**GENERASI *WANG TILES* OTOMATIS MENGGUNAKAN PENDEKATAN PARAMETRIK**

**TESIS**

**Karya tulis sebagai salah satu syarat**

**untuk memperoleh gelar Magister dari**

**Institut Teknologi Bandung**

**Oleh**

**FAIRUZ ASTRA PRATAMA**

**NIM: 23520012**

**(Program Studi Magister Teknik Informatika)**



**INSTITUT TEKNOLOGI BANDUNG  
November 2021**

ABSTRAK

**GENERASI *WANG TILES* OTOMATIS MENGGUNAKAN PENDEKATAN PARAMETRIK**

Oleh

**Fairuz Astra Pratama**

**NIM: 23520012**

**(Program Studi Magister Teknik Informatika)**

Abstrak ditulis dalam bahasa Indonesia yang baik dengan benar. Tulis abstrak dalam bahasa Indonesia di sini. Ukuran huruf pada abstrak adalah 12, jenis huruf *times new roman*, dan tidak ditulis tebal. Abstrak terdiri atas 500 s.d. 800 kata. Kalimat pertama abstrak tesis berjarak tiga spasi dari baris terakhir nama program studi. Kata pertama atau awal paragraf baru dipisahkan dengan jarak satu spasi dari kalimat terakhir paragraf yang mendahuluinya. Pada abstrak harus disertakan kata kunci (*keywords*) dengan maksimal tujuh kata atau kelompok kata (dua kata) yang bermakna tunggal. Peletakan kata kunci terpisah dari abstrak dengan jarak dua spasi. Kata kunci tersebut harus berhubungan dan berasal dari isi abstrak dan **TIDAK BOLEH** berasal/diambil dari tubuh/teks tesis.

Abstrak tesis memuat secara komprehensif permasalahan dan hasil penelitian yang telah dilakukan. Abstrak memuat latar belakang, tujuan, metode dan pendekatan yang digunakan, kajian atau analisis/pembuktian hipotesis/kajian hasil penelitian yang membandingkan dengan penelitian sejenis. Dalam sebuah abstrak dapat pula penulis kemukakan kebaruan (noveltis) dan orisinalitas dari penelitian. Abstrak juga harus dengan jelas menyatakan dan memuat sumbangan hasil penelitian terhadap khazanah ilmu pengetahuan.

Kata kunci: kata kunci 1, kata kunci 2, dst.

Catatan :

Penomoran halaman abstrak dan bagian persiapan tesis diberi nomor yang berbeda dengan nomor halaman tubuh utama tesis. Penomoran halaman bagian persiapan menggunakan angka romawi kecil, yaitu i, ii, iii, iv, …, x, xi, … untuk membedakan dari nomor halaman tubuh utama tesis yang berupa angka arab (1,2,3, … dst.). Istilah “1 spasi” merujuk pada satu kali (ketuk) tombol enter. Format/jenis huruf yang digunakan sama dengan jenis huruf baris sebelumnya, sedangkan ukuran/jarak masing-masing spasi ditunjukan dengan istilah “spasi 1” atau “spasi 1,5”.

*ABSTRACT*

***AUTOMATIC WANG TILES GENERATION USING PARAMETRIC APPROACH***

*By*

**Fairuz Astra Pratama**

**NIM: 23520012**

***(******Master’s Program in Informatics)***

*Pada bagian ini, abstrak ditulis dalam bahasa inggris. Abstrak ditulis dengan ukuran huruf 12, dicetak miring, dan tidak menggunakan huruftebal. Kalimat pertama abstrak tesis berjarak tiga spasi dari baris terakhir nama program studi. Isi abstract yang merupakan hasil terjemahan dari abstrak bahasa Indonesia di atas. Kalimat pertama atau awal paragraf-baru dipisahkan dengan jarak satu spasi dari kalimat terakhir paragraf yang mendahuluinya. Pada abstrak harus disertakan kata kunci (keywords) dengan maksimal tujuh kata atau kelompok kata. Peletakan kata kunci terpisah dari abstrak dengan jarak dua spasi. Kata kunci tersebut harus berhubungan dan berasal dari isi abstrak*.

Keywords: kata kunci 1, kata kunci 2, dst.

Catatan: Format catatan bukan bagian dari naskah tesis sehingga **harus dihapus (tidak disertakan dalam naskah).**.

1. Tesis harus ditik dengan komputer dalam **dua muka/sisi halaman kertas** (**bolak balik**). Oleh sebab itu, gunakanlah fasilitas ***mirror margins*** yang tersedia pada *Word Processor*, sehingga batas untuk **halaman ganjil** adalah **tepi kiri 4 cm** dan **tepi atas, tepi kanan,** serta **tepi bawah masing-masing 3 cm**, sedangkan untuk **halaman genap** berlaku **tepi kiri, tepi atas, dan tepi bawah berukuran 3 cm, sedangkan tepi kanan berukuran 4 cm.**

2. Penyimpangan dari butir 1 dapat dilakukan untuk dokumen tesis yang **belum** dijilid dengan sampul keras (*hard cover*) tebalnya **kurang** dari 1,5 cm (kurang dari ± 115 lembar).

