



**INOVASI DESAIN BATIK FRAKTAL MENGGUNAKAN
GEOMETRI FRAKTAL KOCH *SNOWFLAKE* (m, n, c)**

SKRIPSI

Oleh

**Dyakza Hadi Pramestika Putri
NIM 151810101029**

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



**INOVASI DESAIN BATIK FRAKTAL MENGGUNAKAN
GEOMETRI FRAKTAL KOCH SNOWFLAKE (m, n, c)**

SKRIPSI

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

Oleh

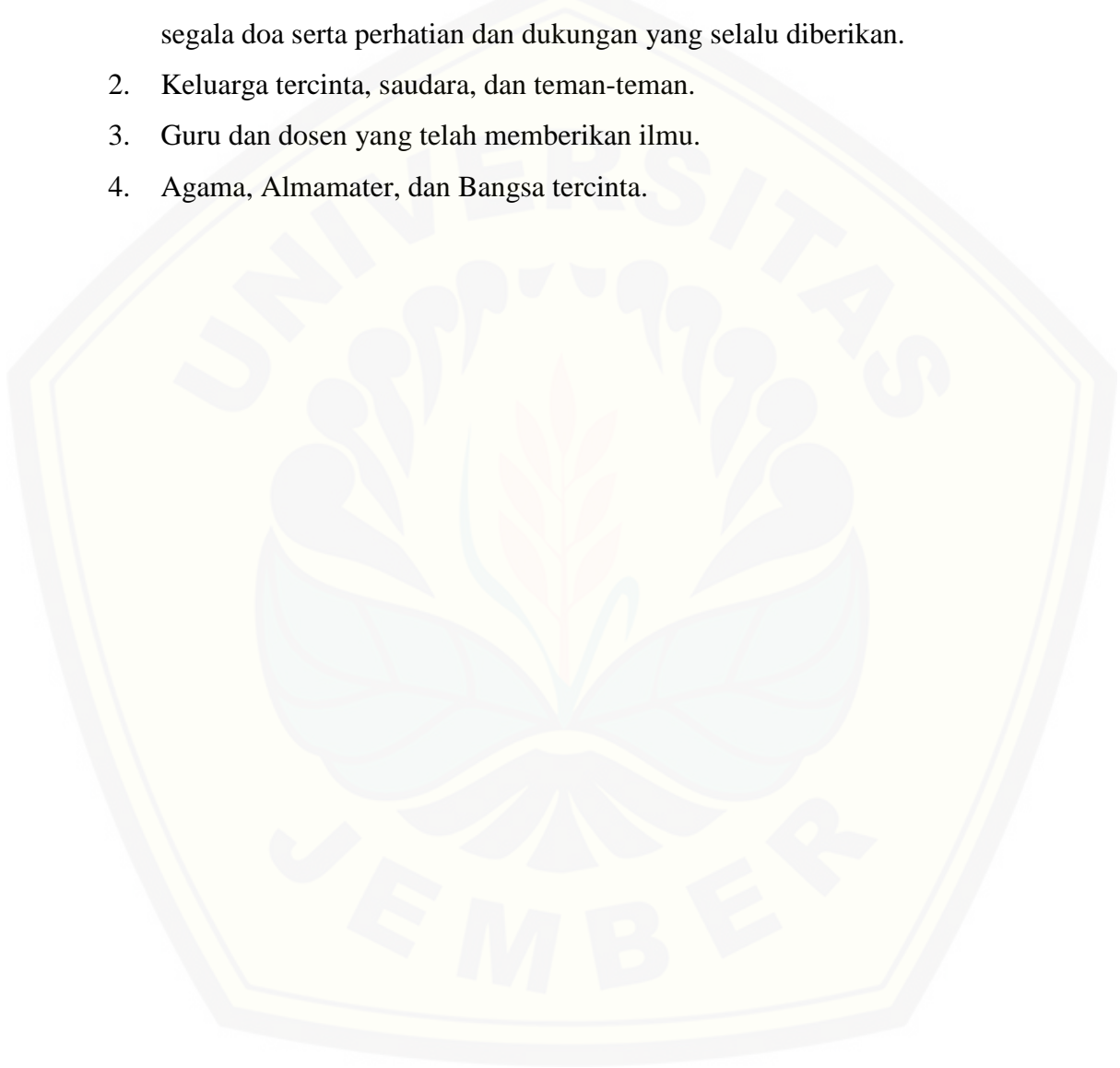
Dyakza Hadi Pramestika Putri
NIM 151810101029

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

PERSEMBAHAN

Dengan mengucap Bismillahirrahmanirrahim, Tugas akhir ini saya persembahkan kepada:

1. Kedua orang tua saya, Bapak Hadi Soeprijanto dan Ibu Diyah Sri Rohani atas segala doa serta perhatian dan dukungan yang selalu diberikan.
2. Keluarga tercinta, saudara, dan teman-teman.
3. Guru dan dosen yang telah memberikan ilmu.
4. Agama, Almamater, dan Bangsa tercinta.



MOTTO

My mother thought me that **there is strength beyond my weaker days, and that perseverance is the legs for those trying to reach better days.** I watch her do the work of hundred man to make sure that I become **ONE**. Sacrifice like this only resurrects hope in me when I'm nailed down.

(Anonim)



PERNYATAAN

Saya yang bertanda tangan dibawah ini:

Nama : Dyakza Hadi Pramestika Putri

NIM : 151810101029

menyatakan dengan sebenar-benarnya bahwa tugas akhir yang berjudul “Inovasi Desain Batik Fraktal menggunakan Geometri Fraktal Koch *Snowflake* (m, n, c)” adalah benar-benar hasil karya sendiri, kecuali kutipan yang telah 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 sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata dikemudian hari pernyataan ini tidak benar.

Jember, Juli 2019

Yang menyatakan,

Dyakza Hadi Pramestika Putri

NIM. 151810101029

SKRIPSI

**INOVASI DESAIN BATIK FRAKTAL MENGGUNAKAN GEOMETRI
FRAKTAL KOCH SNOWFLAKE (m, n, c)**

oleh

Dyakza Hadi Pramestika Putri

NIM 151810101029

Pembimbing

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

Dosen Pembimbing Anggota : Ahmad Kamsyakawuni, S.Si., M.Kom

PENGESAHAN

Tugas akhir yang berjudul “Inovasi Desain Batik Fraktal menggunakan Geometri Fraktal Koch *Snowflake* (m, n, c)” 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

Ahmad Kamsyakawuni, S.Si., M.Kom
NIP. 197211291998021001

Anggota II,

Anggota III,

Ika Hesti Agustin, S.Si., M.Si
NIP. 198408012008012006

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

Mengesahkan,
Dekan,

Drs. Sujito, Ph.D
NIP. 196102041987111001

RINGKASAN

Inovasi Desain Batik Fraktal menggunakan Geometri Fraktal Koch *Snowflake* (m, n, c); Dyakza Hadi Pramestika Putri, 151810101029; 66 halaman; 2019: Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam

Batik fraktal tersusun atas berbagai macam bentuk geometri fraktal, contohnya adalah bentuk geometri fraktal Koch *snowflake* dan Koch anti-*snowflake*. *Snowflake* yang merupakan fraktal adalah ketika *snowflake* terbentuk dari bentuknya sendiri atau disebut dengan *self-similar structure*. Proses ini merupakan sifat dimana sebuah cabang yang mempunyai subcabang, yang dimana tiap subcabang merupakan cabang yang diperkecil dan akan terus berlanjut sampai bentuk yang sangat kecil.

Tujuan dari penulisan penelitian ini adalah membangun algoritma batik sehingga menghasilkan inovasi desain batik fraktal. Batik fraktal disusun atas geometri fraktal Koch *snowflake* dan Koch anti-*snowflake*. Bentuk Koch sendiri didapatkan dari hasil pembangkitan menggunakan metode IFS. Langkah awal pembangkitan adalah menentukan nilai m, n, c kemudian menentukan segmen awal, melakukan dilasi, rotasi, dan translasi. Proses pembangkitan akan dilakukan sebanyak 2 iterasi karena jika dilakukan lebih dari 2 iterasi, hasil pembangkitan tidak terlihat jelas. Ornamen yang telah dibangkitkan disusun dari arah kiri ke kanan dan bawah ke atas dengan skala ornamen terbesar ke terkecil. Bentuk Koch *snowflake* dan Koch anti-*snowflake* yang digunakan sebanyak 8 bentuk. Pola yang digunakan sebanyak 5 pola.

Hasil penelitian yang didapat ialah terdapat kombinasi ornamen pada Pola 1, Pola 2, Pola 3 sebanyak 64 kombinasi dan Pola 4, Pola 5 sebanyak 512 kombinasi. Penyusunan ornamen pada Pola 1 mempunyai kemiripan dengan penyusunan ornamen pada motif batik parang rusak sedangkan penyusunan Pola 5 mempunyai kemiripan dengan penyusunan ornamen pada motif batik nitik.

PRAKATA

Alhamdulillah puji syukur yang tak terhingga penulis panjatkan kehadirat Allah SWT atas segala rahmat, barokah dan hidayah-Nya, sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “Inovasi Desain Batik Fraktal menggunakan Geometri Fraktal Koch *Snowflake* (m, n, c)”. Penulisan tugas akhir ini dimaksudkan untuk memenuhi salah satu syarat menyelesaikan pendidikan Strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember.

Dalam penulisan tugas akhir ini, penulis mendapatkan bantuan dan dorongan baik secara langsung maupun tidak langsung dari berbagai pihak. Untuk itu penulis menyampaikan terima kasih kepada:

1. Kosala Dwidja Purnomo, S.Si., M.Si. selaku Dosen Pembimbing Utama dan Ahmad Kamsyakawuni, S.Si., M.Kom., selaku Dosen Pembimbing Anggota yang telah memberikan bimbingan dan arahan sehingga tugas akhir ini dapat terselesaikan dengan baik;
2. Ika Hesti Agustin, S.Si., M.Si. dan Bagus Juliyanto, S.Si., M.Si. selaku dosen penguji yang telah memberikan kritik, saran dan masukan sehingga tugas akhir ini dapat terselesaikan dengan baik;
3. Kedua orang tua dan adik-adik saya yang telah memberikan dorongan semangat dan doa;
4. Teman-teman SIGMA '15 atas bantuan dan semangatnya;
5. Teman seperjuangan skripsi, Ellenda Alkhori, Hikmah Ardiantika Sari, Choirun Nisa Ulfa, Laylatul Febriana Nila Sari dan Ayu Wulandari;
6. Sahabat terdekat KSS (Kita Selalu Sukses), Lovika Augusta Purwaningtyas, Salwa Syams Salsabila, Ega Putri Nurwita, Desita Prihapsari dan Fahrin Saftya Devina;
7. Laki-laki terkasih, Dhimas Agta Nugroho yang selalu memberikan semangatnya;
8. Serta semua pihak yang telah membantu kelancaran penulisan tugas akhir ini yang tidak bisa disebutkan satu persatu.

Penulis juga menerima segala kritik serta saran demi kesempurnaan tugas akhir ini. Penulis berharap tugas akhir ini dapat bermanfaat bagi pembaca.

