

# Implementasi Motif Batik Truntum pada Aplikasi Bingkai Foto Berbasis Android

Purba Daru Kusuma  
Program Studi Sistem Komputer  
Universitas Telkom  
Bandung, Indonesia  
purbodaru@gmail.com

**Abstrak**—Terdapat banyak aplikasi pengolahan citra berbasis Android. Salah satu kelompok aplikasi tersebut adalah aplikasi pembuatan bingkai foto. Hanya saja terdapat beberapa masalah dalam aplikasi tersebut yang tersedia di *Play Store*. Pertama, sebagian aplikasi hanya membuat bingkai foto dengan menggunakan foto *template* yang sudah ada dan bukan secara algoritmik. Kedua, aplikasi berbasis Android yang mengangkat budaya Indonesia masih terbatas. Penelitian ini dilakukan sebagai salah satu usaha mengembangkan aplikasi pengolahan citra berbasis Android sekaligus ikut serta dalam mengangkat budaya tradisional. Penelitian ini berfokus pada implementasi motif batik Truntum ke dalam pembuatan bingkai foto secara digital. Melalui penelitian ini dihasilkan beberapa desain bingkai foto yang dibuat secara algoritmik dengan mengadopsi motif batik Truntum.

**Kata kunci**—*aplikasi bergerak, Android, bingkai foto, batik Truntum*

## I. PENDAHULUAN

Saat ini, tersedia banyak aplikasi pengolahan citra berbasis Android dan dapat ditemukan dengan mudah melalui *Play Store*, antara lain Camera+, PicsPlay Pro, Pixlr-O-Matic, Photoshop Express, dan masih banyak lagi (Departemen Kesehatan Republik Indonesia, 2006). Salah satu kelompok dalam aplikasi citra tersebut adalah aplikasi pembuatan bingkai foto digital. Beberapa aplikasi memang berfokus pada pembuatan bingkai foto. Beberapa aplikasi yang lain menyediakan fitur pembuatan bingkai foto sebagai salah satu fitur yang tersedia dan bukan secara khusus memfokuskan aplikasi tersebut untuk membuat bingkai foto.

Pada sebagian aplikasi, pembuatan bingkai foto dilakukan dengan cara menempelkan gambar yang akan dijadikan bingkai foto dan kemudian menggabungkannya dengan foto yang akan dibingkai. Metode ini menimbulkan beberapa keterbatasan. Yang pertama, semua orang yang menggunakan gambar tertentu untuk dijadikan bingkai akan mendapatkan bingkai yang sama. Pengguna tidak dapat memilih warna bingkai.

Di tengah pertumbuhan aplikasi berbasis Android yang sangat pesat dengan jumlah aplikasi lebih dari 1 juta aplikasi pada tahun 2014 [4], kontribusi aplikasi yang mengangkat budaya Indonesia masih terbatas. Kondisi ini disebabkan salah satunya adalah jumlah aplikasi Android buatan pengembang Indonesia masih sedikit jika dibandingkan jumlah aplikasi yang

tersedia di *Play Store*. Kondisi ini tentu kontras dengan jumlah pengguna Android di Indonesia yang sangat banyak.

Berdasarkan kedua kondisi di atas, maka dikembangkan aplikasi citra berbasis Android dengan fitur utama adalah pembuatan bingkai foto secara digital. Berbeda dengan beberapa aplikasi yang menyediakan bingkai foto dengan cara menempel gambar yang sudah ada sebagai bingkai foto, aplikasi ini membuat bingkai foto secara algoritmik iteratif. Dengan demikian, terdapat kemungkinan perbedaan hasil, yaitu bingkai foto yang dibangkitkan, khususnya pada ukuran obyek-obyek dalam bingkai foto dan warna obyek-obyek tersebut. Selain itu, pengguna juga dapat memilih warna tertentu sebagai warna acuan bingkai fotonya.