3. Template ini merujuk pada butir 1, sedangkan template yang merujuk pada butir 2 dapat ditemukan di http://www.sps.itb.ac.id

**GENERASI *WANG TILES* OTOMATIS MENGGUNAKAN PENDEKATAN PARAMETRIK**

HALAMAN PENGESAHAN

Oleh

**Fairuz Astra Pratama**

**NIM: 23520012**

**(Program Studi Magister Teknik Informatika)**

Institut Teknologi Bandung

Menyetujui

Tim Pembimbing

Tanggal ………………………..

Ketua

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Nama Pembimbing 1)

Anggota Anggota

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Nama Pembimbing 2) (Nama Pembimbing 3)

PEDOMAN PENGGUNAAN TESIS

Tesis Magister yang tidak dipublikasikan terdaftar dan tersedia di Perpustakaan Institut Teknologi Bandung, dan terbuka untuk umum dengan ketentuan bahwa hak cipta ada pada penulis dengan mengikuti aturan HaKI yang berlaku di Institut Teknologi Bandung. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau peringkasan hanya dapat dilakukan seizin penulis dan harus disertai dengan kaidah ilmiah untuk menyebutkan sumbernya.

Sitasi hasil penelitian Tesis ini dapat di tulis dalam bahasa Indonesia sebagai berikut:

Nama Belakang, Inisial Nama Depan. (Tahun): *Judul tesis*, Tesis Program Magister, Institut Teknologi Bandung.

dan dalam bahasa Inggris sebagai berikut:

Nama Belakang, Inisial Nama Depan. (Tahun): *Judul tesis* *yang telah diterjemahkan dalam bahasa Inggris*, Master’s Thesis, Institut Teknologi Bandung.

Memperbanyak atau menerbitkan sebagian atau seluruh tesis haruslah seizin Dekan Sekolah Pascasarjana, Institut Teknologi Bandung.

Catatan: baris kedua yang merupakan kelanjutan dari baris pertama (satu judul buku), dimulai dengan 7 ketukan (satu Tab) atau rongak (*hanging indentation*: 1,27 cm) dari tepi halaman.

KATA PENGANTAR

Halaman kata pengantar dicetak pada halaman baru. Pada halaman ini mahasiswa S2 berkesempatan untuk menyatakan terima kasih secara tertulis kepada pembimbing dan perorangan lainnya yang telah memberi bimbingan, nasihat, saran dan kritik, serta kepada mereka yang telah membantu melakukan penelitian, kepada perorangan atau badan yang telah memberi bantuan pembiayaan, dan sebagainya.

Cara menulis kata pengantar beraneka ragam, tetapi semuanya hendaknya menggunakan kalimat yang baku. Ucapan terima kasih agar dibuat tidak berlebihan dan dibatasi hanya yang “*scientifically related*”.

DAFTAR ISI

[ABSTRAK 2](#_Toc87359865)

[*ABSTRACT* 3](#_Toc87359866)

[HALAMAN PENGESAHAN 4](#_Toc87359867)

[PEDOMAN PENGGUNAAN TESIS 5](#_Toc87359868)

[KATA PENGANTAR 6](#_Toc87359869)

[DAFTAR ISI 7](#_Toc87359870)

[DAFTAR GAMBAR DAN ILUSTRASI 8](#_Toc87359871)

[Bab I Pendahuluan 9](#_Toc87359872)

[I.1 Latar Belakang 9](#_Toc87359873)

[I.2 Hipotesis 10](#_Toc87359874)

[I.3 Rumusan Masalah 10](#_Toc87359875)

[I.4 Tujuan Penelitian 10](#_Toc87359876)

[I.5 Metodologi Penelitian 11](#_Toc87359877)

[I.6 Sistematika Tesis 11](#_Toc87359878)

[Bab II Studi Literatur 14](#_Toc87359879)

[II.1 *Wang Tile* 14](#_Toc87359880)

[II.1.1 Metode Pembuatan *Wang-Tile* 15](#_Toc87359881)

[II.1.2 *Strict* *Wang-Tile* 16](#_Toc87359882)

[II.2 *Generative Adversarial Network* (GAN) 19](#_Toc87359883)

[II.2.1 *Deep-Convolutional-GAN* 20](#_Toc87359884)

[II.2.2 *Spatial-GAN* 21](#_Toc87359885)

[II.2.3 *Periodic-Spatial-GAN* 23](#_Toc87359886)

[DAFTAR PUSTAKA 24](#_Toc87359887)

DAFTAR GAMBAR DAN ILUSTRASI

[Gambar II.1 Kelompok *Wang-Tiles* dan cara penyusunannya (Wei, 2005) 14](#_Toc87001592)

[Gambar II.2 Perbandingan ubin tekstur biasa dan *Wang-Tile* 15](#_Toc87001593)

[Gambar II.3. Proses pembuatan *Wang-Tile* yang diusulkan Cohen dkk. 16](#_Toc87001594)

[Gambar II.4. Ilustrasi *Corner Problem* (Zhang dan Kim, 2008) 17](#_Toc87001595)