Jember, Juli 2019

Penulis



DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PERSEMBAHAN	ii
HALAMAN MOTTO	iii
HALAMAN PERNYATAAN	iv
HALAMAN PEMBIMBING	v
HALAMAN PENGESAHAN	vi
RINGKASAN	vii
PRAKATA	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiv
BAB 1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	3
1.5 Manfaat	3
BAB 2. TINJAUAN PUSTAKA	4
2.1 Batik	4
2.2 Fraktal	5
2.3 Koch <i>Snowflake</i>	7
2.4 Koch <i>Anti-Snowflake</i>	8
2.5 <i>Iterated Function System (IFS)</i>	9
2.5 GUI pada Matlab	12
BAB 3. METODE PENELITIAN	14
3.1 Membangun Model IFS	15
3.2 Pola Batik	16
3.3 Pembuatan Program	17

BAB 4. HASIL DAN PEMBAHASAN	19
4.1 Hasil	19
4.1.1 Pembangkitan Koch <i>Snowflake</i> dan Koch <i>Anti-Snowflake</i>	
4.1.2 Pembuatan Pola Batik.....	22
4.1.3 Pembuatan Program.....	37
4.1.4 Penggabungan Desain Batik.....	39
4.2 Pembahasan	43
BAB 5. PENUTUP	45
5.1 Hasil	45
5.2 Saran	45
DAFTAR PUSTAKA	46
Lampiran	48

DAFTAR GAMBAR

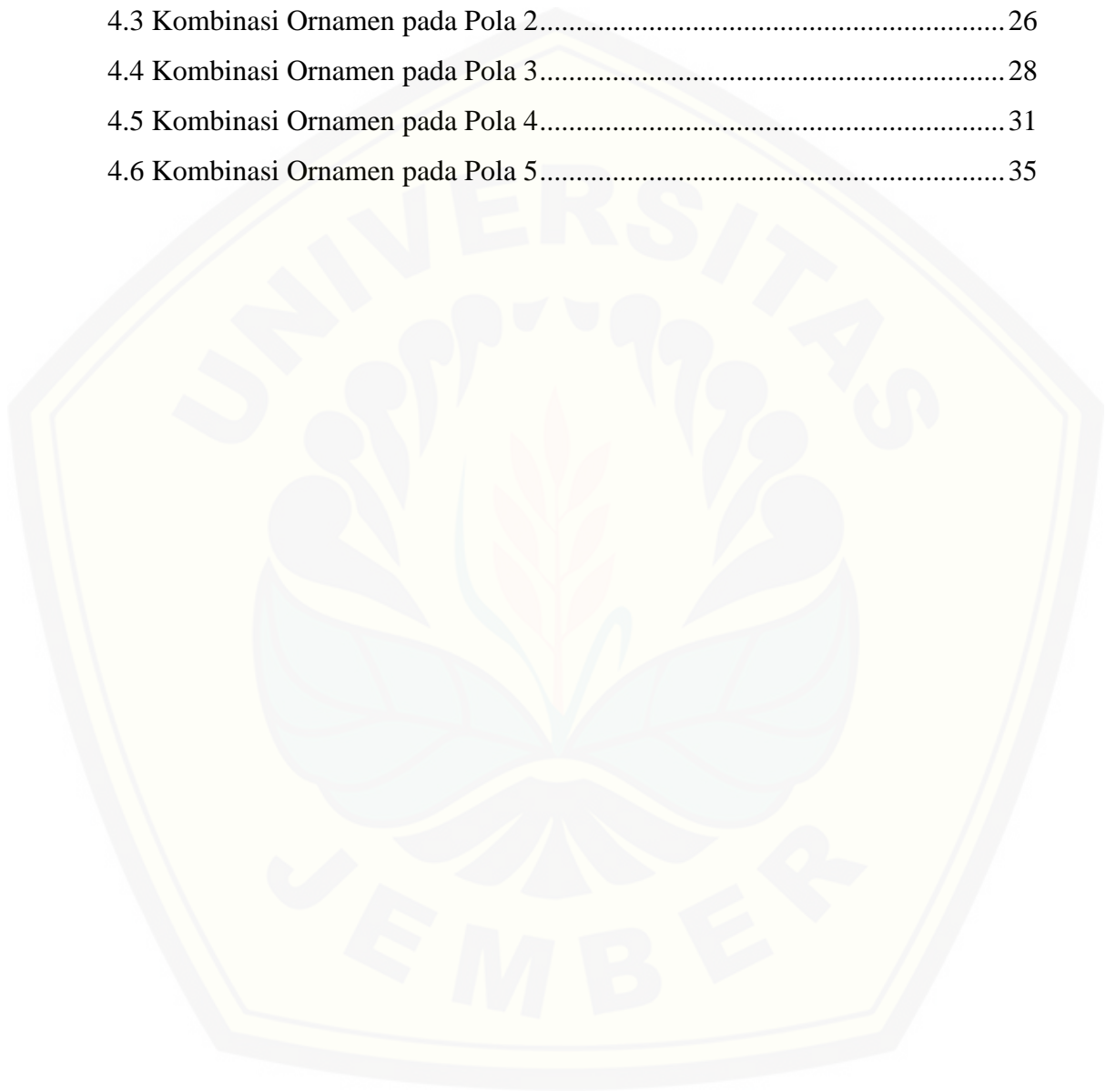
	Halaman
2.1 Batik Motif Parang Rusak	5
2.2 Batik Motif Nitik	5
2.3 Pola Pembangkitan Kurva Koch <i>Snowflake</i>	8
2.4 Pembangkitan Kurva Koch <i>Anti-Snowflake</i>	9
2.5 Kontraksi dari Q.....	10
2.6 Ilustrasi Efek dari Pemiripan pada Suatu Bujursangkar Satuan U.....	11
2.7 Himpunan-himpunan yang Tidak Saling Tumpang Tindih	12
2.8 Transformasi <i>Affine</i>	12
3.1 Skema Penelitian	14
3.2 Pola Dasar Batik 1	16
3.3 Pola Dasar Batik 2.....	16
3.4 Pola Dasar Batik 3.....	17
3.5 Pola Dasar Batik 4	17
3.6 Pola Dasar Batik 5.....	17
4.1 Proses Dilasi pada L	20
4.2 Proses Rotasi $\overline{A'B'}$ di Titik B' pada Sisi Poligon Segi- m	20
4.3 Hasil Rotasi θ Sebanyak $(n - 2)$ Kali pada Poligon Segi- m di Titik B' dan Translasi Sebesar $\left(\frac{1+c}{2}\right)$	21
4.4 Ilustrasi Kolom dan Baris pada Ornamen 2	27
4.5 Ilustrasi Kolom dan Baris pada Ornamen 3	30
4.6 Ilustrasi Kolom dan Baris pada Ornamen 3	34
4.7 Tampilan Desain Program.....	37
4.8 Program setelah dijalankan	39
4.9 Penggabungan Pola 1 dan Motif Gajah Oling 1	40
4.10 Penggabungan Pola 2 dan Motif Gajah Oling 2	40
4.11 Penggabungan Pola 3 dan Motif Gajah Oling 3	41
4.12 Penggabungan Pola 4 dan Motif Daun Tembakau.....	42

4.13 Penggabungan Pola 5 dan Motif Biji Kopi	42
4.14 Perbandingan Pola Batik 1 dan Pola Batik Motif Parang Rusak	43
4.15 Perbandingan Pola Batik 5 dan Pola Batik Motif Nitik	43



DAFTAR TABEL

4.1 Pembangkitan Koch <i>Snowflake</i> dan Koch <i>Anti-Snowflake</i>	21
4.2 Kombinasi Ornamen pada Pola 1	24
4.3 Kombinasi Ornamen pada Pola 2	26
4.4 Kombinasi Ornamen pada Pola 3	28
4.5 Kombinasi Ornamen pada Pola 4	31
4.6 Kombinasi Ornamen pada Pola 5	35



BAB 1. PENDAHULUAN

1.1 Latar Belakang

Batik merupakan kain bergambar yang pembuatannya secara khusus dengan menuliskan atau menerapkan malam pada kain, kemudian diproses dengan cara tertentu yang memiliki kekhasan. Pola batik merupakan rancangan tersusun atas garis-garis alami untuk mendapatkan hasil yang indah. Pola batik yang tidak teratur tetap bersifat alami biasanya digunakan pada kain (Santi, 2011). Pembuatan motif batik telah mengalami banyak kemajuan, tidak hanya menggunakan metode tulisan atau cap saja. Pembuatan motif batik dapat dilakukan dengan menerapkan perhitungan matematis.

Geometri fraktal adalah cabang matematika yang mempelajari sifat-sifat dan perilaku berbagai jenis fraktal (Kamsyakawuni dkk, 2017). Fraktal berasal dari kata *fractus* yang berarti pecah. Berdasarkan definisi umum, fraktal dapat diartikan sebagai pengulangan bentuk geometri yang dibentuk dari bentuk primitif geometri tersebut dipecah atau dibagi kedalam bentuk dengan ukuran panjang dan lebar dan posisi tertentu. Konsep fraktal dapat menguraikan sifat fisis yang rumit menjadi elemen yang lebih sederhana. Proses yang semakin lama membentuk suatu keteraturan tertentu, yakni *self-similarity*, *self-affinity*, *self-inverse*, dan *self-squaring* (Riwansia, 2016).

Batik fraktal merupakan inovasi karya seni Indonesia yang dimodelkan dan dirancang secara sains modern. Batik fraktal ditemukan ketika *Pixel People Project* menganalisis secara ilmiah bahwa banyak corak batik tradisional Indonesia yang memenuhi sifat-sifat geometri fraktal (Mursito, 2009). Batik fraktal adalah batik yang sentuhan desainnya (corak dan ragam hiasnya) dibuat menggunakan rumus-rumus matematika yang dikerjakan dengan teknologi komputer (Kudiya, 2009).

Penelitian mengenai batik fraktal telah banyak dilakukan sebelumnya. Santi (2011) membuat motif batik menggunakan fraktal. Santi menjelaskan tentang batik yang diuji dengan menggunakan metode transformasi *fourier* yang menunjukkan dimensi batik adalah bilangan pecahan sesuai dengan karakter fraktal. Kamsyakawuni dkk (2017) menjelaskan tentang mengembangkan desain batik

labako dengan menggabungkan geometri fraktal kurva naga dan corak tembakau. Riwansia (2016) menjelaskan tentang pengembangan desain batik melalui penggunaan geometri fraktal Koch *snowflake*. Pada penelitiannya, Riwansia (2016) menggunakan geometri fraktal Koch *snowflake* (n, c) yaitu $(3, \frac{1}{3})$ dan metode pembangkitan Koch *snowflake* menggunakan *L-system*.

Berdasarkan uraian diatas, pada tugas akhir ini penulis akan mengembangkan langkah-langkah penyusunan ornamen pada desain batik menggunakan tidak hanya geometri fraktal Koch *snowflake* tetapi juga Koch anti-*snowflake* dengan (m, n, c) bervariasi sehingga desain batik yang dihasilkan pun terlihat beragam. Metode pembangkitan yang digunakan ialah metode *Iterated Function System* (IFS). Metode IFS dipilih karena mempunyai keunggulan, yaitu penulis dapat menentukan panjang segmen yang akan dibangkitkan serta terdapat transformasi Affine yang terdiri dari rotasi, dilasi dan translasi, berbeda dengan *L-system* yang hanya berpusat pada rotasi saja. Langkah-langkah penyusunan ornamen akan menghasilkan berbagai macam pilihan desain batik dari (m, n, c) yang bervariasi.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah yang dibahas pada tugas akhir ini adalah bagaimana langkah-langkah yang digunakan untuk mendapatkan desain batik dengan memanfaatkan geometri fraktal Koch *snowflake* (m, n, c) .

