman akan semakin sulit terbentuk. Sedangkan semakin banyak grid $n \times n$ yang digunakan, maka motif anyaman akan semakin mudah terbentuk
- b. Pola dibuat dengan menginputkan bilangan bulat. Apabila jumlah kotak yang ditandai dengan bilangan bulat terbesar semakin banyak, maka pola dasar akan semakin sulit terbentuk. Sedangkan apabila jumlah kotak yang ditandai dengan bilangan bulat terbesar semakin sedikit, maka pola dasar akan semakin mudah terbentuk.
- c. Motif anyaman dapat dibuat bervariasi dengan memanfaatkan metode IFS yang ada pada fraktal karpet Sierpinski.
- d. Motif anyaman akan terlihat lebih menarik dan bervariasi apabila diberikan beberapa iterasi pada pola dasar anyaman. Iterasi yang diberikan pada pola disesuaikan dengan ukuran grid $n \times n$. Apabila menggunakan pola dasar dengan ukuran grid semakin sedikit, maka iterasi yang diberikan akan semakin banyak. Jika menggunakan pola dasar dengan ukuran grid semakin banyak, maka iterasi yang diberikan akan semakin sedikit.

5.2 Saran

Penelitian ini menggunakan ukuran grid $n \times n$ untuk membuat sebuah motif anyaman. Pada penelitian selanjutnya dapat dilakukan pembuatan motif anyaman dengan ukuran grid $m \times n$ agar lebih banyak variasi motif yang dihasilkan. Metode IFS dapat diletakkan pada kotak selain yang bertanda bilangan bulat terbesar. Iterasi dapat dilakukan pada beberapa kotak yang diinginkan dengan menambahkan persamaan pada program. Selain itu, penelitian ini memiliki

kelemahan yaitu apabila iterasi yang digunakan untuk membuat motif anyaman terlalu besar dan tidak sesuai dengan ukuran grid, maka akan terjadi eror. Oleh karena itu penelitian selanjutnya dapat menggunakan aplikasi pemrograman selain GUI pada Matlab.



DAFTAR PUSTAKA

- Aschwanden, M. 2011. *Self-Organized Critically in Astrophysics: The Statistic of Nonlinear Processes in The Universe*. New York: Springer.
- Baragar, A. 2001. *A Survey Of Classical and Modern Geometry*. Prentice Hall: New Jersey.
- Dekranas. 2011. *Permata Tersembunyi Kalimantan Timur, Seni Kriya Kutai Barat, Malinau, Nunukan*. Jakarta: Dewan Kerajinan Nasional.
- Falconer, K. 2003. *Fractal Geometry: Mathematical Foundations and Applications*. West Sussex: John Wiley & Sons.
- Gough, J. 2012. *Hilbert and Sierpinski Spacefilling Curves, and Beyond*. Australia: Gale Education, Religion and Humanities Lite Package.
- Hasang, S. dan S. Suparjo. 2012. Geometri Fraktal dalam Rancangan Arsitektur. *Media Matrasan* 9 (1): 111-124.
- Hulse, M. 2010. From Sierpinski Carpets to Directed Graphs. *Aberystwyth Wales* 23 (3): 46-71.
- Mandelbrot, B.B. 1983. *The Fractal Geometry of Nature*. New York: W.H. Freeman Company.
- Manik, N.I. dan Manal. 2012. Penggunaan Model Fraktal untuk Pengembangan Motif Ulos. *Mat Stat* 12 (2): 143-151.
- Mutmainah, S. 2014. *Buku Ajar: Kriya Anyam*. Surabaya: Jurusan Seni Rupa Fakultas Bahasa dan Seni UNESA.
- Patria, A.S. dan S. Mutmainah. 2015. Kerajinan Anyam Sebagai Pelestarian Kearifan Lokal. *Dimensi* 12 (1): 1-4.
- Prasasti, I. 2018. Pemanfaatan Metode Iterated Function System dalam Pengembangan Motif Anyaman. *Skripsi*. Jember: Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.
- Rozi. 2017. *Anyaman Indonesia*. <https://rozisenirupa.blogspot.com/2016/02/anyaman-indonesia.html>. [Diakses pada Mei 2018].
- Santoso, P. I. 1994. *Grafika Komputer dan Antarmuka Grafis Teknik Penyusunan Program Aplikasi Berbasis Grafis yang Profesional*. Yogyakarta: Andi Offset.

