



Generative Art sebagai Pembentuk Motif Geometris

Ariesa Pandanwangi
Agus Priyono
Ratnadewi
Universitas Kristen Maranatha
Pos-el: ariesa.pandanwangi@maranatha.edu

DOI: 10.32884/ideas.v8i4.959

Abstrak

Rancang motif masih banyak yang dilakukan dengan manual, menggunakan pensil kemudian di gambar di atas kertas. Adapun tujuan penelitian ini untuk merancang motif geometris yaitu dengan menggunakan *algorithmic art* yang diproduksi melalui teknologi komputer. Metode penelitian adalah deskriptif kualitatif. Peneliti mencari dan mempelajari perangkat-perangkat lunak yang dapat menghasilkan motif geometris secara digital melalui komputer. Perangkat yang digunakan pada penelitian ini adalah perangkat lunak *processing* dengan *algorithmic art*. Motif geometris yang dirancang menggunakan *algorithmic art*. *Algorithmic Art* (seni algoritmik) adalah bagian dari seni generatif yang merupakan hasil dari proses algoritmik dan dirancang oleh seniman serta biasanya menggunakan proses acak untuk menghasilkan variasi berdasarkan input eksternal. Hasil luaran program dengan perangkat lunak *processing* adalah gambar-gambar dengan bentuk geometris. Motif geometris dapat diterapkan pada kain, dekorasi, dan lain-lain.

Kata Kunci

Algorithmic art, generative art, geometris, pembentuk motif, teknologi komputer

Abstract

Designing motifs is still mostly done manually, using a pencil and then drawing on paper. The purpose of this research is to offer a solution in making geometric motifs by utilizing algorithmic art produced through computer technology. This method is a combination of the use of algorithmic art and the design used in making geometric motif designs. Algorithmic Art is a subset of generative art that is the result of an algorithmic process and is designed by the artist and usually uses a random process to generate variations based on external input. The results of this study are geometric motifs using the algorithmic art method. Geometric motifs can be applied to fabrics, decorations, and others.

Keywords

Algorithmic art, computer technology, design.

Pendahuluan

Algorithmic Art (seni algoritmik yang juga dikenal sebagai seni yang dihasilkan oleh komputer) adalah bagian dari seni generatif yang merupakan hasil dari proses algoritmik dan dirancang oleh seniman serta biasanya menggunakan proses acak untuk menghasilkan variasi berdasarkan input eksternal (Gherman, 2018). Misalnya suatu algoritma dianalogikan sebagai resep yang rumit dan inputnya sebagai berbagai macam bahan, maka pada bidang seni akan dapat menghasilkan jumlah hasil yang tak terbatas dengan menggunakan bahan yang berbeda berdasarkan resep aslinya. Input ini dapat berupa generator angka acak atau sumber data lain seperti bingkai (*frame*) dari film.

Algorithmic Art (seni algoritmik) diproduksi dengan bantuan komputer dengan memprogramnya untuk mengikuti beberapa prosedur yang menghasilkan objek seni (Ceric, 2008; Gherman, 2018). Sebagian besar karya awal semacam ini menghasilkan gambar dieksekusi pada *plotter* grafik atau digambar oleh manusia. Namun, proses seperti itu sering dikaitkan dengan seni berbasis waktu yaitu hasil gambar dilihat sebagai proyeksi dari waktu ke waktu oleh penonton.

Seni generatif diciptakan oleh sistem yang beroperasi secara mandiri, atau semi-otonom, bukan langsung oleh artis (Tempel, 2017). Seniman menciptakan sistem dan menetapkan parameter yang mempengaruhi hasil,

tetapi hasil itu sendiri muncul dari sistem imajinasi dan proses kreatif seniman (Pandanwangi, 2020; Suseno, 2014; Tabrani, 2014). Selanjutnya Dehlinger menyatakan bahwa sistem seni generatif kerap memanfaatkan program atau aplikasi yang dapat dengan mudah ditemui di berbagai media teknologi (Dehlinger, 2020). Peneliti lain (Gherman, 2018) yang juga membahas *Algorithmic Art* menyampaikan bahwa praktik artistik juga dapat dihasilkan dari karya yang berbasis non-komputer. Gherman, berupaya untuk melacak keterkaitan antara seni algoritmik dengan gerakan seni rupa modern di barat yang ditelusuri melalui pendekatan sejarah seni. Ia membandingkan prinsip estetika dan teknik yang digunakan oleh seniman. Hasil dari penelitiannya ini menyampaikan bahwa perlu kesadaran untuk mempertimbangkan *Algorithmic Art* yang lebih luas lagi untuk melihat hubungannya dengan bentuk seni non-digital.