1.3 Batasan Masalah

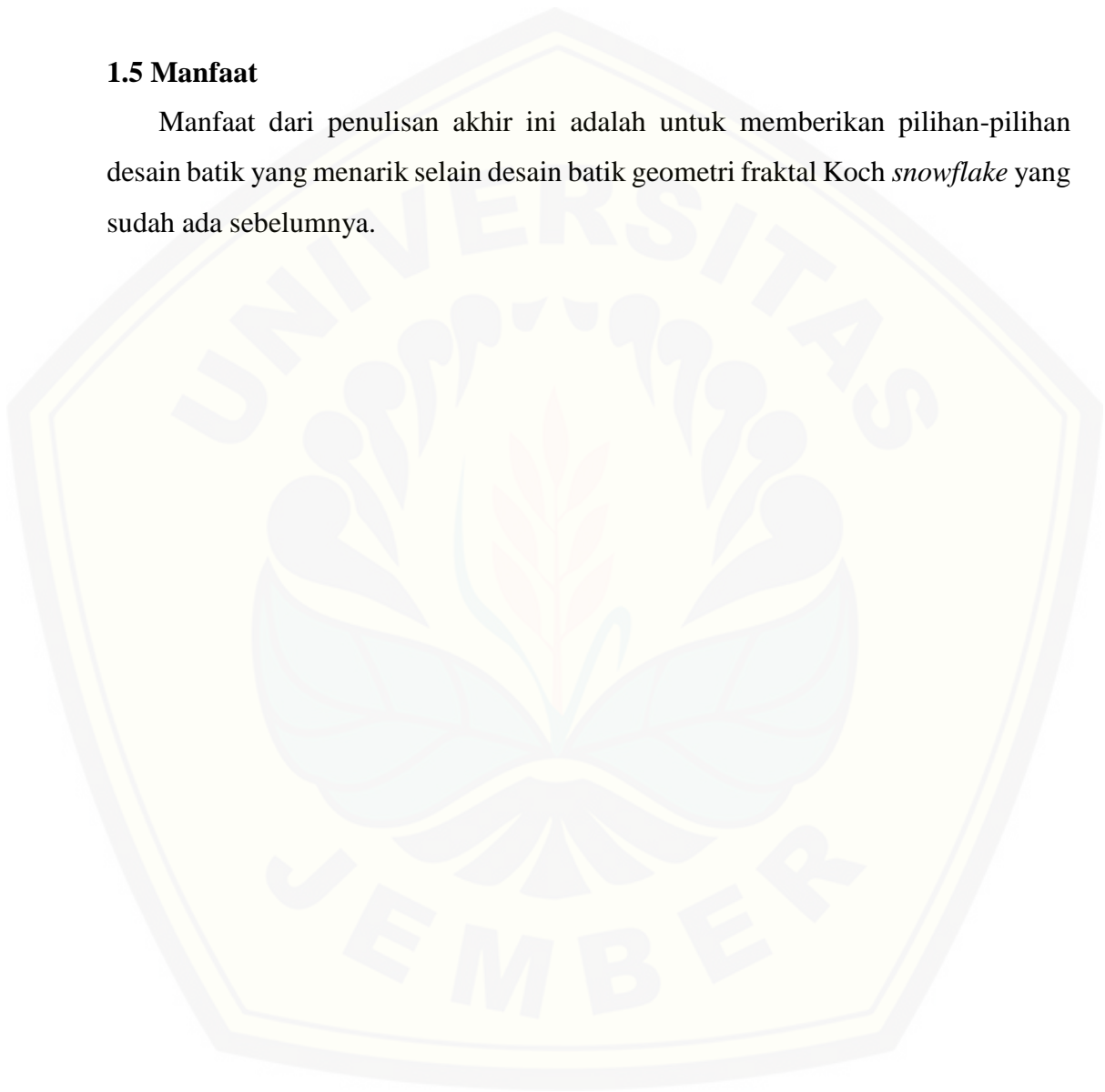
Batasan masalah yang digunakan pada penelitian ini adalah geometri fraktal Koch *snowflake* (m, n, c) dengan $3 \leq m \leq 6$, $3 \leq n \leq 6$ dan nilai c mengikuti sesuai aturan yang telah ditetapkan sehingga menghasilkan nilai estetika bervariasi. Metode pembangkitan yang digunakan adalah IFS dengan batasan 2 (iterasi). Pada penyusunan pola pada desain batik digunakan transformasi affine, yaitu translasi.

1.4 Tujuan

Tujuan dari penulisan tugas akhir ini adalah untuk membangun langkah-langkah penyusunan ornamen pada desain batik dari geometri fraktal Koch *snowflake* (m, n, c) ditinjau dari segi matematis.

1.5 Manfaat

Manfaat dari penulisan akhir ini adalah untuk memberikan pilihan-pilihan desain batik yang menarik selain desain batik geometri fraktal Koch *snowflake* yang sudah ada sebelumnya.



BAB 2. TINJAUAN PUSTAKA

2.1 Batik

Batik merupakan kain bergambar yang pembuatannya secara khusus dengan menuliskan atau menerapkan malam pada kain tersebut. Kain diproses dengan cara tertentu (KBBI.web.id, 2018). Secara etimologi kata batik berasal dari bahasa Jawa yaitu “amba” yang artinya menggambar dan “tik” yang artinya titik/nitik. Gambar pada mori (kain) disebut dengan motif/corak. Motif dibagi menjadi dua, yaitu ornamen dan *isen*. Ornamen adalah bagian utama atau motif yang menonjol dalam membuat gambar batik. *Isen* adalah bagian hiasan/pengisi ruang-ruang kosong. Motif secara umum adalah penyebaran garis atau warna dalam ulangan tertentu. Pengertian motif menjadi lebih sedikit kompleks antara lain hubungannya dengan pengertian simetrik. Motif tidak hanya diulang menurut garis secara paralel melainkan dicerminkan (Soedarso, 1941).

Ragam motif batik mulanya masih didominasi dengan binatang dan tanaman. Motif terdiri atas dasar bentuk/objek, skala /proporsi, dan komposisi. Motif menjadi pangkalan atau pokok dari sesuatu pola setelah motif itu mengalami proses penyusunan dan diterapkan secara berulang-ulang sehingga diperoleh sebuah motif yang akan diterapkan pada benda lain sehingga menghasilkan sebuah ornamen. Motif batik mengalami perkembangan, corak binatang dan tanaman beralih menjadi corak abstrak menyerupai awan, relief candi, bentuk geometris matematika dan sebagainya.

Terdapat dua jenis batik berdasarkan cara pembuatannya, yaitu batik tulis dan batik cap. Batik tulis dikerjakan secara manual menggunakan alat berupa canting dan kuas untuk menggambar motif. Awalnya motif dibuat menggunakan pensil dan ditebalkan menggunakan canting atau kuas yang telah diberi malam. Cairan malam akan meresap kedalam serat kain. Kain yang telah diberi malam kemudian diberi warna sesuai dengan keinginan. Batik cap proses pembuatannya lebih mudah karena menggunakan alat yang telah diukir motifnya lalu diberi warna dan ditempelkan pada kain seperti penggunaan stempel.

Berikut adalah motif yang akan digunakan pada penelitian ini.

1. Batik Motif Parang Rusak

Motif parang rusak memiliki arti pertarungan antara manusia melawan kejahatan dengan cara mengendalikan sifat-sifat dan keinginan mereka. Motif ini dipilih karena proporsi ornamennya seimbang antara ornamen satu dan lainnya. Motif batik parang rusak dapat dilihat pada Gambar 2.1.



Gambar 2.1 Batik Motif Parang Rusak

(Sumber: <https://infobatik.id/?s=motif+parang+rusak>)

2. Batik Motif Nitik

Motif nitik merupakan motif batik yang tertua. Motif batik ini terinspirasi oleh kain tenun patola yang dibawa oleh para pedagang dari Gujarat India yang datang ke Indonesia. Motif nitik dipilih karena susunan ornamen yang penuh dan rapat. Motif batik nitik dapat dilihat pada Gambar 2.2.



Gambar 2.2 Batik Motif Nitik

(Sumber: <https://infobatik.id/pola-batik-nitik/>)

(Djoemena, 1990).

2.2 Fraktal

Buku Benoit Mandelbrot yang berjudul *The Fractal Geometry of Nature* menjelaskan bahwa *Euclidean* digunakan untuk mempresentasikan bentuk-bentuk

yang dibuat manusia seperti bujursangkar, lingkaran, bola, segitiga, dan lain-lain. Fraktal merupakan cara alami mempresentasikan bentuk-bentuk objek di alam (Munir, 2004). Konsep dasar fraktal sebenarnya sudah dilakukan sebelum Mandelbrot. Ide dasar mengenai keserupaan diri ditulis oleh Leibniz, Kart, Lichtberg, Cantor dan Hausdorf (Everest, 1995).

Kata fraktal berasal dari bahasa latin *fractus* (memecahkan/menguraikan) dan kata kerja *frangere* (meretakkan atau membagi menjadi keping-kepingan). Menurut Mandelbrot, fraktal merupakan bentuk geometri yang tampak kasar atau berupa pecahan yang dapat dibagi lagi bagian-bagiannya, masing-masing bagian merupakan (setidak-tidaknya) bentuk tiruan yang diperbanyak dari skala keseluruhannya. Definisi fraktal secara matematik adalah sebagai himpunan titik-titik yang memiliki dimensi melebihi dimensi topologinya (Bourke, 1991). Geometri fraktal adalah cabang matematika yang mempelajari sifat-sifat dan perilaku fraktal.

Fraktal dibagi menjadi tiga macam menurut keserupaan diri, yaitu serupa diri secara persis, serupa diri sebagian, dan serupa diri statistik. Serupa diri secara persis artinya memiliki struktur fraktal yang sangat identik di segala skala, contohnya adalah kurva Koch *snowflake* dan Segitiga Sierpinski. Serupa diri sebagian maksudnya memiliki keserupaan diri yang tidak terlalu mirip jika skalanya diubah, contohnya adalah himpunan Mandelbrot. Serupa diri secara statistik maksudnya keserupaan diri bersifat statistik pada skala tertentu, jenis ini memiliki keserupaan diri yang paling lemah (Subiantoro, 2005).

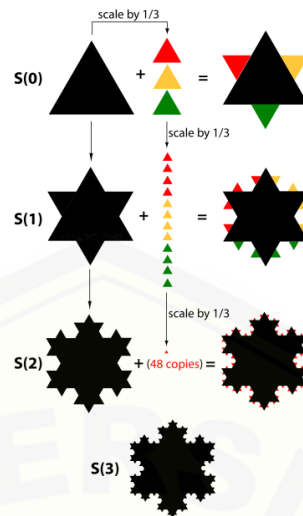
Sifat khas dari objek fraktal adalah setiap objek fraktal dapat dipecah menjadi bagian yang memiliki kesamaan bentuk dengan objek fraktal secara keseluruhan pada tingkat yang berbeda. Sifat yang memiliki kesamaan bentuk dengan objek aslinya disebut *similarity*. Menurut sifat *self-similarity*, fraktal dibagi menjadi dua macam, yaitu *regular fractal* dan *random fractal*. *Regular fractal* memiliki sifat *exactly self-similarity* yaitu setiap bagian dari objek fraktal menyerupai secara persis dengan bentuk objek secara keseluruhan jika dilihat dari berbagai skala. Pada salah satu objek yaitu segitiga Sierpinski, jika diambil sebagian dari dirinya kemudian diperbesar bagian potongan tertentu, maka akan terlihat kesamaan bentuk

yang menyerupai diri secara persis. *Random fractal* mempunyai sifat *statisically self-similarity* yaitu setiap bagian dari objek fraktal tidak menyerupai secara persis dengan bentuk objek secara keseluruhan. Contohnya adalah himpunan Julia, Mandelbrot, dan garis pantai (Addison, 1997).

2.3 Koch Snowflake

Koch *snowflake* merupakan sebuah fraktal atau dikenal juga dengan *the Koch island*, pertama kali dideskripsikan oleh Helge von Koch pada tahun 1904. Koch *snowflake* ini juga kadang dikaitkan dengan proses pembentukan *snowflake* yaitu *branching*. *Snowflake* yang merupakan fraktal adalah ketika *snowflake* terbentuk dari bentuknya sendiri atau disebut dengan *self-similar structure*. Proses ini merupakan sifat dimana sebuah cabang yang mempunyai subcabang, yang dimana tiap subcabang merupakan cabang yang diperkecil dan akan terus berlanjut sampai bentuk yang sangat kecil.