[Gambar II.5. Ilustrasi Metode *Strict Wang-Tile* (Zhang dan Kim, 2008) 17](#_Toc87001596)

[Gambar II.6 Contoh hasil generasi *Wang-Tile* menggunakan metode *Strict Wang-Tiles*, terdapat beberapa *visual artifact* pada hasil 18](#_Toc87001597)

# Pendahuluan

## Latar Belakang

Dalam aplikasi grafis seperti simulasi atau permainan komputer, sering terdapat permukaan besar yang merepresentasikan dinding, lantai, atau tanah. Permukaan ini perlu diberikan tekstur / gambar tertentu agar terlihat seperti dinding batu bata, lantai kayu, atau permukaan lainnya. Menggunakan gambar yang sangat besar untuk memberikan tekstur ke permukaan dalam sebuah aplikasi tidaklah praktis, karena kumpulan gambar tersebut akan memerlukan ruang penyimpanan yang besar pada *harddisk, RAM,* dan memori kartu grafis (Wei, 2005). Salah satu pendekatan yang dapat digunakan untuk menyelesaikan permasalahan ini adalah ubin tekstur (*texture tiling*), di mana sekumpulan gambar / tekstur kecil akan disusun secara berulang untuk menutupi permukaan yang besar.

*Wang-Tiles* (Cohen dkk., 2003) adalah metode generasi dan penyusunan ubin tekstur yang dapat digunakan untuk mengisi tekstur permukaan secara tidak repetitif. Sebuah *Wang-Tile* adalah ubin tekstur kotak yang setiap sisinya akan diberikan tanda tertentu dalam bentuk warna. Dua ubin dapat diletakan bersebelahan jika sisi yang bersentuhan memiliki tanda warna yang sama, di mana tekstur pada kedua ubin akan dapat menyatu secara mulus. Dengan metode ini, sekelompok *Wang-Tile* dapat digunakan untuk memberikan tekstur ke permukaan seluas apa pun.

Kompleksitas utama metode ini terletak pada pembuatan sekelompok *Wang-Tile* untuk merepresentasikan tekstur yang diinginkan. Cohen dkk. (2003) mengusulkan cara untuk membuat sekelompok *Wang-Tile* dari sampel gambar tekstur menggunakan metode *Image Quilting* (Efros dan Freeman, 2001). Selain itu, juga ada banyak alternatif metode pembuatan *Wang-Tile* lain seperti *Strict Wang-Tiles* (Zhang dan Kim, 2008) dan *Diagonal Quilting Wang-Tiles* (Yueran dan Huichuan, 2013). Detail dari beberapa metode ini akan dibahas pada bab berikutnya.

Mayoritas pendekatan pembuatan *Wang-Tile* yang sudah ada seperti *Strict Wang-Tile* menggunakan pendekatan non-parametrik. Metode metode ini pada umumnya memiliki kelemahan saat memproses tekstur yang terstruktur / semi-terstruktur seperti dinding batu bata. *Periodic Spatial Generative Adversarial Network* (Bergmann dkk., 2017)adalah model parametrik yang dirancang untuk belajar dari sebuah gambar tekstur, di mana model akan dapat menghasilkan tekstur yang serupa dengan ukuran yang jauh lebih besar dari sampel. Metode ini juga telah terbukti bisa menghasilkan tekstur yang terstruktur seperti teks atau sarang lebah dengan kualitas yang cukup baik. Penggunaan pendekatan parametrik, seperti yang digunakan pada PS-GAN berpotensi untuk menghasilkan metode pembuatan *Wang-Tile*  dengan kualitas yang lebih baik lagi.

## Hipotesis

Pendekatan parametrik dapat digunakan untuk membuat *Wang-Tile* secara otomatis dari sampel tekstur, dan tekstur *Wang-Tile* yang dihasilkan oleh metode ini akan memiliki kualitas yang lebih bagus dari pendekatan non-parametrik seperti *Strict Wang-Tile*.

## Rumusan Masalah

Masalah yang akan dibahas dalam penelitian adalah sebagai berikut:

1. Bagaimana cara implementasi metode pembuatan *Wang-Tile* otomatis menggunakan pendekatan parametrik?
2. Bagaimana kualitas *Wang-Tile* yang dihasilkan oleh metode di atas jika dibandingkan dengan metode non-parametrik yang sudah ada?

## Tujuan Penelitian

1. Mengkaji metode generasi tekstur dengan pendekatan parametrik untuk mencari metode yang juga bisa digunakan untuk membuat tekstur *Wang-Tile* dengan baik.
2. Mengimplementasi sistem untuk pembuatan *Wang-Tile* dari sampel tekstur menggunakan pendekatan parametrik.
3. Melakukan analisis terhadap kualitas *Wang-Tile* yang dihasilkan dari metode parametrik dibandingkan dengan hasil dari metode non-parametrik untuk berbagai macam target tekstur.