Menurut studi yang dilakukan oleh (Monro, 2009), dituliskan bahwa seni generatif adalah setiap praktik seni di mana seniman menciptakan proses yang pada gilirannya bertindak dengan beberapa derajat otonomi untuk menciptakan semua atau sebagian dari sebuah karya seni. Hal ini yang sangat menarik karena mewakili interaksi antara seni dan sains, karena seniman tidak hanya menggunakan sarana teknologi (biasanya program komputer), tetapi juga seni itu sendiri sering berhubungan dengan pertanyaan mendasar tentang kehidupan dan apa artinya menjadi manusia dalam menghadapi kemajuan teknis yang pesat. Memang banyak seni generatif terkait erat dengan disiplin ilmu yang disebut kehidupan buatan, yang menanyakan pertanyaan serupa dari sudut pandang ilmiah. Sebuah konsep yang mendasari banyak seni generatif, dan ilmu kehidupan buatan, adalah konsep tentang munculnya (*emergence*). Secara umum, sebuah sistem menunjukkan perilaku yang muncul (*emergent*), jika sesuatu yang ekstra terjadi; dalam arti tertentu lebih banyak yang keluar daripada yang dimasukkan. Konsep kemunculan (*emergence*) telah terhubung ke beberapa disiplin ilmu, dan tidak memiliki definisi yang jelas; ternyata ada beberapa konsep dengan nama yang sama. Studi yang dilakukan bertujuan untuk mengungkap definisi kemunculan (*emergence*) yang sesuai dengan studi tentang seni generatif. Ketika komputer menjadi lebih cepat dan pekerjaan *real-time* menjadi lebih praktis, maka fokusnya mulai beralih dari produk ke proses. Sehingga memungkinkan untuk membuat karya seni yang berubah terhadap waktu, atau yang menanggapi masukan dari pengunjung. Istilah seni generatif tampaknya baru digunakan sekitar tahun 1990, sebagai konsekuensi dari kemungkinan yang diperluas ini, maka dalam diskusi informal seni generatif dapat membawa serta rasa ketidakpastian, atau perilaku *real-time*, atau keduanya.

Beberapa alternatif kakas (*tools*) pemrograman yang kemungkinan akan dapat digunakan untuk mengimplementasikan *algorithmic art* (seni algoritmik) yaitu : python, JavaScript Library (p5js.org), openprocessing (www.openprocessing.org), dan lain-lain.

Tujuan penelitian ini untuk merancang motif geometris dengan menggunakan *Algorithmic Art* dan mengimplementasikannya melalui *Algorithmic Art*. Penelitian ini untuk memperkaya motif yang dikembangkan dengan *Algorithmic Art*, sehingga menjadi gagasan baru dalam hal pembuatan motifnya. Diharapkan *Algorithmic Art* ini menjadi nilai kebaruan dalam mengembangkan motif. Tahapan proses awal pembuatan motif, menggunakan komputer sehingga dapat memproses lebih baik dibandingkan dengan proses manual. Urgensi penelitian ini adalah menghasilkan invensi, berupa program yang dapat menghasilkan motif dengan menggunakan metode yang berbasis *Algorithmic Art*.

Data menunjukkan bahwa melalui penjelasan sejarah algoritma dalam seni visual akan banyak menyentuh banyak fase di setiap budaya di setiap kesempatan, misalnya konon Mesir untuk menggambar sosok manusia, permainan geometris tak terbatas dalam seni Islam dan peran perspektif linier dan proporsi dalam seni Renaisans, sedangkan di Cina ditemukan Manual Biji Mustard dan di Byzantium konvensi untuk lukisan ikon. Di Eropa, pada abad ketujuh belas ditemukan algoritma yang sangat canggih untuk merencanakan perspektif yang memusingkan yang menggambarkan perjalanan dari bumi ke surga. Selanjutnya, mengenai sistem notasi untuk seni visual memainkan peran terbatas jika dibandingkan dengan sistem notasi untuk musik. Seorang komposer yang berbakat dapat menggubah musik untuk bagian musik yang sangat menyentuh yang dapat dimainkan ratusan tahun kemudian oleh seorang virtuoso yang terampil. Tetapi tidak demikian bagi seniman karena tidak dapat dengan mudah menggunakan algoritma untuk menciptakan ruang perspektif dalam sebuah karya seninya. Pada sejarah abad ke- 20 ditemukan beberapa contoh seniman (di tahun 1960-an dan 1970-an) yaitu : Fluxus, seniman minimalis dan konseptual menggunakan berbagai metode spesifikasi prosedural yang menantang konsepsi seni tradisional. Misalnya, karya-karya awal George Brecht pada dasarnya adalah instruksi yang sangat ramping yang dicetak pada kartu; kartu, dengan instruksi adalah karya seni (Kleiner, 2014). Seni konseptual

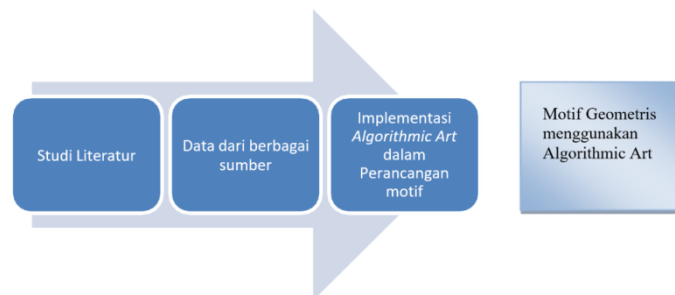
dalam berbagai bentuknya mendahului apa yang disebut seni komputer. Baru setelah seniman memperoleh akses ke sumberdaya komputasi, maka dapat menyusun generator bentuk untuk menghasilkan bentuk seni (Adajian, 2018; Seel, 2013).