Dalam rangka ikut mengangkat budaya tradisional Indonesia, terdapat beberapa model bingkai foto dengan motif yang mengadopsi motif batik Truntum. Salah satu pertimbangan pemilihan motif Truntum adalah motif Truntum memiliki ciri khas, yaitu bunga melati berukuran kecil yang disusun dengan teratur [2]. Hal ini mempermudah pembuatan desain bingkai foto karena motif Truntum termasuk motif yang sederhana. Motif ini juga dipilih karena ukuran obyeknya kecil. Hal ini dianggap baik karena ukuran bingkai foto relatif kecil dibandingkan ukuran foto itu sendiri. Ukuran obyek yang kecil tetapi banyak diharapkan dapat menghindarkan bingkai foto yang lebih dominan dibandingkan foto itu sendiri.

Posisi penelitian ini adalah sebagai bagian dari pengembangan desain bingkai foto digital yang mengangkat budaya tradisional Indonesia, dalam hal ini batik Truntum. Desain adopsi batik Truntum adalah sebagian dari desain-desain bingkai foto yang terdapat dalam aplikasi pembuatan bingkai foto berbasis Android yang dikembangkan.

## II. ADOPSI MOTIF TRUNTUM

Motif Truntum adalah salah satu motif di antara banyak motif batik yang ada di Indonesia. Motif Truntum dapat digunakan secara sendiri atau dikombinasikan dengan motif-motif lain, antara lain Parang, Sidomukti, Sidomulyo, dan yang lain. Motif Truntum identik dengan bentuk bunga melati sederhana yang berukuran kecil dan dalam jumlah banyak. Dalam filosofi budaya Jawa, Truntum diidentikkan sebagai calon kehidupan [3]. Truntum juga dapat didefinisikan sebagai cinta yang tumbuh kembali [5]. Obyek Truntum pada

umumnya seragam. Susunan obyeknya teratur dan dapat dilihat sebagai susunan matrik 2 dimensi. Jarak antar obyek pada motif Truntum pada umumnya sama, baik secara vertikal maupun horisontal. Contoh motif Truntum dapat dilihat pada gambar 1.

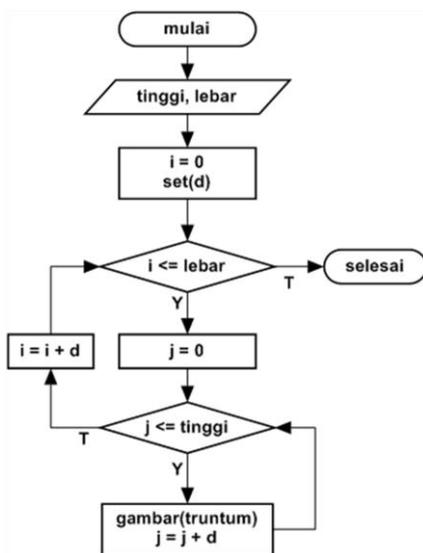
Pada gambar 1 dapat dilihat bahwa susunan obyek pada motif Truntum bersifat teratur. Jarak antar obyek bersifat tetap. Hanya saja, pada gambar 1, selain obyek utama terdapat obyek-obyek pendukung lain dan obyek-obyek pendukung tersebut juga tersusun secara teratur.

Ide dasar obyek utama motif Truntum adalah bunga melati. Dengan demikian, secara bentuk dasar motif Truntum termasuk dalam kelompok motif bunga [6]. Dalam perkembangannya, obyek utama dideskripsikan sebagai kumpulan obyek-obyek yang tersusun melingkar dengan jarak sudut yang sama. Obyek-obyek tersebut dapat berbentuk lingkaran, elips, belah ketupat, ataupun layang-layang. Dengan demikian tidak ada batasan khusus mengenai obyek penyusun obyek utamanya. Pada beberapa kain batik Truntum, terdapat kombinasi bentuk-bentuk dasar elip dengan lingkaran atau kombinasi yang lain.

Dengan sifat tersebut, pengembangan algoritma adopsi motif Truntum menjadi sederhana. Secara umum, dilakukan iterasi 2 tingkat yaitu secara vertikal dan horisontal dengan jarak yang telah ditetapkan. Pada setiap iterasi, diletakkan obyek Truntum. Algoritma ini dapat dilihat dalam bentuk diagram alir pada gambar 2.