- Situngkir, H. 2013. *Cellular-Automata and Innovation within Indonesian Traditional Weaving Crafts*. Munich Personal RePEc Archive, (Online), (<http://mpa.ub.unimuenchen.de/51608/>), diakses 20 April 2018).
- Sugiharto, A. 2006. *Pemrograman GUI dengan Matlab*. Yogyakarta: Andi Offset.
- Sugiono, D. 2008. *Kamus Besar Bahasa Indonesia*. Jakarta: Gramedia Pustaka Utama.
- Suria, O., M. Kartika. dan W. Kusuma. 2014. Membuat motif anyaman bervariasi dengan menggunakan fraktal sierpinski carpet. *Sientika* 2 (1): 511-519.
- Utomo, B. 2011. Fraktal dan Invers Fraktal. *Jurnal Matematika* 2 (1): 28-37.
- Wahjono, T.D., S. Karim. dan B. Riyadi. 2011. Perancangan Perangkat Lunak Generator Gambar dan Musik Fraktal dengan Metode *Iterated Function System*. *CommIT* 1(2): 140-149

LAMPIRAN

4.1 Script Program Matlab

```
% GUI_INGKA MATLAB code for Gui_Ingka.fig
% GUI_INGKA, by itself, creates a new GUI_INGKA or raises the
existing
% singleton*.
% H = GUI_INGKA returns the handle to a new GUI_INGKA or the
handle to
% the existing singleton*.
% GUI_INGKA('CALLBACK',hObject,eventData,handles,...) calls
the local
% function named CALLBACK in GUI_INGKA.M with the given input
arguments.
% GUI_INGKA('Property','Value',...) creates a new GUI_INGKA
or raises the
% existing singleton*. Starting from the left, property
value pairs are
% applied to the GUI before Gui_Ingka_OpeningFcn gets called.
An
% unrecognized property name or invalid value makes property
application
% stop. All inputs are passed to Gui_Ingka_OpeningFcn via
varargin.
% *See GUI Options on GUIDE's Tools menu. Choose "GUI allows
only one
% instance to run (singleton)".
% See also: GUIDE, GUIDATA, GUIHANDLES
% Edit the above text to modify the response to help Gui_Ingka
% Last Modified by GUIDE v2.5 14-Nov-2018 21:14:01
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name', mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @Gui_Ingka_OpeningFcn, ...
                  'gui_OutputFcn', @Gui_Ingka_OutputFcn, ...
                  'gui_LayoutFcn', [] , ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before Gui_Ingka is made visible.
function Gui_Ingka_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
```



```

% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Gui_Ingka (see VARARGIN)
% Choose default command line output for Gui_Ingka
handles.output = hObject;
% Update handles structure
guidata(hObject, handles);

movegui(gcf, 'center');
set(gcf, 'Name', 'Motif Anyaman');
set(handles.edit1, 'string', '');
set(handlesuitable1, 'data', [], 'rowname', '', 'columnname', '');
set(handles.edit2, 'string', '');
set(handles.popupmenu1, 'value', 1);
cla(handles.axes1, 'reset');
axes(handles.axes1);
set(handles.axes1, 'XTick', [], 'YTick', []);
% UIWAIT makes Gui_Ingka wait for user response (see UIRESUME)
% uiwait(handles.figure1);
% --- Outputs from this function are returned to the command line.
function varargout = Gui_Ingka_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Get default command line output from handles structure
varargout{1} = handles.output;
function edit1_Callback(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hints: get(hObject, 'String') returns contents of edit1 as text
%         str2double(get(hObject, 'String')) returns contents of
edit1 as a double
% --- Executes during object creation, after setting all
properties.
function edit1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called
% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject, 'BackgroundColor'),
get(0, 'defaultUiControlBackgroundColor'))
    set(hObject, 'BackgroundColor', 'white');

% --- Executes on button press in pushbutton1.
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)
clc;
N=str2num(get(handles.edit1,'string'));
for i=1:N
    colwidth{i}=40;

set(handles.uitable1,'data',ones(N),'rowname','numbered','columnname',
'numbered',...
'columneditable',true,'columnwidth',colwidth);

% --- Executes on button press in pushbutton2.
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)
clc;
cla(handles.axes1,'reset');
axes(handles.axes1);
set(handles.axes1,'XTick',[],'YTick',[],'UserData',[]);
pause(0.1);

pola=get(handles.uitable1,'data');

N=size(pola,1);

Iter=str2num(get(handles.edit2,'string'));
Mapping=get(handles.popupmenu1,'value');

Fractal=pola;
for t=2:Iter
    temp=Fractal;
    for i=1:N
        for j=1:N
            if pola(i,j)==max(max(pola));
                Fractal((i-1)*N^(t-1)+(1:N^(t-1)),(j-1)*N^(t-1)+(1:N^(t-1)))=temp;
            else
                Fractal((i-1)*N^(t-1)+(1:N^(t-1)),(j-1)*N^(t-1)+(1:N^(t-1)))=repmat(pola(i,j),N^(t-1),N^(t-1));
            end
        end
    end
end

axes(handles.axes1);
imagesc(Fractal);
if Mapping==1
    MyMap=gray;
    colormap(MyMap);
elseif Mapping==2
    MyMap=[0 0 0;1 0 0;1 1 1];
    colormap(MyMap);