Ceric dalam penelitiannya yang berjudul *Algorithmic art: Technology, mathematics and art*, yang menjelaskan bahwa *Algorithmic Art* (seni algoritmik) sebagai seni visual yang dibuat berdasarkan algoritma yang sepenuhnya menggambarkan generasi gambar. Seni algoritmik berakar pada pendekatan rasional untuk seni, teknologi dan aplikasi matematika, yang didasarkan pada teknologi komputer dan khususnya pada pemrograman yang membuat *Algorithmic Art* (seni algoritmik) menjadi mungkin dan paling berpengaruh sehingga dapat dilakukan eksplorasi hubungan teknologi dan matematika ke seni dan kemudian menggambarkan seni algoritmik itu sendiri (Ceric, 2008). Oleh karena itu Ceric selanjutnya mengeksplorasi hubungan teknologi, matematika dengan seni, sehingga dapat menggambarkan seni algoritmik itu sendiri. Ceric juga menegaskan bahwa seni rupa telah melahirkan sejumlah teknologi, terutama dalam enam abad terakhir. Beberapa di antaranya adalah gravure (ditemukan sekitar tahun 1400), cetak (1440), litografi (1796), fotografi (1830) dan serigrifi (1920). Teknologi selalu menjadi pendorong bagi seniman untuk mencoba menggunakannya untuk penciptaan ekspresi baru, ekspresi yang tidak mungkin dilakukan dengan alat teknologi yang ada (Ceric, 2008).

Javier Montenegro Joo, juga menyampaikan bahwa *Algorithmic Art* (seni algoritmik) adalah salah satu dari beberapa manifestasi seni digital atau seni oleh komputer, yang bergantung keberadaannya kepada komputer. Di antara ekspresi *Algorithmic Art* (seni algoritmik) yang menghasilkan karya seni yang indah dan gambar menakjubkan yang dibuat dari aturan/tata cara komputasi (algoritma) pada ekspresi matematika sebagai persamaan, rumus, aturan logika, dll (Joo, 2015).

Metode

Metode penelitian ini menggunakan metode deskriptif kualitatif yaitu untuk mengkaji data-data yang berasal dari gambar (Creswell, 2014; Taylor et al., 2016). Peneliti mencari dan mempelajari perangkat-perangkat lunak yang dapat menghasilkan motif geometris secara digital melalui komputer. Perangkat yang digunakan pada penelitian ini adalah perangkat lunak *processing* dengan *algorithmic art*. Setelah mempelajari penggunaan perangkat lunak *processing*, peneliti berupaya membuat program-program untuk membentuk motif-motif geometris. Sehingga motif geometris yang dihasilkan tidak lagi digambar menggunakan pensil dan kertas, tetapi dihasilkan dengan merancang program-program pembentuk motif geometris. Data diperoleh dari hasil gambar yang dibuat dengan menggunakan *algorithmic art*. Gambar-gambar yang dirancang ini merupakan hasil dari susunan bentuk-bentuk geometris yang terstruktur yang dibuat melalui *generative art*. Metode kuantitatif digunakan dalam *Algorithmic Art*, sedangkan metode praktik digunakan untuk mewujudkan salah satu desain dalam bentuk produk yang dapat memvisualisasikan motif. Metode-metode ini dipilih karena memiliki kemampuan menghasilkan motif secara variatif, sehingga dianggap tepat untuk membentuk sebuah motif baru berdasarkan rumus matematika (Ratnadewi et al., 2020, 2021). Tahap awal yang dilakukan melalui studi literatur, kemudian dilanjutkan dengan memodelkan objek melalui perintah pada program *processing* seperti *line()* untuk membuat garis, *triangle()* untuk membuat bentuk segitiga, *quad()* untuk membuat bentuk jajaran genjang, *rect()* untuk membuat bentuk persegi panjang, *ellipse()* untuk membuat bentuk elips, *bezier()* untuk membuat bentuk kurva melengkung, dan menggambarkan motif yang dibentuk dengan *Algorithmic Art*.



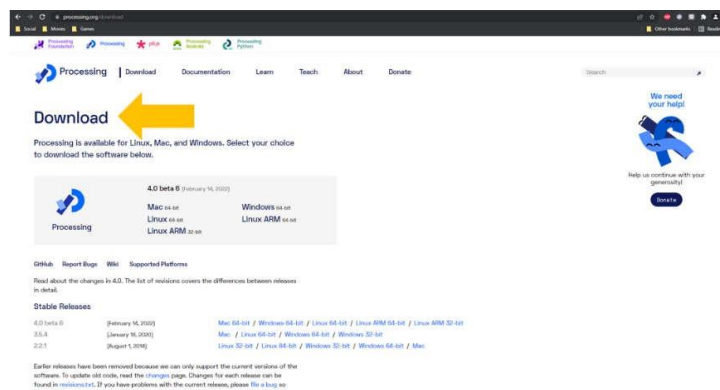
Gambar 1. Prosedur sistem aplikasi pembuatan motif geometris dengan menggunakan *Algorithmic Art*
Dokumentasi: Tim Peneliti 2021

Penelitian tahap awal ini adalah membuat motif geometris dengan variasi bentuk, warna, komposisi dan keteraturan pola. Untuk merealisasikan rancangan motif geometris dengan menggunakan *Algorithmic Art*. Pembentukan pola dilakukan dengan merancang terlebih dahulu motif yang diinginkan, kemudian mencari perintah-perintah pada program processing yang dapat dan cocok digunakan, kemudian menyusun urutan-urutan perintah yang akan digunakan untuk merekonstruksi motif yang diinginkan, setelah program selesai maka urutan perintah mulai dijalankan dan hasil gambar dari *Algorithmic Art* akan terbentuk. Jika motif yang dihasilkan tidak sesuai, maka program dapat diperbaiki kembali dengan mengubah perintah atau nilai yang digunakan sebelumnya dengan perintah dan nilai yang baru. Percobaan ini dapat dilakukan berulang-ulang sampai motif yang dihasilkan telah sesuai dengan yang diinginkan.