## Metodologi Penelitian

1. Studi Pustaka

Analisis dan pembelajaran sumber literatur yang terkait. Beberapa topik yang akan dipelajari adalah *Wang-Tile*, beberapa metode pembuatan *Wang-Tile* secara otomatis, batasan / kelemahan dari metode pembuatan *Wang-*Tile yang ada, serta metode parametrik untuk generasi tekstur dari sampel tekstur.

1. Perancangan Sistem

Perancangan metode pembuatan *Wang-Tile* secara otomatis menggunakan pendekatan parametrik.

1. Implementasi dan Pelatihan Model

Implementasi dari metode yang telah dirancang sebelumnya. Proses pelatihan model untuk membuat *Wang-Tile* dari berbagai target tekstur juga akan dilakukan pada tahap ini.

1. Pengujian

Perbandingan kualitas *Wang-Tile* yang dihasilkan oleh model yang telah dilatih terhadap *Wang-Tile* yang dihasilkan oleh metode non-parametrik. Hasil dari perbandingan akan dianalisis dan digunakan untuk penarikan kesimpulan

## Sistematika Tesis

Bab I Pendahuluan berisi penjelasan singkat mengenai topik yang akan dibahas, hipotesis yang dibuat, serta tujuan dan metodologi penelitian yang akan dilakukan.

Bab II Studi Literatur akan berisi penjelasan lebih mendetail tentang konsep yang penting pada topik penelitian. Bab ini akan berisi penjelasan mengenai *Wang-Tile*, metode pembuatan *Wang-Tile* secara otomatis yang ada saat ini, serta metode generasi tekstur menggunakan pendekatan parametrik yang ada.

Bab III Rancangan Sistem berisi hasil analisis dan desain dari metode pembuatan *Wang-Tile* menggunakan pendekatan parametrik. Bab ini juga akan berisi beberapa contoh *Wang-Tile* yang dihasilkan menggunakan metode ini.

Bab IV Pengujian berisi perbandingan antara kualitas *Wang-Tile* yang dihasilkan oleh metode yang dirancang pada bab sebelumnya dengan metode non-parametrik yang sudah ada. Perbandingan kualitas akan difokuskan kepada banyaknya *visual artifact* pada tekstur yang dihasilkan.

Bab V Kesimpulan dan Saran, akan berisi kesimpulan dan saran yang ditemukan.

Bab pendahuluan sedikitnya memuat (dapat dirinci dalam bentuk anak bab) hal-hal berikut:

1. Deskripsi topik penelitian dan latar belakang;
2. Masalah penelitian (s*tatement of the problem*), tujuan, lingkup permasalahan, asumsi-asumsi yang digunakan, serta hipotesis;
3. Cara pendekatan dan metode penelitian yang digunakan serta diagram alir penelitian;
4. Pelaksanaan penelitian secara garis besar;
5. Sistematika (*outline*) tesis; Masalah yang hendak diselesaikan dalam tesis hendaknya dinyatakan dengan jelas, tegas, dan terinci mengingat sudah sangat menjurus dan runcingnya masalah tersebut dalam bidang spesialisasi kandidat magister.

Cara penulisan rujukan/pustaka pada kalimat yang disadur, adalah sebagai berikut:

rujukan yang ditulis 1 orang: “(Hill, 2007)”

rujukan yang ditulis lebih dari 2 orang. disertai no. halaman buku yang dirujuk: “(Baker dkk., 1998: 23-25)”

1 rujukan yang ditulis 2 orang: “(Gao dan Zhao, 2009)”

2 rujukan dengan jumlah penulis yang berbeda: “(Hill, 2007 dan Kramer dkk., 2005)”

2 rujukan yang masing-masing rujukan ditulis lebih dari 2 orang: “(Kramer dkk., 2005 dan Kumai dkk., 1985)”

2 rujukan yang masing-masing ditulis oleh penulis utama yang sama, namun diterbitkan pada waktu yang berbeda : “(Culver dkk., 2003a dan 2003b)”

Cara penulisan rujukan/pustaka pada kalimat secara jelas:“... hasil penelitian Wijaya (1996)“

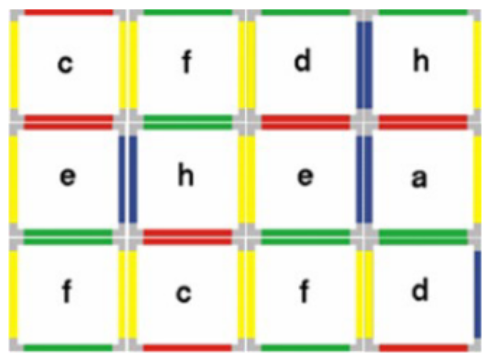
# Studi Literatur

## *Wang Tile*

Salah satu permasalahan dalam grafik komputer adalah bagaimana mengisi permukaan dengan tekstur yang sesuai seperti dinding, atau rumput. Penggunaan gambar yang besar akan memakan terlalu banyak memori, salah satu metode tradisional yang digunakan untuk mengatasi hal ini adalah menggunakan sampel tekstur kecil yang bisa disusun seperti ubin untuk menutupi permukaan seluas apa pun (Difrancesco dkk., 1993). Kelemahan dari metode ini adalah repetisi pola pada tekstur yang dihasilkan bisa terlihat dengan jelas, membuat tekstur terlihat tidak realistis. *Wang-Tile* bisa digunakan untuk mengatasi kelemahan ini.