Gambar. 1. Contoh motif Truntum



Gambar. 2. Algoritma penempatan Tuntrum.

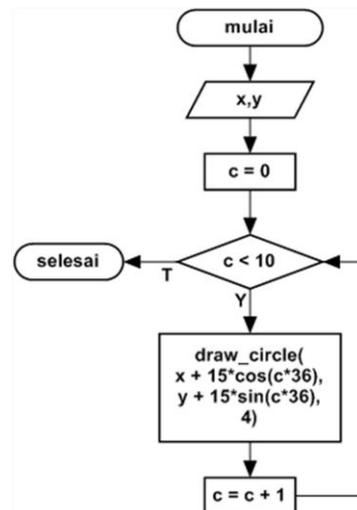
### III. PENGEMBANGAN MODEL DESAIN TRUNTUM

Dalam penelitian ini, dikembangkan 4 model desain bingkai foto dengan mengadopsi motif batik Truntum. Desain dikembangkan dari motif yang paling sederhana sampai dengan motif kombinasi.

#### A. Desain Pertama

Konsep desain yang pertama adalah obyek Truntum berbentuk lingkaran. Terdapat 10 lingkaran kecil yang mengelilingi titik pusat Truntum. Algoritma pembuatan obyek Truntum pada desain yang pertama dapat dilihat pada Gambar 3. Kode pemrograman desain pertama menggunakan bahasa Java dapat dilihat pada Gambar 4.

Penjelasan gambar 3 adalah sebagai berikut. Variabel  $x$  adalah koordinat  $x$  pusat Tuntrum. Variabel  $y$  adalah koordinat  $y$  pusat Tuntrum. Variabel  $c$  berfungsi sebagai counter di mana akan dibuat 10 buah lingkaran yang mengitari pusat Tuntrum. Untuk membuat lingkaran, digunakan fungsi `draw_circle` dengan 3 parameter masukan. Parameter pertama adalah koordinat  $x$  lingkaran. Parameter kedua adalah koordinat  $y$  lingkaran. Parameter ketiga adalah jari-jari lingkaran. Dari fungsi tersebut dapat dilihat bahwa jarak antara pusat Tuntrum dengan pusat lingkaran adalah 15, jari-jari lingkaran adalah 4, dan jarak sudut antar lingkaran adalah 36 derajat. Hasil fungsi tersebut dapat dilihat pada Gambar 5.



Gambar. 3. Algoritma desain pertama.

```

for(c=0;c<10;c++)
{
  canvas.drawCircle(i + (float) (15 *
  Math.cos(Math.toRadians(c * 36))), j +
  (float) (15 * Math.sin(Math.toRadians(c *
  36))), 4, paint);
}

```

Gambar. 4. Kode program desain pertama.

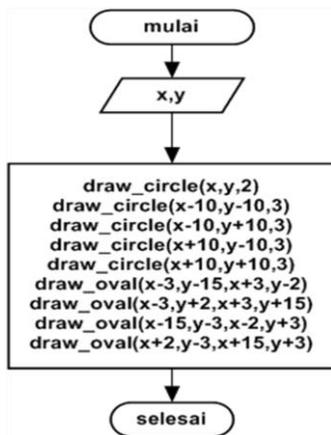


Gambar. 5. Hasil desain pertama.

### B. Desain Kedua

Deskripsi desain Tuntrum yang kedua adalah sebagai berikut. Obyek Tuntrum terdiri dari 4 buah elip di sebelah kiri, kanan, atas, dan bawah pusat Tuntrum. Terdapat lingkaran kecil di pusat Tuntrum. Terdapat 4 lingkaran kecil di sebelah kiri atas, kiri bawah, kanan atas, dan kanan bawah pusat Tuntrum. Algoritma pembuatan obyek Tuntrum desain kedua dapat dilihat pada diagram alir pada Gambar 6. Kode pemrograman desain kedua dapat dilihat pada Gambar 7.