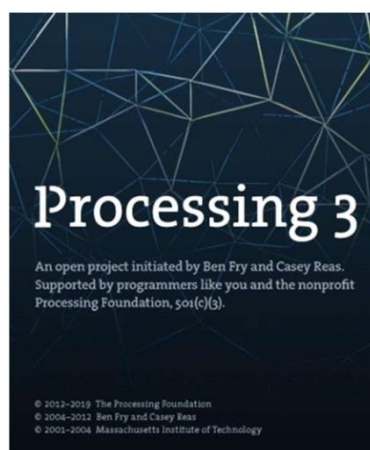
Hasil dan Pembahasan

Hasil

Penelitian ini dilakukan melalui proses instalasi perangkat lunak yang akan digunakan untuk mengimplementasikan *Algorithmic Art*, yaitu perangkat lunak *Processing* yang dapat di *download* di www.processing.org/download/ dan dapat digunakan untuk sistem operasi Mac OS X, sistem operasi Windows, dan sistem operasi Linux. Perangkat lunak *Processing* memiliki kelebihan yaitu berukuran kecil dan ringan untuk digunakan, sehingga pemakai tidak perlu menyediakan komputer yang spesifikasinya tinggi. Keuntungan lain perangkat lunak ini menggunakan perintah-perintah dalam bahasa pemrograman yang mudah untuk dipahami dan pengguna tidak perlu menyediakan waktu yang lama untuk mempelajarinya.

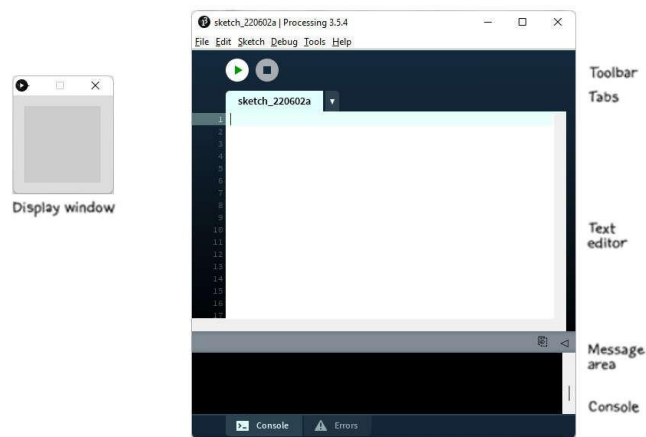


Gambar 2. Tampilan Situs Web : processing.org/download
Dokumentasi: Tim Peneliti. 2022



Gambar 3. Tampilan Awal *software Processing* setelah dilakukan instalasi pada komputer.

Berikut ini (pada Gambar 3) adalah *Processing Development Environment* yang terdiri dari display window, dan bagian editor untuk pemrograman (terdapat *Toolbar Tabs*, *Text editor*, *Message area*, dan *Console*).



Gambar 4. Processing Development Environment.

Fungsi-fungsi komponen *toolbar* pada *software Processing* :



Run

Berfungsi untuk mengeksekusi/menjalankan sebuah program yang telah selesai di tulis pada area kerja *software Processing*.



Stop

Stop atau berhenti (jeda) adalah untuk menghentikan proses eksekusi dari *software Processing*.

Penjelasan beberapa fungsi yang berkaitan dengan bentuk geometri, sebagai berikut :

1. Fungsi Garis (*line*)

Garis adalah jarak terpendek antara dua titik dan dengan demikian membutuhkan dua titik untuk mendefinisikannya; titik awal dan titik akhir. Setiap titik yang digunakan diformat dengan posisi x pertama dan posisi y kedua. Dalam memproses fungsi untuk menggambar garis adalah: *line*(mulai x, mulai y, akhir x, akhir y). Seperti sebuah titik, maka sebuah garis memiliki ketebalan dan warna. Hal ini ditentukan dengan cara yang persis sama dengan fungsi *strokeWeight()* dan *stroke()*.

2. Fungsi Segitiga (*triangle*)

Segitiga adalah bentuk tiga sisi yang dibangun dari garis-garis yang ditarik di antara tiga poin. Dengan demikian, fungsi segitiga pada *software Processing* membutuhkan tiga set koordinatnya sebagai berikut: *triangle*(x1, y1, x2, y2, x3, y3); maka dapat dilihat bahwa setiap pasangan x,y untuk koordinat disimpan bersama-sama seperti untuk garis. Setiap titik disebut simpul. Setiap garis yang membentuk ketiga sisinya disebut rusuk.

3. Fungsi *rect* untuk Bentuk Bujur Sangkar dan Persegi Panjang

Bujur sangkar dan persegi panjang mempunyai bentuk geometri yang hampir sama karena keduanya memiliki empat sisi yang menghubungkan empat simpul. Semua sisi bujur sangkar sama panjangnya, sedangkan persegi panjang memiliki dua pasang sisi yang sama panjang. Keduanya (baik bujur sangkar dan persegi panjang) membentuk semua sudut interior adalah sebesar 90 derajat. Pada *software Processing*, fungsi untuk membuat bentuk bujursangkar akan sama dengan fungsi untuk membuat bentuk persegi panjang: *rect*(x, y, lebar, tinggi); selain itu juga dapat mengatur bobot goresan dan warna goresan dan isian dengan cara yang sama seperti untuk bentuk segitiga dengan memasukkan *strokeWeight()*, *stroke()* dan *fill()* pemanggilan fungsi sebelum menggambar bentuk.