*Wang-Tile* (Cohen dkk., 2003)adalah sekelompok sampel tekstur kotak di mana tiap sisi sampel akan diberikan “warna” tertentu. Penyusunan kelompok sampel tekstur yang valid adalah penyusunan di mana setiap sisi dua sampel yang bersentuhan, memiliki kode “warna” yang sama seperti yang dapat dilihat pada Gambar II.1. Apabila terdapat lebih dari satu sampel tekstur dari kelompok yang valid untuk bagian tertentu, sampel yang digunakan akan dipilih secara acak, menghasilkan pola yang tidak terlihat repetitif.





Gambar II.1 Kelompok *Wang-Tiles* dan cara penyusunannya (Wei, 2005)







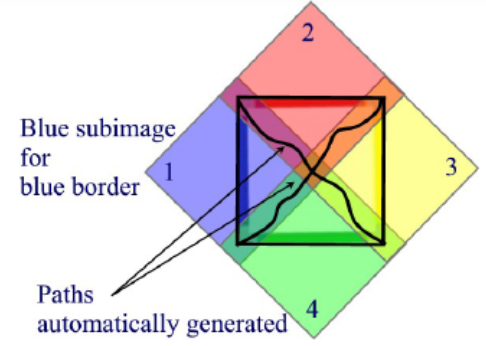
Gambar II.2 Perbandingan ubin tekstur biasa dan *Wang-Tile*

Pada Gambar II.2, kita bisa melihat perbedaan antara tekstur ubin tekstur biasa dan *Wang-Tile*. Gambar kiri adalah 9 tekstur dengan ukuran asli 512x512 piksel yang disusun ke dalam bentuk 3x3. Karena setiap ubin menggunakan tekstur yang sama, repetisi pola dapat terlihat pada hasil penyusunan. Gambar kanan tersusun dari 8 *Wang-Tile* berukuran 256x256 piksel yang dibuat menggunakan algoritma *Strict Wang-Tile*. Sekumpulan *Wang-Tile* ini disusun dalam bentuk 6x6, menghasilkan tekstur yang lebih susah dideteksi repetisi polanya. yang dibuat secara otomatis dari tekstur yang digunakan untuk pada gambar kanan

### Metode Pembuatan *Wang-Tile*

Kompleksitas utama metode *Wang-Tile* adalah pada proses pembuatan kelompok *Wang-Tile* yang dapat merepresentasikan tekstur tertentu. Tekstur pada setiap ubin harus bersifat kontinu terhadap semua kemungkinan ubin yang bisa diletakan disebelahnya. Selain itu, isi tekstur setiap ubin juga harus bervariasi, agar dapat menghasilkan tekstur akhir yang tidak repetitif.

Cohen dkk. mengusulkan metode untuk pembuatan *Wang-Tile* dari sampel tekstur secara otomatis yang dibuat berdasarkan metode *Image Quilting* (Efros dan Freeman, 2001). Pada metode ini, untuk setiap kemungkinan “warna”, akan diambil sampel tekstur berbentuk belah ketupat. Untuk membangun sebuah ubin *Wang-Tile*, empat sampel belah ketupat (belah ketupat yang sama bisa digunakan dua kali) akan digabungkan seperti pada Gambar II.3. Kotak hitam di tengah akan menjadi *Wang-Tile* dengan “warna” sisi sesuai warna belah ketupat pada sisi tersebut. Pada bagian belah ketupat yang bertumpang tindih, garis perbatasan yang akan menghasilkan gambar termulus akan dicari menggunakan *Minimum Error Boundary Cut* yang digunakan oleh Efros dan Freeman (2001).