Langkah-langkah pembentukan kurva Koch *snowflake* dimulai dengan sebuah segitiga sama sisi $T = S(0)$ (inisiator). T diskalakan dengan faktor $\frac{1}{3}$ dan meletakkan 3 (tiga) salinan di sisi inisiator sehingga membentuk $S(1)$. Langkah berikutnya T diskalakan dengan faktor $\frac{1}{9} = \left(\frac{1}{3}\right)^2$ dan letakkan $12 = 4 \times 3$ salinan di sepanjang sisi $S(1)$ sehingga mendapatkan $S(2)$. Iterasi berikutnya, ambil $48 = 4 \times 12$ salinan T yang diskalakan dengan faktor $\frac{1}{27} = \left(\frac{1}{3}\right)^3$ dan letakkan disekitar sisi $S(2)$ sehingga membentuk $S(3)$. Proses pembangkitan Koch *snowflake* dapat dilihat pada Gambar 2.3.

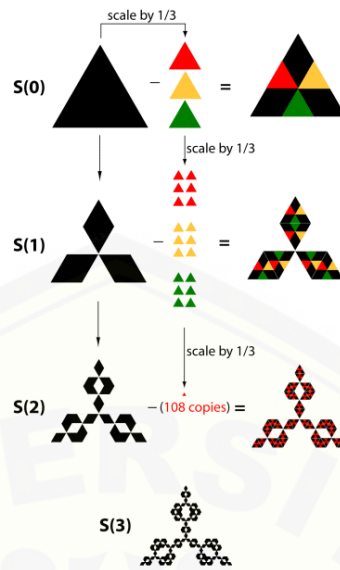


Gambar 2.3 Pola Pembangkitan Kurva Koch *Snowflake*
 (Sumber: <http://ecademy.agnesscott.edu/~lriddle/ifs/ksnow/ksnow.htm>)

Variasi kurva Koch *snowflake* dapat dilakukan dengan cara membangkitkan berbagai generator (hasil dari *production rule*) pada sembarang segmen garis pada sisi bidang inisiator (bentuk awal). Pengembangan variasi Koch *snowflake* diantaranya Koch *snowflake* luas bertambah, luas berkurang dan luas tetap (Kamil, 2004).

2.4 Koch anti-*Snowflake*

Koch anti-*snowflake* didapatkan dengan cara membangkitkan generator ke arah dalam atau ke pusat dari sisi-sisi inisiatornya. Langkah pertama pembuatan kurva Koch anti-*snowflake* dimulai dengan menggambar segitiga sama sisi $T = S(0)$. Generator kemudian dibangkitkan dengan menambahkan 3 (tiga) segitiga tanpa alas di setiap inisiator ke arah dalam T dan akan membentuk tiga bagian yang berbentuk berlian $S(1)$. Masing-masing berlian terdiri dari 2 (dua) salinan T dan diskalakan dengan faktor $\frac{1}{3}$. $S(2)$ didapatkan dengan mengulang langkah pertama pada masing-masing 6 segitiga. Terdapat 36 salinan T dimana masing-masing diskalakan dengan faktor $\frac{1}{9} = \left(\frac{1}{3}\right)^2$. Proses pembangkitan Koch anti-*snowflake* dapat dilihat pada Gambar 2.4.



Gambar 2.4 Pembangkitan Kurva Koch Anti-Snowflake
(Sumber: <http://ecademy.agnesscott.edu/~lriddle/ifs/ksnow/kantisnow.htm>)

Pada Koch *snowflake* (m, n, c) , m adalah jumlah sisi poligon, n adalah bentuk pembangkitan, biasanya sama dengan nilai n dan c adalah faktor yang digunakan untuk membagi segmen pada poligon segi- m . c adalah panjang segmen yang dihilangkan saat pembangkitan setiap iterasi dan terletak pada tengah segmen. Penentuan faktor c ini digunakan agar pada saat pembangkitan kurva Koch tidak terjadi tumpang tindih. Aturan penentuan fraktor c adalah sebagai berikut:

$$c < \frac{\sin^2\left(\frac{\pi}{n}\right)}{\cos^2\left(\frac{\pi}{n}\right) + 1}, \quad n \text{ genap} \quad (2.1)$$

$$c < 1 - \cos\left(\frac{\pi}{n}\right), \quad n \text{ ganjil}$$

dengan n adalah bilangan asli dan c adalah bilangan pecahan (Kaleti dan Paquatte, 2010).

2.5 Iterated Function System (IFS)

Iterated Function System (IFS) membangkitkan fraktal dengan cara mengulang transformasi berkali-kali untuk sebarang pola. Pola awal ditransformasikan menjadi suatu pola berulang dengan struktur yang sama pada detail tertentu, hal tersebut merupakan karakteristik dasar dari himpunan fraktal. Teori transformasi fraktal

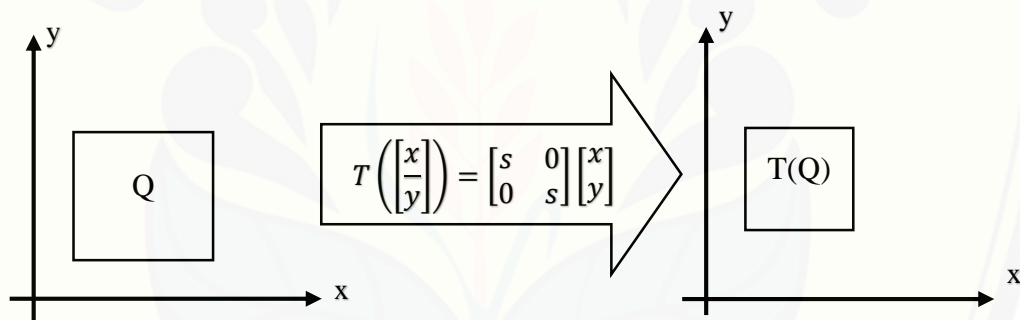
merupakan teori IFS lokal, dimana walaupun merupakan teori yang cukup rumit untuk menemukan fraktal tetapi dapat menyederhanakan proses (Utomo, 2011). *Iterated Function System* (IFS) merupakan suatu fungsi iterasi yang terdiri dari sekumpulan transformasi *affine* yang digunakan untuk membangun suatu objek fraktal. Transformasi *affine* adalah transformasi linier yang diikuti dengan dilasi, rotasi, dan translasi (Budhi, 1995).

Transformasi Affine adalah pemetaan R^2 menjadi R^2 dalam persamaan

$$T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} \quad (2.2)$$

dengan a, b, c, d, e dan f adalah besaran-besaran skala (Anton & Rorres, 2004).

Anton dan Rorres (2004) menyatakan jika $T: R^2 \rightarrow R^2$ adalah suatu operator linier yang mengubah skala dengan faktor s dan jika Q adalah sebuah himpunan titik didalam R^2 , maka $T(Q)$ disebut kontraksi dari Q jika $0 < s < 1$.



Gambar 2.5 Kontraksi dari Q

Menurut Kusno (2003) rotasi adalah suatu perpindahan benda pada gerakan melingkar. Pada dimensi dua, benda akan berputar pada pusat rotasi. Jika $T: R^2 \rightarrow R^2$ adalah suatu transformasi yang memetakan titik (x, y) ke titik (x', y') dan misalkan θ adalah sebuah sudut tetap maka didefinisikan sebagai:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Translasi merupakan transformasi yang memetakan titik (x, y) bergeser sejauh e satuan ke arah sumbu x dan f satuan ke arah sumbu y sehingga didapatkan persamaan:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix} = \begin{bmatrix} x + e \\ y + f \end{bmatrix}$$

Sebuah objek fraktal mempunyai sifat pemiripan (*similitude*). Sebuah pemiripan dengan faktor skala s merupakan sebuah pemetaan dari R^2 menjadi R^2 dalam bentuk

$$T\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$

dengan s, θ, e dan f adalah besaran-besaran skala. Pemiripan terdiri dari tiga pemetaan sederhana yang sudah diuraikan di atas, yaitu sebuah pengubahan skala dengan faktor s , sebuah rotasi terhadap titik asal sebesar sudut θ , dan sebuah translasi (e satuan dalam arah x dan f satuan dalam arah y).

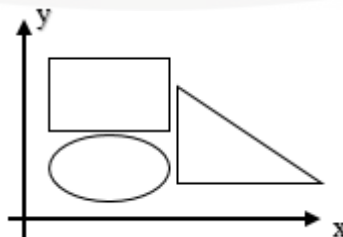


Gambar 2.6 Ilustrasi Efek dari Pemiripan pada suatu Bujursangkar Satuan U

Pemiripan merupakan hal yang penting dalam membangkitkan objek fraktal. Sebuah subhimpunan tertutup dan terbatas pada bidang euclidean R^2 dikatakan saling serupa (*self-similar*) jika dapat dinyatakan dalam bentuk

$$S = S_1 \cup S_2 \cup S_3 \cup \dots \cup S_k$$

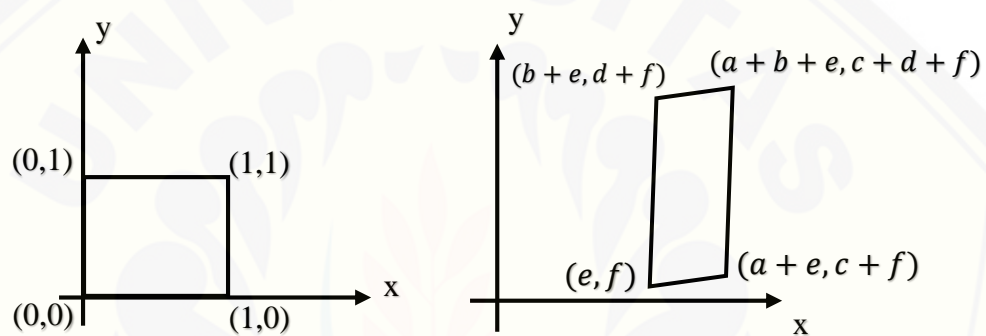
dengan $S = S_1, S_2, S_3, \dots, S_k$ adalah himpunan-himpunan yang tidak saling tumpang tindih, masing-masing kongruen dengan S yang diubah skalanya dengan faktor s yang sama ($0 < s < 1$). Dua himpunan di dalam R^2 disebut sama dan sebangun atau kongruen jika kedua himpunan ini dapat dibuat tepat saling berhimpit dengan mentranslasikan dan merotasikannya secara tepat di dalam R^2 .



Gambar 2.7 Himpunan-himpunan yang Tidak Saling Tumpang Tindih

Metode IFS merupakan sekumpulan transformasi Affine, untuk menghasilkan objek fraktal yang mempunyai pemiripan dibutuhkan pemiripan yang bersifat kontraksi. Sebuah transformasi dikatakan mengkontraksi apabila jarak Euclidean diantara dua titik sebarang pada suatu bidang berkurang setelah kedua titik tersebut dipetakan oleh transformasi tersebut. Sebarang k transformasi Affine yang mengkontraksi T_1, T_2, \dots, T_k dapat menentukan sebuah himpunan tertutup dan terbatas pada S , yang memenuhi Persamaan 2.3.

$$S = T_1(S) \cup T_2(S) \cup T_3(S) \dots \cup T_k(S) \quad (2.3)$$



(a) Sebelum Transformasi *Affine* (b) Sesudah Transformasi *Affine*

Gambar 2.10 Transformasi *Affine*

Persamaan (2.2) menggunakan transformasi Affine yang mengkontraksi sebuah himpunan saling serupa S , namun himpunan yang ditentukannya mempunyai sifat saling serupa (Anton dan Rorres, 2004).

2.5 GUI pada Matlab

Matlab (*Matrix Laboratory*) adalah sebuah lingkungan komputasi dan bahasa pemrograman yang dikembangkan oleh *Mathwork*. Matlab dikembangkan sebagai bahasa pemrograman sekaligus alat visualisasi untuk menyelesaikan berbagai kasus yang berhubungan dengan ilmu matematika.