4. Fungsi *ellipse* untuk Bentuk Lingkaran dan Elips

Lingkaran dan elips adalah yang paling menarik dari semua bentuk karena mereka memiliki banyak properti yang menarik, karena konstruksi yang dimiliki yang mengandung kurva, yaitu kurva tertutup sederhana yang tertutup berarti sentuhan awal dan akhir untuk membuat bagian dalam. Lingkaran didefinisikan sebagai himpunan semua titik pada bidang yang berjarak sama dari titik tengah (pusat lingkaran). Pada *software Processing*, sebuah lingkaran digambar dengan fungsi *ellipse*, sehingga fungsinya juga digunakan untuk menggambar elips. Lingkaran pada dasarnya adalah elips dengan lebar yang sama dan tinggi, maka dapat menggunakan fungsi yang sama. Fungsi *ellipse* sebagai berikut: *ellipse*(x, y, lebar, tinggi); yaitu *default* x, y,

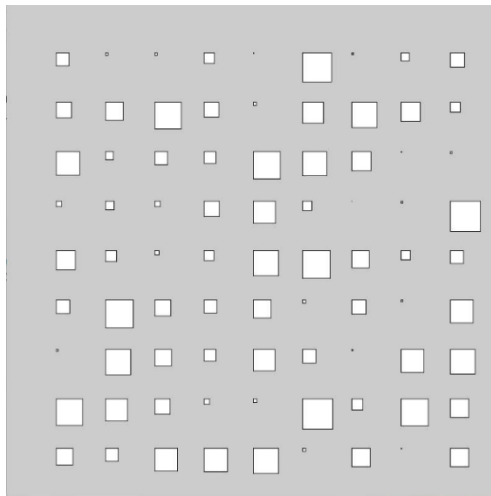
adalah pusat. Jika lebar dan tinggi sama maka dibuat lingkaran, sedangkan jika lebar dan tinggi tidak sama, maka akan dihasilkan bentuk elips.

5. Fungsi *arc*

Sektor dan busur dapat digambar pada *software Processing* menggunakan fungsi *arc*. Apabila sebagai busur, maka digambar pada keliling lingkaran atau elips, maka fungsi yang digunakan sama parameternya sebagai elips default dan diikuti oleh sudut awal dan sudut akhir, yaitu: *arc(x, y, lebar, tinggi, start_angle, end_angle)*; besarnya sudut yang ditentukan diukur dari 0 derajat yang berada di horizontal di sisi kanan dari elips dan meningkat searah jarum jam.

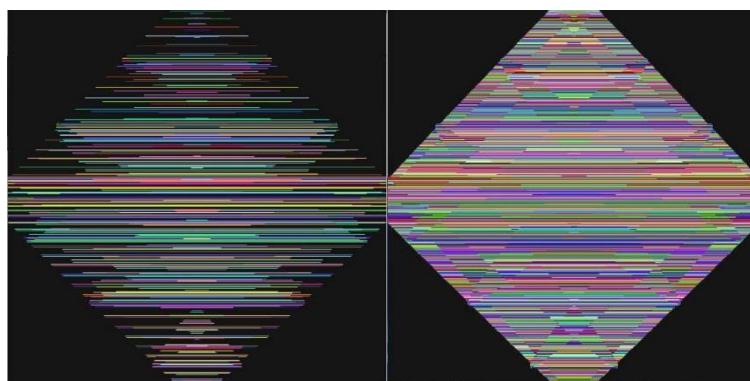
Pembahasan

Hasil dari penelitian selanjutnya adalah melakukan pemrograman dengan menggunakan *software Processing* untuk menggambar motif geometris seperti dapat dilihat pada contoh-contoh tampilan hasil program berikut ini :



Gambar 5. Hasil program yang untuk membuat bentuk-bentuk bujursangkar dengan menggunakan fungsi *random*

Pada Gambar 5 di atas diperlihatkan contoh tampilan hasil program untuk membuat motif geometris sederhana berupa bentuk-bentuk bujur sangkar dengan menggunakan fungsi *random*. Dengan menggunakan perintah *random* maka ukuran bujur sangkar yang dihasilkan akan bervariasi. Peletakan bujur sangkar dengan ukuran berbeda-beda pada posisi acak akan menghasilkan motif yang artistic dan menarik.



Gambar 6. Hasil program berupa garis berwarna-warni dengan parameter $x(t)$ dan $y(t)$ serta menggunakan fungsi *random*

Pada Gambar 6 di atas, adalah hasil dari program untuk membuat garis berwarna-warni dengan parameter $x(t)$ dan $y(t)$ serta menggunakan fungsi *random*. Parameter $x(t)$ dan $y(t)$ digunakan untuk memberi masukan pada program nilai yang berubah-ubah sesuai dengan nilai t , pada percobaan ini nilai t dinaikkan terus sehingga diperoleh panjang garis dan letak yang berubah-ubah. Selain itu digunakan juga fungsi *random* agar warna yang dihasilkan berubah-ubah secara acak. Hasil yang diperoleh akan menghasilkan variasi warna, garis, panjang garis, dan letak garis yang berubah-ubah sehingga menghasilkan komposisi warna yang indah.