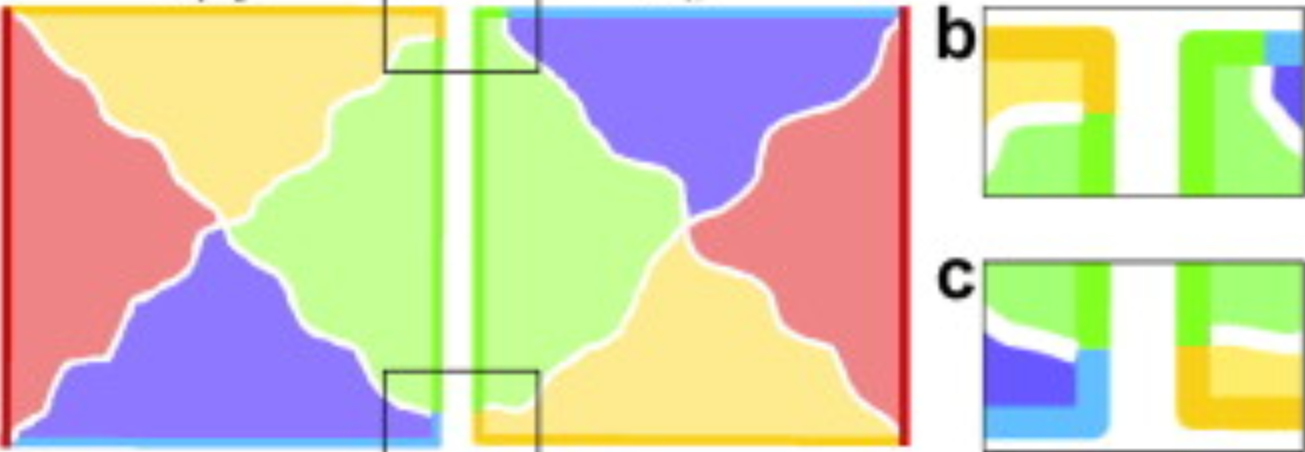


Gambar II.3. Proses pembuatan *Wang-Tile* yang diusulkan Cohen dkk.

Proses di atas kemudian akan diulangi untuk setiap kombinasi “warna” sisi *Wang-Tile* yang diinginkan. Karena belah ketupat yang sama akan digunakan untuk setiap ubin yang memiliki warna sisi yang sama, maka dua ubin tekstur tersebut akan dapat disusun secara bersebelahan. Sebagai contoh, apabila kita membuat ubin kedua yang menggunakan belah ketupat kuning di sisi kanan, maka ubin tekstur tersebut akan dapat diletakan di sebelah kanan ubin tekstur pada Gambar II.3.

### *Strict* *Wang-Tile*

Metode pembuatan *Wang-Tile* yang diusulkan oleh Cohen dkk. memiliki beberapa kelemahan. Permasalahan pertama adalah garis potong sampel belah ketupat yang menghasilkan gambar termulus tidak pasti akan menyentuh ujung ubin tekstur. Hal ini dapat menyebabkan sisi ubin *Wang-Tile* disekitar ujung ubin memiliki “warna” yang salah (Zhang dan Kim, 2008). Fenomena ini sering disebut sebagai *Corner Problem*, dan contoh ilustrasi dapat dilihat pada Gambar II.4. Selain itu, karena proses pencarian titik potong hanya dapat dilakukan pada bagian kecil tekstur yang tertumpang tindih, kemungkinan algoritma *Image Quilting* gagal menemukan garis potong yang bagus akan lebih tinggi, menghasilkan tekstur ubin yang tidak mulus.



Gambar II.4. Ilustrasi *Corner Problem* (Zhang dan Kim, 2008)

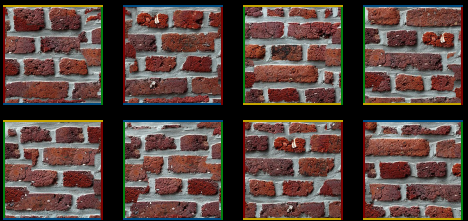
*Strict Wang-Tile* adalah metode alternatif untuk membuat *Wang-Tile* buatan Zhang dan Kim (2008) dengan tujuan mengatasi kelemahan metode yang diusulkan Cohen dkk. Pada metode ini, untuk setiap kemungkinan “warna”, akan diambil sampel tekstur berbentuk belah ketupat juga. Untuk membangun ubin tekstur, mirip seperti metode Cohen dkk., empat belah ketupat akan digabung dimana persegi pada tengah gabungan belah ketupat akan menjadi ubin tekstur dengan “warna” sisi sesuai warna belah ketupat pada sisi tersebut. Berbeda dengan metode Cohen dkk., tidak akan ada bagian dari belah ketupat yang bertumpang tindih, dan ujung dari persegi ubin akan menyentuh sisi dari gabungan belah ketupat, seperti yang dapat dilihat pada Gambar II.5.

Gambar II.5. Ilustrasi Metode *Strict Wang-Tile* (Zhang dan Kim, 2008)

Untuk membuat bagian tengah ubin tekstur menjadi tekstur yang mulus, sampel tekstur lain berbentuk kotak akan diambil secara acak dan diletakan diatas kandidat ubin tekstur. Dengan mencari perbatasan / garis potong optimal antara gabungan belah ketupat dan sampel tekstur kotak yang baru ini, kita bisa menutupi perbatasan belah ketupat yang tidak mulus seperti yang dapat dilihat pada Gambar II.5. Garis potong optimal ini ditentukan secara otomatis menggunakan algoritma *Graph-Cut*. Karena daerah yang tertumpang tindih merupakan keseluruhan ubin tekstur, daerah untuk mencari garis potong yang bagus akan jauh lebih besar dibandingkan dengan metode yang diusulkan oleh Cohen dkk.