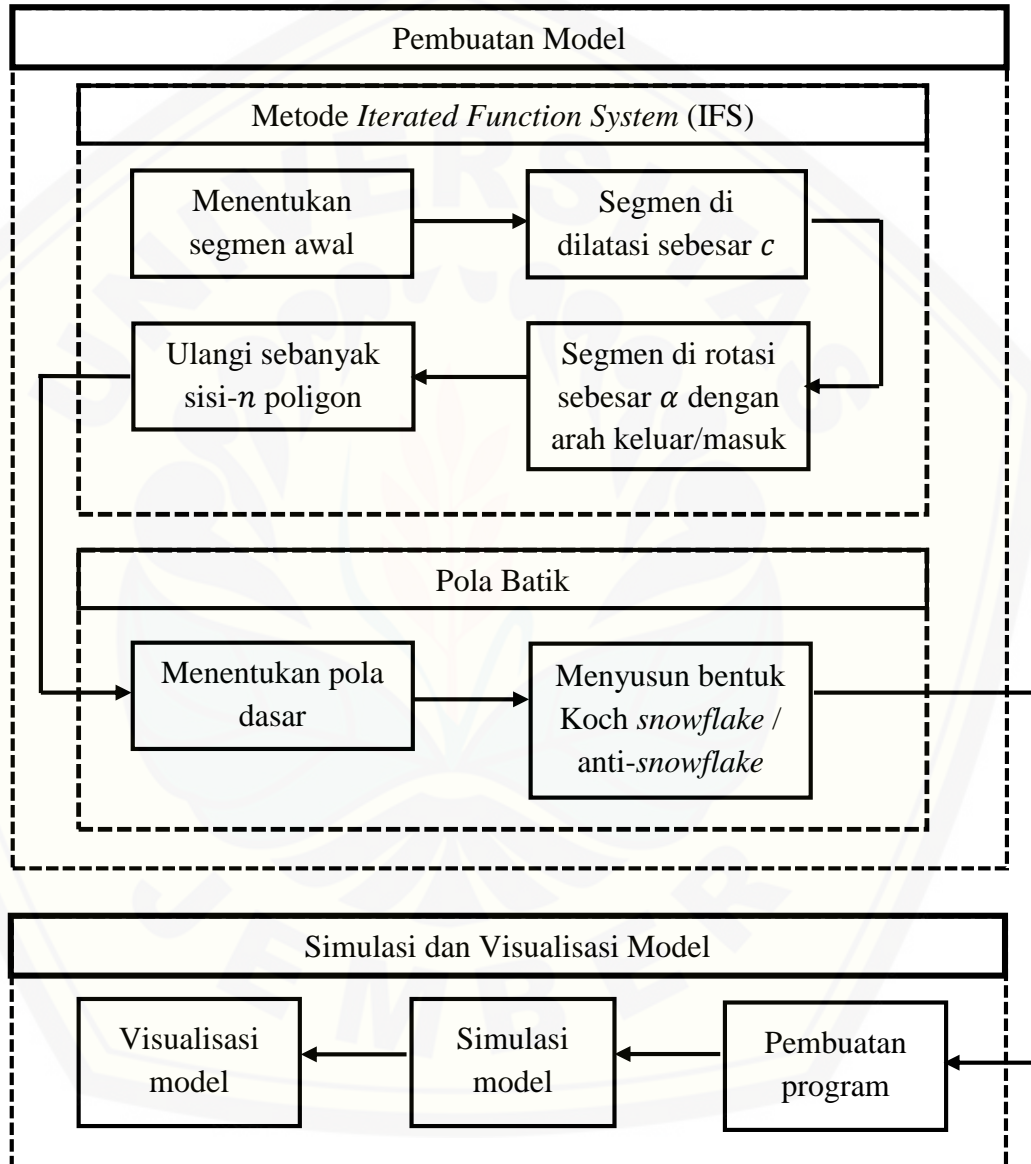
Matlab merintis kearah pemrograman yang menggunakan *Graphical User Interface* (GUI). *GUIDE* atau *GUI builder* merupakan sebuah GUI yang menyediakan media tampilan grafis sebagai pengganti perintah teks untuk berinteraksi antara user dengan program. Program yang dihasilkan dengan

menggunakan GUI jauh lebih menarik dan menjadi lebih interaktif serta penggunaan program menjadi efektif. GUIDE menghasilkan *m-file* yang berisi kode program untuk menangani inisialisasi dan menjalankan GUI. *m-file* kemudian menyediakan suatu kerangka untuk implementasi *callback*, yaitu fungsi yang bekerja ketika para pemakai mengaktifkan suatu komponen di dalam GUI. Matlab menyediakan keperluan untuk membuat program GUI, matlab menyediakan komponen standar, seperti *edit*, *text*, *pushbutton*, *frame*, *checkbox*, dan lain-lain (Sugiharto, 2006).



BAB 3. METODE PENELITIAN

Berdasarkan rumusan masalah, pada bab ini dijelaskan mengenai langkah-langkah yang dilakukan dalam membangkitkan Koch *snowflake* dan Koch anti-*snowflake* dengan menggunakan metode IFS.



Gambar 3.1 Skema Penelitian

Pada penelitian ini terdapat dua tahapan. Tahap pertama adalah pembuatan model. Pembuatan model terdiri dibagi menjadi dua tahap, yaitu metode IFS dan pola batik. Metode IFS menjelaskan bagaimana langkah-langkah pembangkitan

Koch *snowflake* (m, n, c) . Pola batik menjelaskan bagaimana desain batik dibuat berdasarkan pola penyusunan Koch *snowflake* (m, n, c) . Tahap kedua yaitu visualisasi dan simulasi model pada program yang akan dibuat.

3.1 Membangun Model IFS

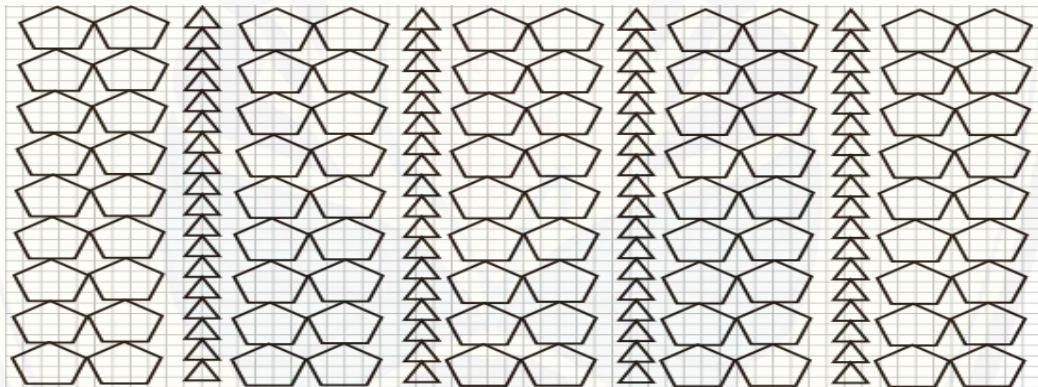
Langkah-langkah dalam membangkitkan Koch *snowflake* / anti-*snowflake* (m, n, c) dengan menggunakan metode IFS yaitu menentukan parameter atau besaran skala dari transformasi Affine dan selanjutnya perhitungan titik berikutnya. Berikut adalah langkah-langkah yang diperlukan:

- a. Menentukan bentuk poligon Koch *snowflake* / anti-*snowflake* (m, n, c)
Tentukan bentuk m -poligon yang akan menjadi bentuk dasar dan n yang akan menjadi bentuk yang akan dibangkitkan dari pembangkitan Koch *snowflake* atau anti-*snowflake* (m, n, c) dan c yang akan menjadi pembagi segmen pada sisi poligon. Setiap n -poligon akan mempunyai nilai c yang didapatkan dari perhitungan pada persamaan 2.1.
- b. Menentukan segmen awal
Tentukan segmen awal (L) dari sisi poligon segi- m yang akan dibangkitkan dan c yang akan digunakan.
- c. Dilasi segmen awal
 L didilasi sebesar a faktor sebagai segmen yang menyatakan dibangkitkannya kurva Koch sehingga menghasilkan \overline{AB} . \overline{AB} didilasi sebesar $\frac{c}{a}$ di titik B sehingga menghasilkan $\overline{A'B'}$.
- d. Hasil dilasi dirotasi di titik B dengan sudut $\alpha = 180^\circ - \theta$ searah jarum jam. $\overline{A'B'}$ dirotasi sebanyak $(n - 2)$ kali di titik A' dan B' sebesar $\theta = \frac{(n-2)}{2} \cdot 180^\circ$ berlawanan arah jarum jam. Translasi sebesar $\frac{1+c}{2}$ disepanjang \overline{AB} di titik terakhir.
- e. Lakukan pengulangan langkah b sampai d pada setiap sisi n -poligon.
- f. Pembangkitan pada iterasi selanjutnya L didilasi sebesar $\left(\frac{1-c}{2}\right)^2 L$ dan $c^2 L$ kemudian di translasi sebesar $\left(\frac{1+c}{2}\right)^2 L$.

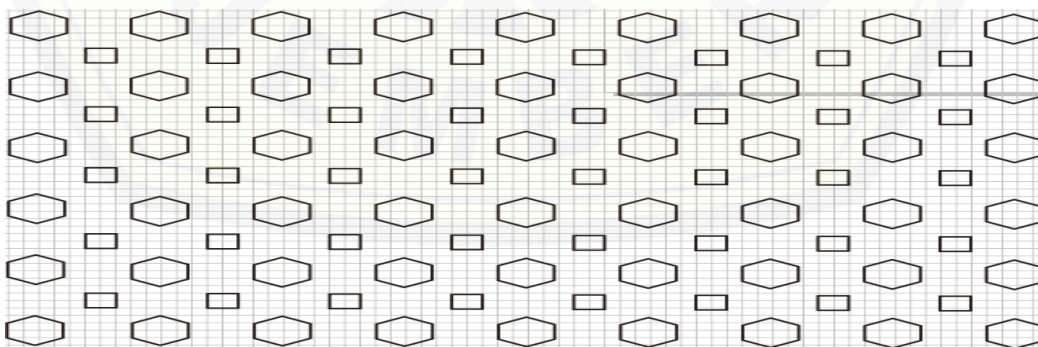
Pada Koch *snowflake* cL akan dibangkitkan ke arah luar poligon, sedangkan pada Koch *anti-snowflake* cL akan dibangkitkan ke arah dalam poligon. Total iterasi yang dilakukan adalah sebanyak 2 (kali) agar bentuk Koch *snowflake* atau *anti-snowflake* (m, n, c) terlihat jelas.

3.2 Pola Batik

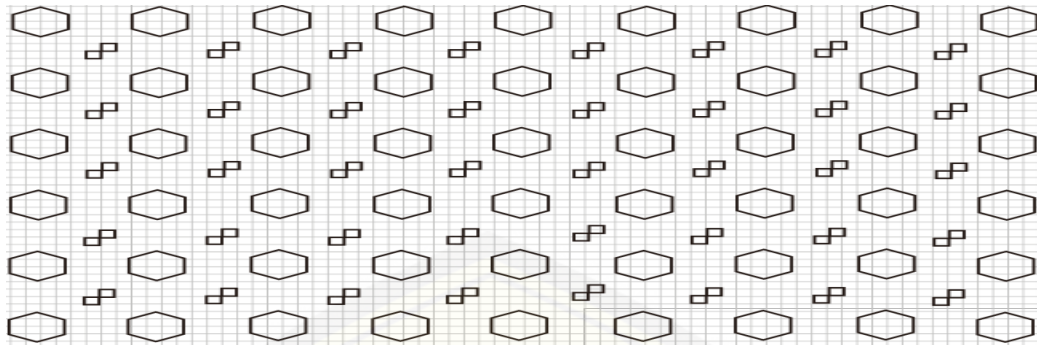
Pada pembuatan pola batik, terlebih dahulu tentukan pola dasar yang akan digunakan. Pola dasar ini berguna pada saat pembuatan program. Bentuk Koch *snowflake* atau *anti-snowflake* (m, n, c) akan disusun secara horizontal dan vertikal dengan memanfaatkan transformasi geometri seperti rotasi, translasi, refleksi, dan dilatasi serta aturan yang diterapkan pada masing-masing pola batik. Pola batik tersusun atas beberapa Koch *snowflake* atau *anti-snowflake* (m, n, c) dari berbagai poligon segi- m . Berikut pola batik yang akan digunakan.