Motif dengan bentuk lingkaran yang akan membentuk motif kawung menggunakan program seperti di bawah ini:

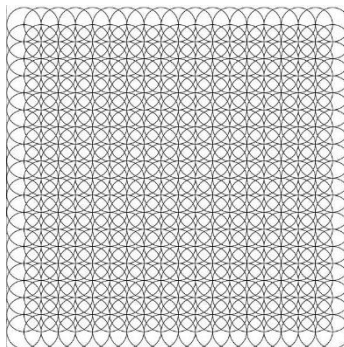
```
1 void setup() {  
2   size(300, 300);  
3   noFill();  
4   stroke(255);  
5   strokeWeight(8);  
6 }  
7  
8 void draw() {  
9   background(32);  
10  
11   // corner circles  
12   circle(0, 0, 300);  
13   circle(300, 0, 300);  
14   circle(0, 300, 300);  
15   circle(300, 300, 300);  
16  
17   // top, bottom, left, and right circles  
18   circle(150, 0, 300);  
19   circle(150, 300, 300);  
20   circle(0, 150, 300);  
21   circle(300, 150, 300);  
22  
23   // centered circles  
24   circle(150, 150, 300);  
25   circle(150, 150, 150);  
26 }
```

Gambar 7. Program yang menghasilkan pola pada gambar 8



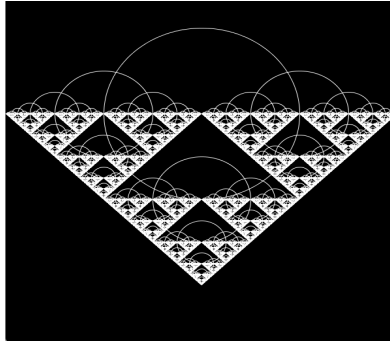
Gambar 8. Hasil program untuk membuat bentuk lingkaran-lingkaran yang saling berpotongan

Pada Gambar 8 di atas, adalah hasil program untuk membuat bentuk lingkaran-lingkaran yang saling berpotongan. Di sini posisi dan ukuran lingkaran yang diameternya terdiri dari dua ukuran. Lingkaran dengan dua ukuran akan diletakkan di posisi tertentu sehingga dihasilkan sinergi bentuk yang indah. Untuk membuat bentuk yang teratur, titik pusat lingkaran diatur sepusat untuk lingkaran berukuran kecil dan berukuran besar, kemudian penempatan posisi lingkaran lain diatur setengah ketinggian lingkaran yang lain baik ke sisi vertikal maupun ke sisi horizontal.



Gambar 9. Hasil program berupa bentuk lingkaran-lingkaran dengan menggunakan pengulangan

Pada Gambar 9 di atas, adalah hasil dari program untuk membuat bentuk lingkaran-lingkaran dengan menggunakan pengulangan. Pada rancangan ini ukuran lingkaran dibuat sama, hanya posisi yang diatur berjarak setengah dari diameter lingkaran, baik ke arah vertikal maupun ke arah horizontal. Hasil yang diperoleh berbeda dengan rancangan pada Gambar 7.



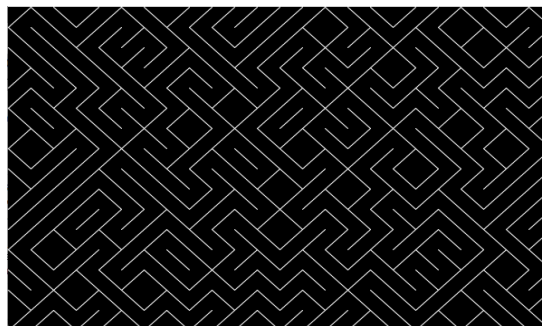
Gambar 10. Hasil program berupa bentuk lingkaran-lingkaran dengan menggunakan rekursif

Pada Gambar 10 di atas, adalah hasil dari program untuk membuat bentuk lingkaran-lingkaran dengan menggunakan rekursif. Pada rancangan, perancang mengubah ide dengan mengatur ukuran lingkaran dari yang berukuran kecil sampai besar dengan berbagai ukuran. Kemudian peletakkan lingkaran juga diatur agar lingkaran yang ukurannya lebih kecil, titik pusat lingkaran terletak digaris lingkaran besar di sisi kanan dan di sisi kirinya. Pengulangan dilakukan ke kiri dan ke kanan serta tambahan bentuk ke bawah sehingga terlihat seperti membentuk bentuk segitiga. Hal ini berlaku untuk semua ukuran, sehingga hasil motif yang dihasilkan berbeda dengan rancangan sebelumnya.

Motif banji dapat dibuat dengan menggunakan program berikut di bawah ini:

```
1 int x = 0;
2 int y = 0;
3 int d = 25;
4
5 void setup() {
6   size(600, 400);
7   stroke(255);
8   background(0);
9 }
10
11 void draw() {
12   float r = floor(random(0,2));
13   if (r == 1) line(x, y, x+d, y+d);
14   if (r == 0) line(x+d, y, x, y+d);
15   if (y < height) x+=d;
16   if (x > width){
17     y += d;
18     x = 0;
19   }
20 }
```

Gambar 11. Program yang menghasilkan pola pada gambar 12

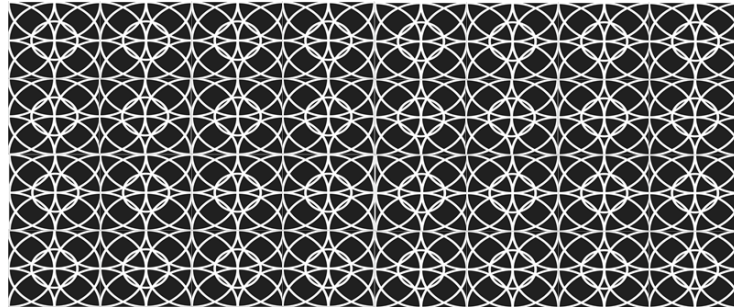


Gambar 12. Hasil program untuk membuat garis-garis yang membentuk pola dengan menggunakan fungsi random.