Penjelasan gambar 5 adalah sebagai berikut. Terdapat satu lingkaran dengan besar jari-jari 2 di titik pusat Tuntrum. Selanjutnya terdapat 4 lingkaran di sisi kiri atas, kiri bawah, kanan atas, dan kanan bawah pusat Tuntrum dengan jarak vertikal dan horisontal sebesar 10 dari pusat Tuntrum dengan besar jari-jari 3. Terdapat 4 buah elips dengan panjang 13 dan lebar 6 di sebelah kiri, kanan, atas, dan bawah pusat Tuntrum dengan jarak 2 dari pusat Tuntrum. Hasil fungsi tersebut dapat dilihat pada Gambar 8.

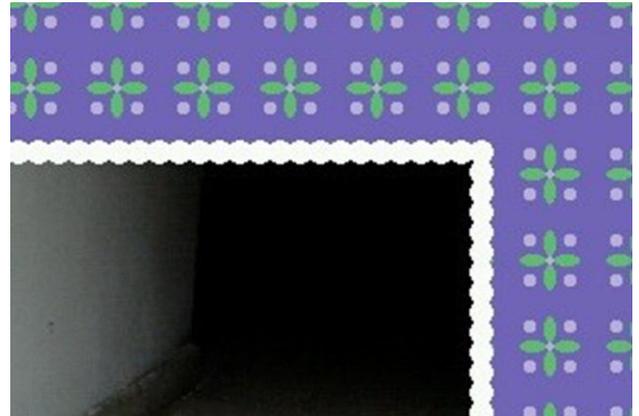


Gambar. 6. Algoritma desain kedua.

```

    canvas.drawCircle(i - 10, j - 10, 3,
    paint);
    canvas.drawCircle(i - 10, j + 10, 3,
    paint);
    canvas.drawCircle(i + 10, j - 10, 3,
    paint);
    canvas.drawCircle(i + 10, j + 10, 3,
    paint);
    canvas.drawCircle(i, j, 2, paint);
    opal.set(i - 3, j - 15, i + 3, j -
    2);canvas.drawOval(opal, paint);
    opal.set(i - 3, j + 2, i + 3, j +
    15);canvas.drawOval(opal, paint);
    opal.set(i - 15, j - 3, i - 2, j +
    3);canvas.drawOval(opal, paint);
    opal.set(i + 2, j - 3, i + 15, j +
    3);canvas.drawOval(opal, paint);
  
```

Gambar. 7. Kode program desain kedua.

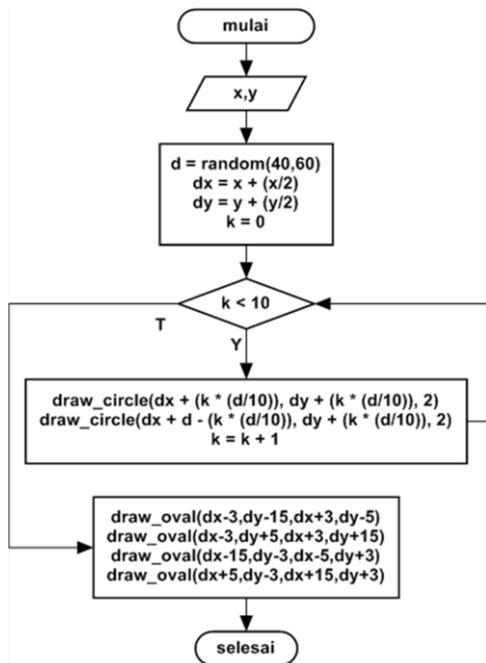


Gambar. 8. Hasil desain kedua.

### C. Desain Ketiga

Desain ketiga merupakan modifikasi dari desain kedua. Pada desain ketiga, lingkaran pusat dan lingkaran yang mengelilingi pusat Tuntrum dihilangkan. Selanjutnya dilakukan penambahan lingkaran-lingkaran kecil yang disusun berderet secara menyilang dari kiri atas ke kanan bawah dan dari kanan atas ke kiri bawah yang selanjutnya bertemu di tengah-tengah Tuntrum. Algoritma pembuatan obyek Tuntrum desain kedua dapat dilihat pada diagram alir pada Gambar 9. Kode program desain ketiga dapat dilihat pada Gambar 10.