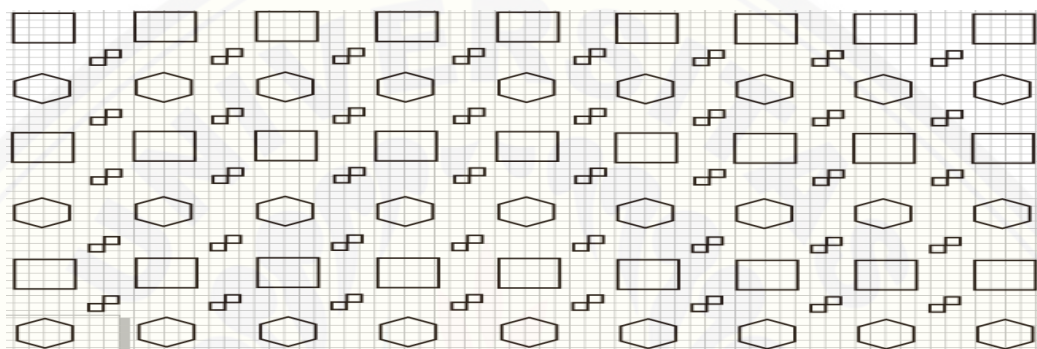
Gambar 3.2 Pola Dasar Batik 1



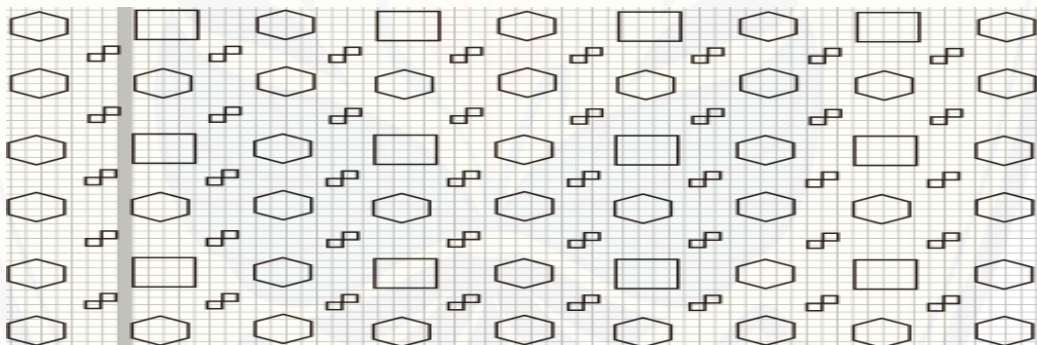
Gambar 3.3 Pola Dasar Batik 2



Gambar 3.4 Pola Dasar Batik 3



Gambar 3.5 Pola Dasar Batik 4



Gambar 3.6 Pola Dasar Batik 5

Poligon yang terdapat pada pola dasar batik dapat diganti sesuai poligon segi- m yang telah ditentukan. Pada setiap pola terdapat aturan-aturan untuk menentukan posisi atau besaran translasi ornamen. Titik tengah poligon segi- m sebagai titik awal dimulainya translasi.

3.3 Pembuatan Program

Program berfungsi sebagai alat untuk membantu dalam mengaplikasikan pola batik yang telah ditentukan. Program yang digunakan adalah *software* Matlab. Pada

Matlab akan dibuat GUI Matlab dimana akan disediakan *pop-up menu* pola batik, ornamen yang digunakan, dan warna yang diinginkan. *Click button* digunakan untuk membuat halaman baru, menggambar ornamen batik, menggambar batik, dan menyimpan batik yang telah selesai dibuat.



BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan penelitian yang telah dilakukan, didapatkan langkah-langkah pembangkitan dan penyusunan ornamen serta hasil desain batik yang beragam dari kombinasi 5 pola dan ornamen. Pembangkitan ornamen Koch *snowflake* dan Koch anti-*snowflake* dilakukan menggunakan metode IFS sebanyak 2 iterasi agar hasil dari pembangkitan masih terlihat jelas. Pembangkitan diawali dengan menentukan m, n, c kemudian ditentukan segmen awal, melakukan dilasi, rotasi dan translasi. Ornamen yang telah dibangkitkan lalu disusun dari arah kiri ke kanan dan atas ke bawah dengan ornamen skala besar terlebih dahulu. Ornamen disusun berdasarkan letak ornamen pada tiap-tiap pola menggunakan transformasi, yaitu translasi. Batik fraktal Koch *snowflake* dan Koch anti-*snowflake* menghasilkan sebanyak 64 kombinasi ornamen pada pola 1, pola 2 dan pola 3 serta 512 kombinasi ornamen pada pola 4 dan pola 5. Kombinasi meliputi bentuk-bentuk ornamen dan dipadukan dengan warna ornamen, warna latar, dan penambahan motif gajah oling, daun tembakau dan biji kopi. Hasil desain batik yang didapatkan mempunyai kemiripan penyusunan ornamen dengan penyusunan motif tradisional, yaitu penyusunan ornamen pada Pola 1 dengan motif parang rusak dan penyusunan ornamen pada Pola 5 dengan motif nitik.

5.2 Saran

Berdasarkan hasil akhir program yang telah dibuat, terdapat beberapa saran yang perlu diperhatikan.

- a. Jarak antar ornamen dibuat lebih rapat sehingga terlihat lebih penuh dan *space* kosong tidak terlihat jelas.
- b. Motif lokal yang digabungkan dengan desain batik ditempatkan berdampingan dengan ornamen-ornamen bukan tumpang tindih.
- c. Ukuran panjang dan lebar ornamen yang digunakan ditambah menjadi lebih dari 4 satuan sehingga ornamen terkecil dapat terlihat jelas.

DAFTAR PUSTAKA

- Addison, P.S. 1997. *Fractal and Chaos*. Bristol and Philadelphia: Institute of Phisic Publishing.
- Bourke, P. 1991. An Introduction to Fractals.
<http://astronomy.swin.edu.au/~pbourke/fractals/francito.html>. [30 Mei 2018]
- Anton, H dan Rorres, C. 2004. *Aljabar Linear Elementer versi Aplikasi Jilid 2*. Jakarta: Erlangga.
- Destiarmand, A. H., Hariadi, Y., Likman, M. 2010. Batik Fractal: Marriage of Art and Science. *ITB J. Vis. Art & Des.* 4(1): 84-94.
- Djoemena, N. 1990. *Ungkapan Sehelai Batik*. Jakarta: Penerbit Djambatan.
- Evertsz, C. J. G., Peitgen H. O., dan Voss R. F. 1995. *Fractal Geometry and Analysis*. Singapore: World Scientific.
- Kaleti, T and Paquatte, E. 2010. *The Trouble With Von Koch Curves Built From n Gons. The American Mathematical Monthly Vol 117 No 2*. America: Mathematical Association of America.
- Kamil, A. 2004. Penentuan Luas Fraktal Koch Snofwflake. *Skripsi*. Jember: FMIPA Universitas Jember.
- Kamsyakawuni, A., Purnomo, K. D., Wulandari. E.K. 2017. Pengembangan Desain Batik Labako dengan Menggabungkan Geometri Fraktal Kurva Naga dan Corak dari Daun Tembakau. *Jurnal Ilmu Dasar*. 18(2): 125-132.
- KBBI. 2018. Batik. <https://kbbi.web.id/batik>. [30 Mei 2018]
- Kudiya, H. K. 2009. Proses Pembuatan Batik Fraktal VS Batik Tradisional. <http://netsains.com/2009/10/proses-pembuatan-batik-fractal-vc-batik-tradisional/>. [17 Februari 2019]
- Kusno, 2003. *Geometri Rancang Bangun Studi Surfes Putar Transformasi Titik dan Proyeksi*. Jember: Fakultas MIPA Universitas Jember.

Munir, R. 2004. *Pengolahan Citra Digital dengan Pengolahan Algoritmik*. Edisi Pertama. Informatika Bandung.

Mursito, 2009. Batik Fraktal. computational.engineering.or.id/Batik_Fraktal [30 Mei 2018]

Riddle, L. 2017. Koch Anti-Snowflake.
<http://ecademy.agnesscott.edu/~lriddle/ifs/ksnow/kantisnow.htm>
[23 Februari 2019]

Riwansia R. R. 2016. Pengembangan Desain Batik Melalui Penggunaan Geometri Faktal Koch Snowflake. *Skripsi*. Jember: FMIPA Universitas Jember.

Santi, R.C. N. 2011. Motif Batik dengan Menggunakan Fraktal. *Jurnal Dinamika Informatika*. 3(2): 1.

Soedarso. 1941. *Seni Lukis Batik*. Yogyakarta: BP ISI.

Subiantoro, N. 2005. *Penentuan Dimensi Objek Fraktal dengan Metode Box Counting*. Skripsi. Jember: FMIPA Universitas Jember.

Sugiharto, A. 2006. *Pemograman GUI dengan Matlab*. Yogyakarta: C.V Andi Offset.

Utomo, B. 2011. Fraktal dan Invers Fraktal. *Jurnal Matematika* 2(1): 28-35.

LAMPIRAN

Lampiran 1. Script IFS Koch Snowflake

```
function KochP=IFS_Koch(Segment,n,s1,s2,theta,alpha,iter,KochP)
if iter==0
    KochP=[KochP Segment(:,2)];
else
    T=s1*eye(2)*(Segment-repmat(Segment(:,1),1,2))+repmat(Segment(:,1),1,2);
    Koch=T;
    T=s2*eye(2)*[cosd(alpha) -sind(alpha);sind(alpha) cosd(alpha)]*(T-
repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
    T=[T(:,2) T(:,1)];
    Koch=[Koch T(:,2)];
    for i=1:n-2
        T=[cosd(theta) sind(theta);-sind(theta) cosd(theta)]*(T-
repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
        T=[T(:,2) T(:,1)];
        Koch=[Koch T(:,2)];
    end
    T=1/s2*eye(2)*[cosd(alpha) -sind(alpha);sind(alpha) cosd(alpha)]*(T-
repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
    T=[T(:,2) T(:,1)];
    Koch=[Koch T(:,2)];
    for i=1:size(Koch,2)-1
        KochP=IFS_Koch(Koch(:,i+1),n,s1,s2,theta,alpha,iter-1,KochP);
    end
end
end
```

Lampiran 2. Script IFS Koch Anti-Snowflake

```
function KochP=IFS_KochAnti(Segment,n,s1,s2,theta,alpha,iter,KochP)
if iter==0
    KochP=[KochP Segment(:,2)];
else
    T=s1*eye(2)*(Segment-repmat(Segment(:,1),1,2))+repmat(Segment(:,1),1,2);
    Koch=T;
    T=s2*eye(2)*[cosd(alpha) sind(alpha);-sind(alpha) cosd(alpha)]*(T-
repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
    T=[T(:,2) T(:,1)];
    Koch=[Koch T(:,2)];
    for i=1:n-2
        T=[cosd(theta) -sind(theta);sind(theta) cosd(theta)]*(T-
repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
        T=[T(:,2) T(:,1)];
        Koch=[Koch T(:,2)];
    end
    T=1/s2*eye(2)*[cosd(alpha) sind(alpha);-sind(alpha) cosd(alpha)]*(T-
repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
```

```

T=[T(:,2) T(:,1)];
Koch=[Koch T(:,2)];
for i=1:size(Koch,2)-1
    KochP=IFS_KochAnti(Koch(:,i+1),n,s1,s2,theta,alpha,iter-1,KochP);
end
end
end