Pada Gambar 12 di atas, adalah hasil dari program untuk membuat garis-garis yang membentuk pola dengan menggunakan fungsi random. Pada rancangan di sini dibuat garis-garis dengan ukuran panjang dan

posisi yang diatur berbeda. Perancang membuat ketentuan untuk mengatur letak di sumbu vertikal dan letak di sumbu horizontal yang bervariasi sehingga diperoleh motif yang elegan.

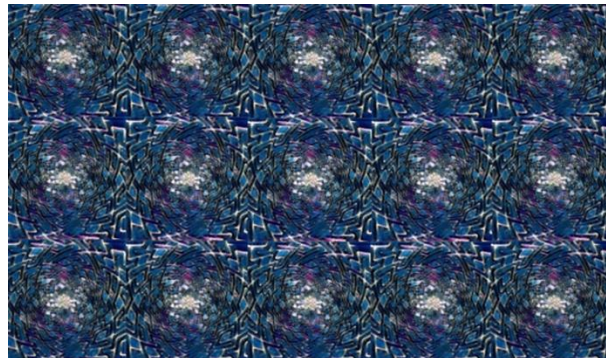
Motif-motif batik yang dihasilkan dari beberapa program di atas dapat dilihat di bawah ini



Gambar 13. Pengulangan bentuk lingkaran yang dibuat repetitive menghasilkan bentuk motif kawung.

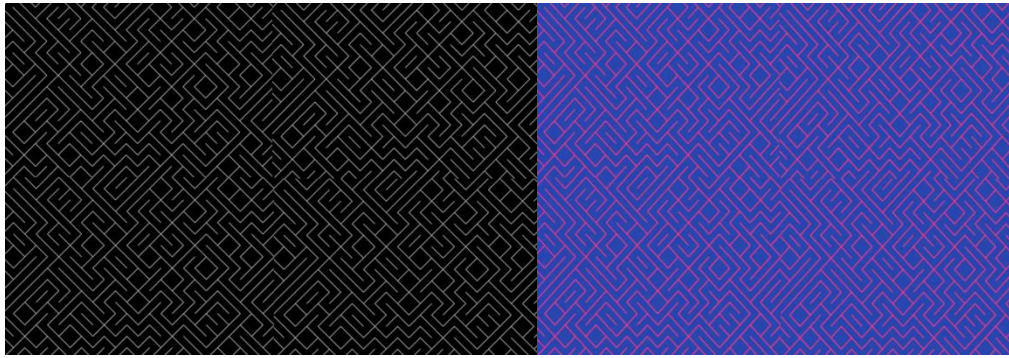
Menurut Prasetyo motif batik adalah kerangka gambar yang mewujudkan batik secara keseluruhan (Prasetyo, 2016). Lebih lanjut dijelaskan bahwa motif batik dibuat pada bidang-bidang segi tiga, segi empat, dan/atau lingkaran. Motif pada gambar yang membetuk pengulangan lingkaran ini masuk dalam kategori geometris, dan motif ini dalam batik dikenal dengan motif Kawung. Sejarah merekam bahwa batik dengan motif Kawung ini sudah dikenal di Jawa sejak abad 13, dapat ditemukan pada beberapa relief yang terdapat pada dinding candi Prambanan dan daerah Kediri (Hartanti & Setiawan, 2019). Motif ini dibuat dengan cara pengulangan dari gambar 7 yaitu hasil program untuk membuat bentuk lingkaran-lingkaran yang saling berpotongan.

Menurut Hartanti (2019) lingkaran-lingkaran yang terdapat dalam motif kawung, terkadang diisi dengan dua atau lebih tanda silang atau ornamen lain seperti garis-garis berpotongan atau titik-titik. Pada gambar ini rancangan motif lebih kompleks karena memvariasikan lagi lingkaran kecil pada perpotongan arah garis diagonal yang saling menyilang. Rancangan motif yang sama dengan beberapa aplikasi yang terdapat dalam komputer seperti picsart dapat dihasilkan motif geometris yang saling berputar seperti di bawah ini:



Gambar 14. Pengulangan bentuk lingkaran yang dibuat repetitiv dibantu dengan aplikasi picsart menghasilkan motif geometris yang kompleks, dan memungkinkan lebih banyak lagi motif geometris.

Karya di atas dibuat dengan bantuan digital. Saat ini digital seolah menjadi bagian dari proses penciptaan. Berbagai aplikasi dan *software* dalam komputer saat ini tersedia untuk banyak perancangan, sehingga merevolusi cara berpikir dan juga proses kreatif. Berbagai kerja yang terkait dengan seni yang mungkin tidak dapat diselesaikan dengan metode tradisional, sekarang dimungkinkan melalui digital, yang salah satunya adalah *generative art* secara tidak langsung memfasilitasi proses tradisional, yang menawarkan banyak kemungkinan baru dalam proses kreatif termasuk proses pembentukan ide, sehingga dapat banyak dihasilkan berbagai karya kreatif yang tidak biasa.



Gambar 15. Pengulangan dari hasil program untuk membuat garis-garis yang membentuk pola dengan menggunakan fungsi random, dihasilkan motif banji dengan latar berwarna hitam (ki) berwarna ungu (ka).