Penjelasan gambar 7 adalah sebagai berikut. Panjang dan lebar bidang Tuntrum ditentukan secara acak dengan rentang 40 sampai dengan 60. Dilakukan pengulangan sebanyak 10 kali untuk membuat 10 lingkaran kecil dengan jari-jari 2 masing-masing menyilang. Pusat Tuntrum ditentukan oleh koordinat dx dan dy. Terdapat 4 buah elips di sebelah kiri, kanan, atas, dan bawah pusat Tuntrum dengan lebar 6 dan panjang 10. Hasil fungsi tersebut dapat dilihat pada Gambar 11.



Gambar. 9. Algoritma desain ketiga.

```

dx = i + (d/2);
dy = j + (d/2);
for(k=0;k<10;k++)
{
canvas.drawCircle(dx + (k * (d/10)), dy +
(k * (d/10)), 2, paint);
canvas.drawCircle(dx + d - (k * (d/10)),
dy + (k * (d/10)), 2, paint);
}
opal.set(dx - 3, dy - 15, dx + 3, dy -
5);canvas.drawOval(opal, paint);
opal.set(dx - 3, dy + 5, dx + 3, dy +
15);canvas.drawOval(opal, paint);
opal.set(dx - 15, dy - 3, dx - 5, dy +
3);canvas.drawOval(opal, paint);
opal.set(dx + 5, dy - 3, dx + 15, dy +
3);canvas.drawOval(opal, paint);

```

Gambar. 10. Kode program desain ketiga.

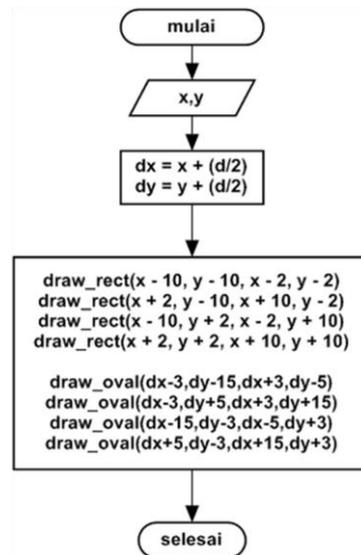


Gambar. 11. Hasil desain ketiga.

#### D. Desain Keempat

Desain keempat merupakan modifikasi dari desain kedua. Seperti pada desain ketiga, pada desain keempat seluruh lingkaran dihilangkan. Selanjutnya ditambahkan 4 bujur sangkar yang terletak di sisi kiri atas, kiri bawah, kanan atas, dan kanan bawah pusat Tuntrum. Algoritma pembuatan obyek Tuntrum desain kedua dapat dilihat pada diagram alir pada Gambar 12. Kode program desain keempat dapat dilihat pada Gambar 13.

Penjelasan gambar 9 adalah sebagai berikut. Seperti pada desain ketiga, panjang elips adalah 10 dan lebar elips adalah 6. Ditambahkan bujur sangkar dengan ukuran sisi 8 pada sisi kiri atas, kiri bawah, kanan atas, dan kanan bawah pusat Tuntrum. Hasil fungsi tersebut dapat dilihat pada Gambar 14.



Gambar. 12. Algoritma desain keempat.

```

dx = i + (d/2);
dy = j + (d/2);
canvas.drawRect(i - 10, j - 10, i - 2, j -
2, paint);
canvas.drawRect(i + 2, j - 10, i + 10, j -
2, paint);
canvas.drawRect(i - 10, j + 2, i - 2, j +
10, paint);
canvas.drawRect(i + 2, j + 2, i + 10, j +
10, paint);
opal.set(dx - 3, dy - 15, dx + 3, dy -
5);canvas.drawOval(opal, paint);
opal.set(dx - 3, dy + 5, dx + 3, dy +
15);canvas.drawOval(opal, paint);
opal.set(dx - 15, dy - 3, dx - 5, dy +
3);canvas.drawOval(opal, paint);
opal.set(dx + 5, dy - 3, dx + 15, dy +
3);canvas.drawOval(opal, paint);

```

Gambar. 13. Kode program desain keempat.