Walau secara umum lebih baik dari metode yang diusulkan Cohen dkk., *Strict Wang-Tile* masih memiliki kelemahan. Karena proses pemilihan sampel tekstur belah ketupat dan tekstur kotak tambahan dilakukan secara acak, algoritma *Graph-Cut* mungkin tidak dapat menemukan garis potong yang bagus. Selain itu, apabila tekstur memiliki struktur global atau pola yang hampir teratur, *Wang-Tile* yang dihasilkan sistem kemungkinan besar akan gagal menjaga pola tersebut. Hal ini bisa dilihat saat mencoba membuat *Wang-Tile* dari tekstur dinding batu bata yang semi-terstruktur seperti yang terlihat pada Gambar II.6

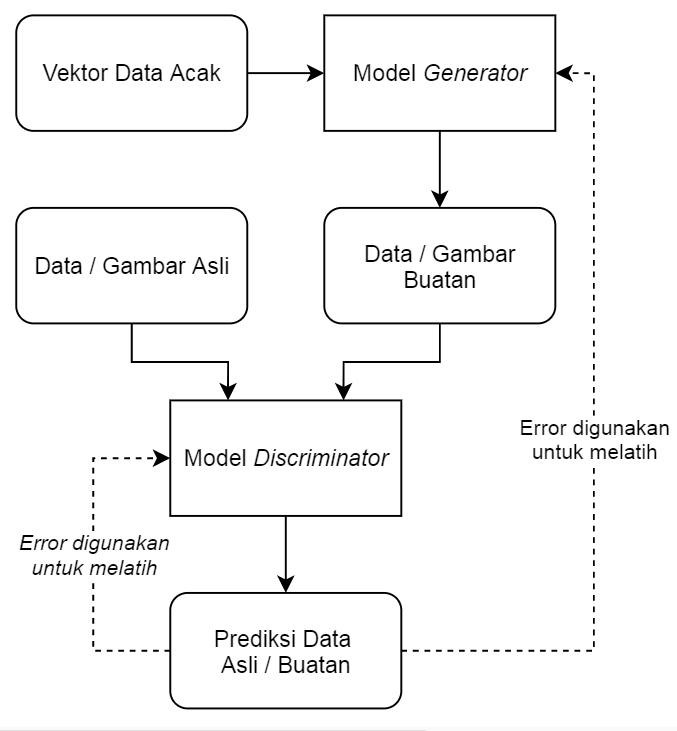


Gambar II.6 Contoh hasil generasi *Wang-Tile* menggunakan metode *Strict Wang-Tiles*, terdapat beberapa *visual artifact* pada hasil

## *Generative Adversarial Network* (GAN)

Dalam pembelajaran mesin, salah satu contoh kasus / jenis permasalahan yang dihadapi adalah pemodelan generatif. Pada permasalahan ini, terdapat sekumpulan data yang mengikuti distribusi tertentu (e.g. kumpulan foto anjing, kumpulan suara burung) dan dibutuhkan sebuah model yang dapat menghasilkan data baru yang terlihat telah berasal dari distribusi yang sama. *Generative Adversarial Network* (Goodfellow dkk., 2014) adalah salah satu metode untuk mengimplementasi model generatif berbasis Jaringan Saraf Tiruan.

GAN terdiri dari dua model Jaringan Saraf Tiruan (*Artificial Neural Network*) yang dilatih melawan satu sama lain. Model G (*Generator*) akan menerima masukan vektor data acak, dan dilatih untuk mengeluarkan data yang mirip dengan data target. Model D (Diskriminator) akan menerima masukan data asli (e.g. foto anjing) dan data yang dihasilkan oleh Model G; Diskriminator kemudian akan dilatih untuk mengeluarkan angka 1 jika data masukan merupakan data asli, dan 0 jika data dihasilkan oleh model D. Ilustrasi dari model ini dapat dilihat pada Gambar II.7.



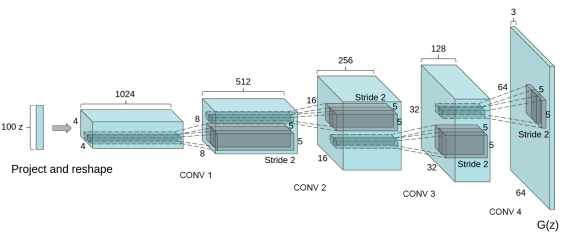
Gambar II.7. Ilustrasi Model *Generative Adversarial Network*

Pelatihan kedua model dilakukan secara bergantian menggunakan *Stochastic Gradient Descent* atau metode latih lainnya. Model D akan dilatih untuk meminimalisir log(D(x)) + log(1-D(G(z))), di mana D(x) adalah keluaran model untuk data asli dan log(1-D(G(z))) adalah keluaran model untuk data yang dihasilkan model G. Di sisi lain, Model G akan dilatih untuk memaksimalkan log(D(x)) + log(1-D(G(z))). Karena Model G tidak bisa mempengaruhi log(D(x)), maka Model G akan berusaha meminimalkan log(1-D(G(z))). Apabila setelah proses pelatihan akurasi model D stabil di sekitar 0.5, maka model G bisa dikatakan telah menghasilkan data yang tidak dapat dibedakan dari data asli.