Motif ini dalam batik dikenal dengan motif banji. Motif ini memiliki makna keteraturan dalam kehidupan atau kunci perhiasan yang terkunci rapat (Prasetyo, 2016). Motif ini dibuat dengan cara pengulangan dari gambar 10 yaitu hasil dari program untuk membuat garis-garis yang membentuk pola dengan menggunakan fungsi random. Rancangan motif yang sama dengan perubahan warna dari latar hitam menjadi ungu dapat dilihat di bawah ini. Di Indonesia yang sangat kaya akan motif geometris merupakan peluang besar untuk membuat variasi dalam perancangan motifnya.

Simpulan

Algoritmik seni merupakan salah satu solusi cara dalam membuat motif geometris yang dirancang melalui teknologi komputer. Proses perancangan motif batik yang menghasilkan pola-pola geometris belum banyak dipergunakan, masih harus terus di sosialisasikan kepada masyarakat, khususnya di sentra pematikan. Konsep geometris dengan menggunakan teknologi komputer menawarkan keberagaman objek yang dibentuk dari rumus matematika dan menghasilkan garis horizontal, vertikal dan saling membentuk sebuah objek. Prospek pengembangan hasil penelitian ini, dapat diimplementasikan kedalam perancangan motif yang dibuat di sentra pematikan yang tentunya membutuhkan arahan dan pelatihan, banyak variatif kemungkinan motif yang dapat dirancang sehingga dibutuhkan penelitian selanjutnya dengan memanfaatkan berbagai lintas keilmuan.

Daftar Rujukan

- Adajian, T. (2018). The Definition of Art. Stanford Encyclopedia of Philosophy. <https://plato.stanford.edu/entries/art-definition/>
- Ceric, V. (2008). Algorithmic Art: Technology, Mathematics and Art. Proceedings of the International Conference on Information Technology Interfaces, ITI, May, 75–81. <https://doi.org/10.1109/ITI.2008.4588386>
- Creswell, J. W. (2014). *Penelitian Kualitatif dan Desain Riset* (S. Z. Qudsy (ed.); 3rd ed.). Pustaka Pelajar.
- Dehlinger, H. (2020). Generative Art. Journal of Mathematics and the Arts, 14(1–2), 33–36. <https://doi.org/10.1080/17513472.2020.1729058>
- Gherman, I. (2018). Text and Image in the Digital Age – The Digital Revolution. Open Journal for Information Technology, 1(1), 13–24. <https://doi.org/10.32591/coas.ojit.0101.02013g>
- Hartanti, G., & Setiawan, B. (2019). Pendokumentasian Aplikasi Ragam Hias Batik Jawa Tengah Motif Kawung, Sebagai Upaya Konservasi Budaya Bangsa Khususnya Pada Perancangan Interior. Aksens, 3(2), 25–37. <https://doi.org/10.37715/aksen.v3i2.807>
- Joo, J. M. (2015). The Breathtaking Algorithmic Art. In The VirtualDynamics Intelligencer (Issue October 2006). https://www.researchgate.net/publication/288827387_The_Breathtaking_Algorithmic_Art
- Kleiner, F. S. (2014). Gardner's Art Through The Ages: A Concisw Western History. In S. A. Poore (Ed.), Wadsworth, Cengage Learning, USA (Third Edit). Wadsworth, Cengage Learning.
- Monro, G. (2009). Emergence and Generative Art [University of Sydney]. In Leonardo (Vol. 42, Issue 5). <https://doi.org/10.1162/leon.2009.42.5.476>
- Pandanwangi, A. (2020). Upaya Perupa dalam Menyikapi Pandemi Covid 19. Prosiding Seminar Nasional Hardiknas, 91–98. <http://proceedings.ideaspublishing.co.id/index.php/hardiknas/article/view/14>

- Prasetyo, S. A. (2016). Karakteristik Motif Batik Kendal Interpretasi dari Wilayah dan Letak Geografis. *Jurnal Imajinasi*, 10(1), 51–60. <https://doi.org/10.15294/imajinasi.v10i1.8816>
- Ratnadewi, Prijono, A., & Pandanwangi, A. (2020). Application of Turtle Graphics to Kawung Batik in Indonesia. *International Journal of Innovation, Creativity and Change*, 13(2), 643–658.
- Ratnadewi, R., Pandanwangi, A., & Prijono, A. (2021). Bahasa Rupa Motif Batik Ciamis Melalui Program Lindenmayer Systems. *Jurnal Bahasa Rupa*, 5(1), 122–133. <https://doi.org/10.31598/bahasarupa.v5i1.961>
- Seel, G. (2013). The Role of Art in History and the Art of the Future. *SAGE Journal*, 59(1–2), 158–167.
- Suseno, B. A. (2014). Eksistensi Seni Grafis Monoprint dalam Kesenirupaan Yogyakarta. *Journal of Urban Society's Arts*, 1(2), 110–120. <https://doi.org/10.24821/jousa.v1i2.792>
- Tabrani, P. (2014). *Proses Kreasi-Gambar Anak-Proses Belajar* (1st ed.). Erlangga.
- Taylor, S. J., Bogdan, R., & DeVault, M. L. (2016). *Qualitative Reseach Methods* (4th Editio). John Wiley & Sons, Inc.
- Tempel, M. (2017). Generative art for all. *Journal of Innovation and Entrepreneurship*, 6(1), 1–14. <https://doi.org/10.1186/s13731-017-0072-1>