```

Lampiran 3. Script Pola Batik

```

function varargout = BatikKoch(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @BatikKoch_OpeningFcn, ...
    'gui_OutputFcn', @BatikKoch_OutputFcn, ...
    'gui_LayoutFcn', [] , ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function BatikKoch_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;

guidata(hObject, handles);
clc;
movegui(gcf,'center');
set(handles.popupmenu1,'value',1);
set(handles.popupmenu2,'value',1);
set(handles.popupmenu3,'value',1);
set(handles.popupmenu4,'value',1);
set(handles.popupmenu5,'value',1);
set(handles.popupmenu6,'value',1);
set(handles.popupmenu7,'value',1);
set(handles.popupmenu8,'value',1);
set(handles.popupmenu9,'value',1);
set(handles.popupmenu10,'value',1);
cla(handles.axes1,'reset');
set(handles.axes1,'XTick','','YTick','');
cla(handles.axes2,'reset');
set(handles.axes2,'XTick','','YTick','');
cla(handles.axes3,'reset');
set(handles.axes3,'XTick','','YTick','');
cla(handles.axes4,'reset');

```

```
set(handles.axes4,'XTick','','YTick','');
```

```
function varargout = BatikKoch_OutputFcn(hObject, eventdata, handles)  
varargout{1} = handles.output;
```

```
function pushbutton1_Callback(hObject, eventdata, handles)
```

```
clc;  
movegui(gcf,'center');  
set(handles.popupmenu1,'value',1);  
set(handles.popupmenu2,'value',1);  
set(handles.popupmenu3,'value',1);  
set(handles.popupmenu4,'value',1);  
set(handles.popupmenu5,'value',1);  
set(handles.popupmenu6,'value',1);  
set(handles.popupmenu7,'value',1);  
set(handles.popupmenu8,'value',1);  
set(handles.popupmenu9,'value',1);  
set(handles.popupmenu10,'value',1);  
cla(handles.axes1,'reset');  
set(handles.axes1,'XTick','','YTick','');  
cla(handles.axes2,'reset');  
set(handles.axes2,'XTick','','YTick','');  
cla(handles.axes3,'reset');  
set(handles.axes3,'XTick','','YTick','');  
cla(handles.axes4,'reset');  
set(handles.axes4,'XTick','','YTick','');
```

```
function pushbutton2_Callback(hObject, eventdata, handles)
```

```
clc;  
cla(handles.axes1,'reset');  
set(handles.axes1,'XTick','','YTick','');  
cla(handles.axes2,'reset');  
set(handles.axes2,'XTick','','YTick','');  
cla(handles.axes3,'reset');  
set(handles.axes3,'XTick','','YTick','');  
cla(handles.axes4,'reset');  
set(handles.axes4,'XTick','','YTick','');
```

```
%Ornamen 1
```

```
pm2=get(handles.popupmenu2,'value');  
color1=get(handles.popupmenu5,'value');  
if color1>9  
    clr1=rand(1,3);  
else  
    fixcolor=[0 0 0;.4 .4 .4;1 0 0;0 1 0;0 0 1;1 1 0;1 0 1;0 1 1;1 1 1];  
    clr1=fixcolor(color1,:);  
end  
set(handles.popupmenu5,'UserData',clr1);  
N1=mod(pm2-1,4)+3;  
c=[1/3, 0.3, 0.19, 0.14];
```

```

% Sudut
theta=(N1-2)*180/N1;
alpha=180-theta;
% Ukuran panjang dan lebar
s1=(1-c(N1-2))/2;
R=1/c(N1-2);
s2=2/(R-1);
% Poligon
T=[0 1;0 0];
Poly=T;
for i=1:N1-1
    T=[cosd(theta) sind(theta);-sind(theta) cosd(theta)]*(T-
    repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
    T=[T(:,2) T(:,1)];
    Poly=[Poly T(:,2)];
end
KochP=[0;0];
if ceil(pm2/4)==1
    for i=1:N1
        KochP=IFS_Koch(Poly(:,i:i+1),N1,s1,s2,theta,alpha,2,KochP);
    end
else
    for i=1:N1
        KochP=IFS_KochAnti(Poly(:,i:i+1),N1,s1,s2,theta,alpha,2,KochP);
    end
end
xmin=min(KochP(1,:));
xmax=max(KochP(1,:));
ymin=min(KochP(2,:));
ymax=max(KochP(2,:));
L1=xmax-xmin;
L2=ymax-ymin;
if L1>=L2
    sk=1/L1;
else
    sk=1/L2;
end
KochP=KochP-[repmat(xmin,1,length(KochP));repmat(ymin,1,length(KochP))];
KochP=sk*eye(2)*KochP;
KochP(:,end)=KochP(:,1);
axes(handles.axes1);
patch(KochP(1,:),KochP(2,:),clr1);
axis image
set(handles.axes1,'UserData',KochP,'XTick','','YTick','');

% Ornamen 2
pm3=get(handles.popupmenu3,'value');
color2=get(handles.popupmenu6,'value');
if color2>9
    clr2=rand(1,3);

```



```

else
    fixcolor=[0 0 0;.4 .4 .4;1 0 0;0 1 0;0 0 1;1 1 0;1 0 1;0 1 1;1 1 1];
    clr2=fixcolor(color2,:);
end
set(handles.popupmenu6,'UserData',clr2);
N2=mod(pm3-1,4)+3;
c=[1/3, 0.3, 0.19, 0.14];
% Sudut
theta=(N2-2)*180/N2;
alpha=180-theta;
% Ukuran panjang dan lebar
s1=(1-c(N2-2))/2;
R=1/c(N2-2);
s2=2/(R-1);
% Poligon
T=[0 1;0 0];
Poly=T;
for i=1:N2-1
    T=[cosd(theta) sind(theta);-sind(theta) cosd(theta)]*(T-
repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
    T=[T(:,2) T(:,1)];
    Poly=[Poly T(:,2)];
end
KochP=[0;0];
if ceil(pm3/4)==1
    for i=1:N2
        KochP=IFS_Koch(Poly(:,i:i+1),N2,s1,s2,theta,alpha,2,KochP);
    end
else
    for i=1:N2
        KochP=IFS_KochAnti(Poly(:,i:i+1),N2,s1,s2,theta,alpha,2,KochP);
    end
end
end
xmin=min(KochP(1,:));
xmax=max(KochP(1,:));
ymin=min(KochP(2,:));
ymax=max(KochP(2,:));
L1=xmax-xmin;
L2=ymax-ymin;
if L1>=L2
    sk=1/L1;
else
    sk=1/L2;
end
KochP=KochP-[repmat(xmin,1,length(KochP));repmat(ymin,1,length(KochP))];
KochP=sk*eye(2)*KochP;
KochP(:,end)=KochP(:,1);
axes(handles.axes2);
patch(KochP(1,:),KochP(2,:),clr2);
axis image

```

```

set(handles.axes2,'UserData',KochP,'XTick','YTick');

%Ornamen 3
pm4=get(handles.popupmenu4,'value');
color3=get(handles.popupmenu7,'value');
if color3>9
    clr3=rand(1,3);
else
    fixcolor=[0 0 0;.4 .4 .4;1 0 0;0 1 0;0 0 1;1 1 0;1 0 1;0 1 1;1 1 1];
    clr3=fixcolor(color3,:);
end
set(handles.popupmenu7,'UserData',clr3);
N3=mod(pm4-1,4)+3;
c=[1/3, 0.3, 0.19, 0.14];
%Sudut
theta=(N3-2)*180/N3;
alpha=180-theta;
%Ukuran panjang dan lebar
s1=(1-c(N3-2))/2;
R=1/c(N3-2);
s2=2/(R-1);
%Poligon
T=[0 1;0 0];
Poly=T;
for i=1:N3-1
    T=[cosd(theta) sind(theta);-sind(theta) cosd(theta)]*(T-
repmat(T(:,2),1,2))+repmat(T(:,2),1,2);
    T=[T(:,2) T(:,1)];
    Poly=[Poly T(:,2)];
end
KochP=[0;0];
if ceil(pm4/4)==1
    for i=1:N3
        KochP=IFS_Koch(Poly(:,i:i+1),N3,s1,s2,theta,alpha,2,KochP);
    end
else
    for i=1:N3
        KochP=IFS_KochAnti(Poly(:,i:i+1),N3,s1,s2,theta,alpha,2,KochP);
    end
end
xmin=min(KochP(1,:));
xmax=max(KochP(1,:));
ymin=min(KochP(2,:));
ymax=max(KochP(2,:));
L1=xmax-xmin;
L2=ymax-ymin;
if L1>=L2
    sk=1/L1;
else
    sk=1/L2;

```

```

end
KochP=KochP-[ repmat(xmin,1,length(KochP)); repmat(ymin,1,length(KochP))];
KochP=sk*eye(2)*KochP;
KochP(:,end)=KochP(:,1);
axes(handles.axes3);
patch(KochP(1,:),KochP(2,:),clr3);
axis image
set(handles.axes3,'UserData',KochP,'XTick','','YTick','');

function pushbutton3_Callback(hObject, eventdata, handles)
clc;
cla(handles.axes4,'reset');
set(handles.axes4,'XTick','','YTick','');

pm1=get(handles.popupmenu1,'value');
clr1=get(handles.popupmenu5,'UserData');
clr2=get(handles.popupmenu6,'UserData');
clr3=get(handles.popupmenu7,'UserData');

axes(handles.axes4);
pm9=get(handles.popupmenu9,'value');
bgcolor=[1 1 1;0 0 0;1 .22 .33;5 0 0;.2 1 .2;0 .3 0;.3 .8 1;0 0 .3;.75 0 .75];
if pm9>9
    clr9=rand(1,3);
else
    clr9=bgcolor(pm9,:);
end
set(handles.axes4,'color',clr9);
if pm1==1
    Orna1=get(handles.axes1,'UserData');
    Orna2=get(handles.axes2,'UserData');
    for i=1:9
        for j=1:10
            if mod(j,2)==1
                t1=13*(j-1)/2;
                t2=4.5*(i-1);

KochP=4*eye(2)*Orna1+[ repmat(t1,1,size(Orna1,2)); repmat(t2,1,size(Orna1,2))];
            else
                t1=13*(j-2)/2+4.5;
                t2=4.5*(i-1);

KochP=4*eye(2)*Orna1+[ repmat(t1,1,size(Orna1,2)); repmat(t2,1,size(Orna1,2))];
            end
            patch(KochP(1,:),KochP(2,:),clr1);
            axis image
            set(handles.axes4,'XTick','','YTick','');
            pause(0.01);
        end
    end
end
end

```

```

for i=1:20
    for j=1:4
        t1=13*(j-1)+9.75;
        t2=2*(i-1);
        KochP=2*eye(2)*Orna2+[ repmat(t1,1,size(Orna2,2)); repmat(t2,1,size(Orna2,2))];
        patch(KochP(1,:),KochP(2,:),clr2);
        axis image
        set(handles.axes4,'XTick','YTick','');
        pause(0.01);
    end
end
elseif pm1==2
    Orna1=get(handles.axes1,'UserData');
    Orna2=get(handles.axes2,'UserData');
    for i=1:6
        for j=1:9
            t1=7*(j-1);
            t2=7*(i-1);
            KochP=4*eye(2)*Orna1+[ repmat(t1,1,size(Orna1,2)); repmat(t2,1,size(Orna1,2))];
            patch(KochP(1,:),KochP(2,:),clr1);
            axis image
            set(handles.axes4,'XTick','YTick','');
            pause(0.01);
        end
    end
    for i=1:5
        for j=1:8
            t1=7*(j-1)+4.5;
            t2=7*(i-1)+4.5;
            KochP=2*eye(2)*Orna2+[ repmat(t1,1,size(Orna2,2)); repmat(t2,1,size(Orna2,2))];
            patch(KochP(1,:),KochP(2,:),clr2);
            axis image
            set(handles.axes4,'XTick','YTick','');
            pause(0.01);
        end
    end
elseif pm1==3
    Orna1=get(handles.axes1,'UserData');
    Orna2=get(handles.axes2,'UserData');
    for i=1:6
        for j=1:9
            t1=7*(j-1);
            t2=7*(i-1);
            KochP=4*eye(2)*Orna1+[ repmat(t1,1,size(Orna1,2)); repmat(t2,1,size(Orna1,2))];
            patch(KochP(1,:),KochP(2,:),clr1);
            axis image
            set(handles.axes4,'XTick','YTick','');
            pause(0.01);
        end
    end
end
end