#### IV. KESIMPULAN

Berdasarkan paparan di atas, telah dihasilkan 4 desain bingkai foto yang mengadopsi motif batik Tuntrum. Keempat desain di atas dikembangkan dengan menggunakan 3 bentuk dasar, yaitu lingkaran, bujur sangkar, dan atau elips. Keempat desain tersebut telah diimplementasikan pada aplikasi pembuatan bingkai foto digital berbasis Android. Selain desain di atas, terdapat potensi pengembangan desain-desain yang lain baik mengadopsi motif batik Tuntrum atau motif-motif yang lain.

#### DAFTAR PUSTAKA

- [1] A. Panji, "6 Aplikasi Edit Foto Terbaik di Ponsel", *kompas.com*, 30 Juli 2012.
- [2] Fella dan Aniq, "Batik Truntum", *halamanbahajar.wordpress.com*, 23 Juli 2014.
- [3] Iswahyudi, "Batik dalam Perspektif Mencari Pengakuan: Ketercapaian dan Pelestariannya", Jurusan Pendidikan Seni Rupa UNY, tidak diterbitkan.
- [4] M.A. Goenawan, "Jumlah Aplikasi Android Salip iOS", *detik.com*, 17 Januari 2015.
- [5] S.K. Pandu dan A. Zpalanzani, "Picture Book Sosialisasi Motif dan Kisah Penciptaan Batik Truntum", *Jurnal Tingkat Sarjana bidang Seni Rupa dan Desain*.
- [6] \_\_\_\_\_, *Katalog Batik Khas Yogyakarta*, Kanwil Departemen Perindustrian Propinsi Daerah Istimewa Yogyakarta, 1996.



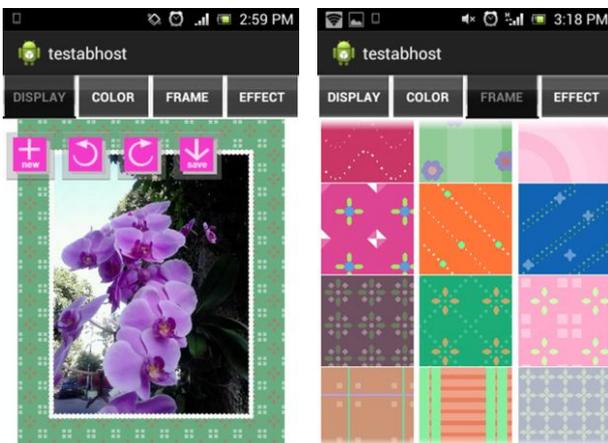
Gambar. 14. Hasil desain keempat.

#### E. Implementasi Aplikasi

Keempat desain tersebut selanjutnya diimplementasikan sebagai bagian dari desain-desain bingkai foto pada aplikasi bingkai foto. Aplikasi berbasis Android ini dikembangkan menggunakan bahasa Java. Antarmuka dibuat sederhana agar pengguna dapat mengoperasikan aplikasi ini dengan mudah. Antarmuka aplikasi dapat dilihat pada Gambar 15.

Pada aplikasi, terdapat 4 tab menu. Tab menu pertama untuk menampilkan hasil bingkai foto. Tab menu kedua digunakan untuk memilih warna bingkai. Tab menu ketiga digunakan untuk memilih jenis bingkai. Tab menu keempat digunakan untuk memilih efek foto.

Pada tab menu displai, terdapat 4 tombol perintah. Tombol tambah digunakan untuk memilih foto yang akan ditambahkan bingkainya. Tombol putar kiri digunakan untuk memutar foto berlawanan jarum jam. Tombol putar kanan digunakan untuk memutar foto searah jarum jam. Tombol simpan digunakan untuk menyimpan hasil foto yang telah ditambah bingkai ke dalam folder tertentu.



Gambar. 15. Antarmuka aplikasi.