```

```

elseif Mapping==3
    MyMap=[0 0 0;0 1 0;1 1 1];
    colormap(MyMap);
elseif Mapping==4
    MyMap=[0 0 0;0 0 1;1 1 1];
    colormap(MyMap);
elseif Mapping==5
    kbw=[0 0 0;1 0 0;0 1 0;0 0 1;1 1 0;1 0 1;0 1 1;1 1 1];
    NC=max(max(pola))+1;
    wrn=randperm(8);
    MyMap=kbw(wrn(1:NC),:);
    colormap(MyMap);
else
    MyMap=rand(max(max(pola))+1,3);
    colormap(MyMap);
end
axis('off')
set(handles.axes1,'UserData',MyMap);

function edit2_Callback(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of edit2 as text
%        str2double(get(hObject,'String')) returns contents of
edit2 as a double

% --- Executes during object creation, after setting all
properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to edit2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    empty - handles not created until after all
CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% -----
% -----
function uipushtool1_ClickedCallback(hObject, eventdata, handles)
% hObject    handle to uipushtool1 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;

```

```

set(gcf,'Name','Motif Anyaman');
set(handles.edit1,'string','');
set(handles.uitable1,'data',[],'rowname','','columnname','');
set(handles.edit2,'string','');
set(handles.popupmenu1,'value',1);
cla(handles.axes1,'reset');
axes(handles.axes1);
set(handles.axes1,'XTick',[],'YTick',[]);

% -----
----
function uipushtool2_ClickedCallback(hObject, eventdata, handles)
% hObject    handle to uipushtool2 (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB
% handles    structure with handles and user data (see GUIDATA)
clc;
Fractal=getimage(handles.axes1);
if ~isempty(Fractal)
    [name_file,name_path] = uiputfile( ...
        {'*.jpg','File Type JPEG (*.jpg)';
        '*.bmp','File Type BITMAP (*.bmp)';
        '*.tif','File Type TIF (*.tif)';
        '*.png','File Type PNG (*.png)';
        '*.jpg;*.bmp;*.tif;*.png','Files of type
(*.jpg,*.bmp,*.tif,*.png)'},...
        'Save As Image');
    if name_file~=0
        MyMap=get(handles.axes1,'UserData');
        imwrite(Fractal+1,MyMap,fullfile(name_path,name_file));
    end
end

% --- Executes on button press in pushbutton3.
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)
clc;
N=str2num(get(handles.edit1,'string'));
for i=1:N
    colwidth{i}=40;
end
set(handles.uitable1,'data',round(rand(N)), 'rowname','numbered','c
olumnname','numbered',...
    'columneditable',true,'columnwidth',colwidth);

% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu1 (see GCBO)