```

```

for i=1:10
    for j=1:16
        if mod(j,2)==1 && mod(i,2)==1
            t1=7*(j-1)/2+4.5;
            t2=7*(i-1)/2+4.5;

KochP=1*eye(2)*Orna2+[ repmat(t1,1,size(Orna2,2)); repmat(t2,1,size(Orna2,2))];
            elseif mod(j,2)==0 && mod(i,2)==0
                t1=7*(j-2)/2+5.5;
                t2=7*(i-2)/2+5.5;

KochP=1*eye(2)*Orna2+[ repmat(t1,1,size(Orna2,2)); repmat(t2,1,size(Orna2,2))];
            end
            patch(KochP(1,:),KochP(2,:),clr2);
            axis image
            set(handles.axes4,'XTick','YTick','');
            pause(0.01);
        end
    end
    elseif pm1==4
        Orna1=get(handles.axes1,'UserData');
        Orna2=get(handles.axes2,'UserData');
        Orna3=get(handles.axes3,'UserData');
        for i=1:6
            for j=1:9
                t1=7*(j-1);
                t2=7*(i-1);
                if mod(j,2)==1 && mod(i,2)==1

KochP=4*eye(2)*Orna1+[ repmat(t1,1,size(Orna1,2)); repmat(t2,1,size(Orna1,2))];
                    patch(KochP(1,:),KochP(2,:),clr1);
                    elseif mod(j,2)==0 && mod(i,2)==0

KochP=4*eye(2)*Orna1+[ repmat(t1,1,size(Orna1,2)); repmat(t2,1,size(Orna1,2))];
                    patch(KochP(1,:),KochP(2,:),clr1);
                    else

KochP=4*eye(2)*Orna2+[ repmat(t1,1,size(Orna2,2)); repmat(t2,1,size(Orna2,2))];
                    patch(KochP(1,:),KochP(2,:),clr2);
                    end
                    axis image
                    set(handles.axes4,'XTick','YTick','');
                    pause(0.01);
                end
            end
        end
        for i=1:10
            for j=1:16
                if mod(j,2)==1 && mod(i,2)==1
                    t1=7*(j-1)/2+4.5;
                    t2=7*(i-1)/2+4.5;

```



```

        axis image
        set(handles.axes4,'XTick','YTick');
        pause(0.01);
    end
end
end
pm8=get(handles.popupmenu8,'value');
GajahOling=importdata('GajahOling.mat');
Bako=importdata('Bako.mat');
color4=get(handles.popupmenu10,'value');
if color4>9
    clr4=rand(1,3);
else
    fixcolor=[0 0 0;.4 .4 .4;1 0 0;0 1 0;0 0 1;1 1 0;1 0 1;0 1 1;1 1 1];
    clr4=fixcolor(color4,:);
end
if pm8==2
    for i=1:2
        GajahOling1=18*eye(2)*GajahOling+[repmat(28*(i-
1)+16,1,size(GajahOling,2));repmat(5.5,1,size(GajahOling,2))];
        patch(GajahOling1(1,:),GajahOling1(2,:),clr4);
        axis image
        set(handles.axes4,'XTick','YTick');
        pause(0.01);
    end
elseif pm8==3
    for i=1:4
        GajahOling1=10*eye(2)*GajahOling+[repmat(14*(i-
1)+9,1,size(GajahOling,2));repmat(mod(i-1,2)*14+5.5,1,size(GajahOling,2))];
        patch(GajahOling1(1,:),GajahOling1(2,:),clr4);
        axis image
        set(handles.axes4,'XTick','YTick');
        pause(0.01);
    end
elseif pm8==4
    for i=1:2
        GajahOling1=8*eye(2)*GajahOling+[repmat(28*(i-
1)+16,1,size(GajahOling,2));repmat(19.5,1,size(GajahOling,2))];
        tumpu=repmat(GajahOling1(:,1),1,size(GajahOling1,2));
        for j=1:4
            GajahOling2=[cosd(90*(j-1)) sind(90*(j-1));-sind(90*(j-1)) cosd(90*(j-
1))]*(GajahOling1-tumpu)+tumpu;
            patch(GajahOling2(1,:),GajahOling2(2,:),clr4);
            axis image
            set(handles.axes4,'XTick','YTick');
            pause(0.01);
        end
    end
elseif pm8==5
    Kopi=importdata('Kopi.mat');

```

```

for i=1:4
    if mod(i,2)==1
        Kopi1=4*eye(2)*Kopi+[repmat(14*(i-
1)+9,1,size(Kopi,2));repmat(14,1,size(Kopi,2))];
    else
        Kopi1=4*eye(2)*Kopi+[repmat(14*(i-
1)+9,1,size(Kopi,2));repmat(28,1,size(Kopi,2))];
    end
    tumpu=repmat(Kopi1(:,1)+[0;-1.5],1,size(Kopi1,2));
    for j=1:4
        Kopi2=[cosd(90*(j-1)) sind(90*(j-1));-sind(90*(j-1)) cosd(90*(j-1))]*(Kopi1-
tumpu)+tumpu;
        patch(Kopi2(1,:),Kopi2(2,:),clr4);
        axis image
        set(handles.axes4,'XTick','','YTick','');
        pause(0.01);
    end
end
elseif pm8==6
    BGDaun=importdata('BGDaun.mat');
    for i=1:2
        Bako1=6*eye(2)*Bako+[repmat(28*(i-
1)+16,1,size(Bako,2));repmat(21,1,size(Bako,2))];
        BGDaun1=6*eye(2)*BGDaun+[repmat(28*(i-
1)+16,1,size(BGDaun,2));repmat(19.5,1,size(BGDaun,2))];
        tumpu=repmat(Bako1(:,1)+[0;-1.5],1,size(Bako1,2));
        tumpu2=repmat(BGDaun1(:,1)+[0;-1.5],1,size(BGDaun1,2));
        for j=1:4
            BGDaun2=[cosd(90*(j-1)) sind(90*(j-1));-sind(90*(j-1)) cosd(90*(j-
1))]*(BGDaun1-tumpu2)+tumpu2;
            if color4~=9
                patch(BGDaun2(1,:),BGDaun2(2:,:),'w','EdgeColor','w');
            else
                patch(BGDaun2(1,:),BGDaun2(2:,:),'k','EdgeColor','k');
            end
            axis image
            set(handles.axes4,'XTick','','YTick','');
        end
        for j=1:4
            Bako2=[cosd(90*(j-1)) sind(90*(j-1));-sind(90*(j-1)) cosd(90*(j-1))]*(Bako1-
tumpu)+tumpu;
            patch(Bako2(1,:),Bako2(2,:),clr4);
            axis image
            set(handles.axes4,'XTick','','YTick','');
            pause(0.01);
        end
    end
end
end
end

```

```
function pushbutton4_Callback(hObject, eventdata, handles)
```



```
clc;
frame=getframe(handles.axes4);
image=frame2im(frame);
[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
    imwrite(image,fullfile(name_path,name_file));
end

function popupmenu1_Callback(hObject, eventdata, handles)
function popupmenu1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function popupmenu2_Callback(hObject, eventdata, handles)
function popupmenu2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function popupmenu3_Callback(hObject, eventdata, handles)
function popupmenu3_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function popupmenu4_Callback(hObject, eventdata, handles)
function popupmenu4_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function popupmenu5_Callback(hObject, eventdata, handles)
function popupmenu5_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function popupmenu6_Callback(hObject, eventdata, handles)
```

```
function popupmenu6_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function popupmenu7_Callback(hObject, eventdata, handles)
function popupmenu7_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function popupmenu8_Callback(hObject, eventdata, handles)
function popupmenu8_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function popupmenu9_Callback(hObject, eventdata, handles)
function popupmenu9_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function popupmenu10_Callback(hObject, eventdata, handles)
function popupmenu10_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

Lampiran 4. Titik koordinat ornamen Pola 1

0,36	4.5,36	9.75,38	17.5,36	26,36	30,38	39,36	43.5,36	50.25,38	56.5,36	65,36	70.5,38	78,36	82.5,36
		9.75,36			30,36			50.25,36			70.5,36		
		9.75,34			30,34			50.25,34			70.5,34		
0,31.5	4.5,31.5	9.75,32	17.5,31.5	26,31.5	30,32	39,31.5	43.5,31.5	50.25,32	56.5,31.5	65,31.5	70.5,32	78,31.5	82.5,31.5
		9.75,30			30,30			50.25,30			70.5,30		
0,27	4.5,27	9.75,28	17.5,27	26,27	30,28	39,27	43.5,27	50.25,28	56.5,27	65,27	70.5,28	78,27	82.5,27
		9.75,26			30,26			50.25,26			70.5,26		
0,22.5	4.5,22.5	9.75,24	17.5,22.5	26,22.5	30,24	39,22.5	43.5,22.5	50.25,24	56.5,22.5	65,22.5	70.5,24	78,22.5	82.5,22.5
		9.75,22			30,22			50.25,22			70.5,22		
0,18	4.5,18	9.75,20	17.5,18	26,18	30,20	39,18	43.5,18	50.25,20	56.5,18	65,18	70.5,20	78,18	82.5,18
		9.75,18			30,18			50.25,18			70.5,18		
		9.75,16			30,16			50.25,16			70.5,16		
0,13.5	4.5,13.5	9.75,14	17.5,13.5	26,13.5	30,14	39,13.5	43.5,13.5	50.25,14	56.5,13.5	65,13.5	70.5,14	78,13.5	82.5,13.5
		9.75,12			30,12			50.25,12			70.5,12		
0,9	4.5,9	9.75,10	17.5,9	26,9	30,10	39,9	43.5,9	50.25,10	56.5,9	65,9	70.5,10	78,9	82.5,9
		9.75,8			30,8			50.25,8			70.5,8		
0,4.5	4.5,4.5	9.75,6	17.5,4.5	26,4.5	30,6	39,4.5	43.5,4.5	50.25,6	56.5,4.5	65,4.5	70.5,6	78,4.5	82.5,4.5
		9.75,4			30,4			50.25,4			70.5,4		
0,0	4.5,0	9.75,2	17.5,0	26,0	30,2	39,0	43.5,0	50.25,2	56.5,0	65,0	70.5,2	78,0	82.5,0
		9.75,0			30,0			50.25,0			70.5,0		

Lampiran 5. Titik koordinat ornamen Pola 2

0,70		14,70		28,70		42,70		56,70		70,70		84,70		98,70		112,70
	11,63		25.5,63		39.5,63		53.5,63		67.5,63		81.5,63		95.5,63		109.5,63	
0,56		14,56		28,56		42,56		56,56		70,56		84,56		98,56		112,56
	11,49		25.5,49		39.5,49		53.5,49		67.5,49		81.5,49		95.5,49		109.5,49	
0,42		14,42		28,42		42,42		56,42		70,42		84,42		98,42		112,42
	11,35		25.5,35		39.5,35		53.5,35		67.5,35		81.5,35		95.5,35		109.5,35	
0,28		14,28		28,28		42,28		56,28		70,28		84,28		98,28		112,28
	11,21		25.5,21		39.5,21		53.5,21		67.5,21		81.5,21		95.5,21		109.5,21	
0,14		14,14		28,14		42,14		56,14		70,14		84,14		98,14		112,14
	11,7		25.5,7		39.5,7		53.5,7		67.5,7		81.5,7		95.5,7		109.5,7	
0,0		14,0		28,0		42,0		56,0		70,0		84,0		98,0		112,0