```

```
% eventdata reserved - to be defined in a future version of
MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns
popupmenu1 contents as cell array
% contents{get(hObject,'Value')} returns selected item from
popupmenu1

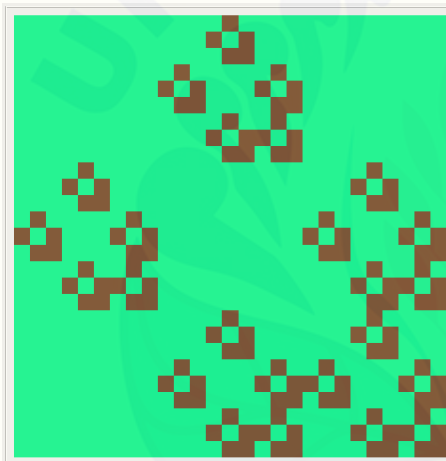
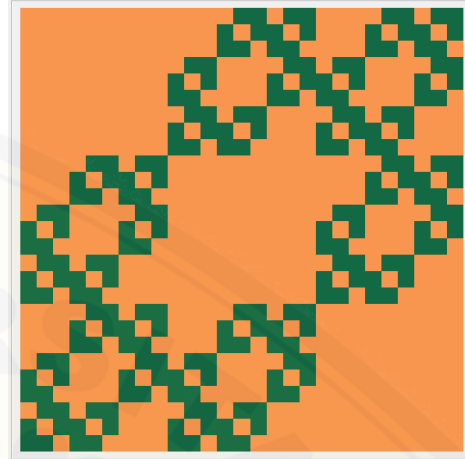
% --- Executes during object creation, after setting all
properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
% hObject handle to popupmenu1 (see GCBO)
% eventdata reserved - to be defined in a future version of
MATLAB
% handles empty - handles not created until after all
CreateFcns called

% Hint: popupmenu controls usually have a white background on
Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');

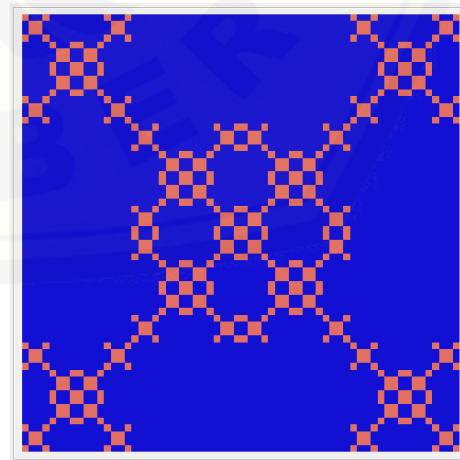
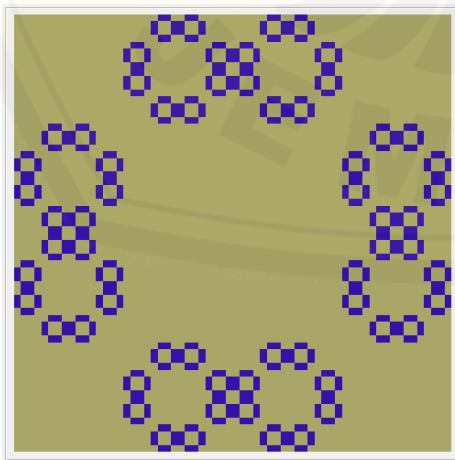
% --- Executes when entered data in editable cell(s) in uitable1.
function uitable1_CellEditCallback(hObject, eventdata, handles)
% hObject handle to uitable1 (see GCBO)
% eventdata structure with the following fields (see
MATLAB.UI.CONTROL.TABLE)
% Indices: row and column indices of the cell(s) edited
% PreviousData: previous data for the cell(s) edited
% EditData: string(s) entered by the user
% NewData: EditData or its converted form set on the Data
property. Empty if Data was not changed
% Error: error string when failed to convert EditData to
appropriate value for Data
% handles structure with handles and user data (see GUIDATA)
```

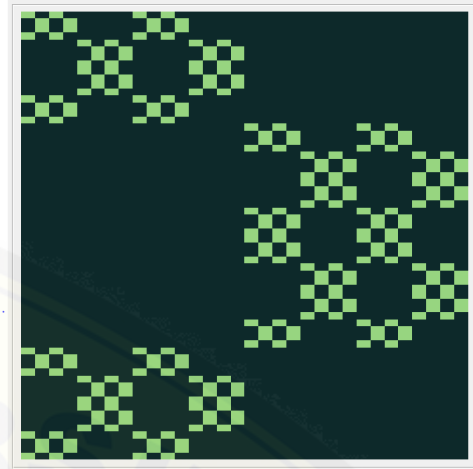
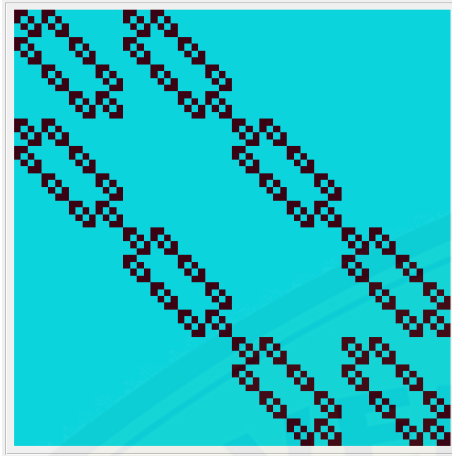
4.2 Motif Anyaman

a. Ukuran grid 3×3



b. Ukuran grid 4×4





c. Ukuran grid 5×5