### *Deep-Convolutional-GAN*

Untuk implementasi GAN terhadap data gambar, *Deep-Convolutional-GAN* (Radford dkk., 2015)merupakan salah satu arsitektur GAN yang terbukti dapat menghasilkan gambar dengan kualitas yang baik. Pada DCGAN, model *Generator* dan Diskriminator diimplementasikan menggunakan CNN (*Convolutional Neural Network*) yang sering digunakan dalam permasalahan klasifikasi gambar. Selain itu, DCGAN menambahkan beberapa restriksi yang bertujuan meningkatkan stabilitas model dalam proses pelatihan sebagai berikut:

* Gunakan *strided-convolution* pada model Diskriminator sebagai pengganti *Pooling Layer*
* Gunakan fungsi aktivasi LeakyReLU pada semua *layer* di Diskriminator kecuali *layer* terakhir yang menggunakan fungsi aktivasi Sigmoid
* Gunakan *fractional-strided-convolution* pada Generator untuk menghasilkan *layer* dengan ukuran keluaran yang lebih besar dari data masukan
* Gunakan fungsi aktivasi ReLU pada semua *layer* di model Generator kecuali *layer* terakhir yang menggunakan fungsi aktivasi Tanh
* Hindari penggunaan *fully-connected-layer* kecuali pada awal model Generator atau pada bagian akhir Diskriminator
* Gunakan *Batch Normalization* pada setiap *layer*, mengubah data masukan sehingga memiliki rata rata 0 dan varians 1



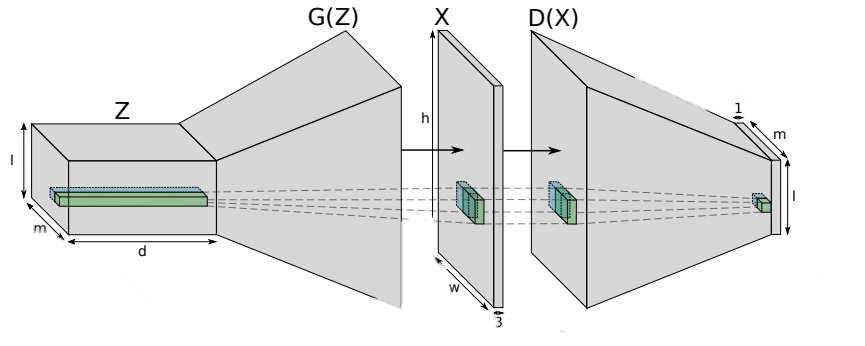
Gambar II.8. Ilustrasi Generator DCGAN (Radford dkk., 2015)

Pada Gambar II.8, dapat dilihat contoh model Generator yang mengikuti arsitektur DCGAN. Pada *layer* awal, vektor data acak dipetakan menjadi matriks tiga dimensi yang akan digunakan sebagai masukan *fractionally-strided convolution layer* di depannya. Setiap *layer* memperbesar tinggi dan lebar data sebesar dua kali lipat, dan *layer* terakhir akan menghasilkan representasi gambar buatan model. Pada model Diskriminator, model ini akan dibalik, dimulai dari data dengan panjang dan lebar yang besar, setiap *convolution layer* akan memperkecil ukuran data secara bertahap hingga *layer* terakhir yang menghasilkan data berukuran 1x1x1 sebagai hasil prediksi apakah data masukan merupakan data asli atau data buatan.

### *Spatial-GAN*

*Spatial-GAN* adalah arsitektur GAN yang dapat digunakan untuk generasi tekstur, di mana model akan dilatih menggunakan sampel tekstur dan akan dapat mengeluarkan tekstur yang serupa dengan ukuran yang jauh lebih besar. Arsitektur SGAN adalah arsitektur DCGAN dengan beberapa modifikasi sebagai berikut:

* Gunakan matriks / tensor 3-dimensi acak sebagai masukan dari Generator, berbeda dengan GAN pada umumnya yang menggunakan vektor 1-dimensi
* Tidak ada *fully-connected-layer* pada Generatordan Diskriminator, model hanya akan terdiri dari *Convolution Layer*
  + Diskriminator akan menghasilkan matriks 2-dimensi, isi setiap sel pada matriks akan dilatih dengan target yang sama seperti GAN pada umumnya (1 jika menerima data asli, dan 0 jika data yang dihasilkan Generator)



Gambar II.9 Ilustrasi Model SGAN (Jetchev dkk., 2016)

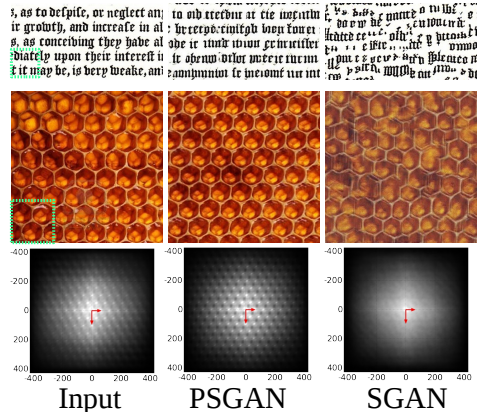
Pada Gambar II.9, dapat dilihat arsitektur umum dari model SGAN. Dapat dilihat bahwa vektor data acak telah digantikan dengan tensor / matriks 3-dimensi yang berisi data acak. Hasil akhir Diskriminator juga tidak berupa satu angka, namun berupa matriks 2-dimensi. Selain itu, mayoritas arsitektur menggunakan desain yang sama seperti DCGAN, menggunakan beberapa lapis *fractionally-strided convolution layer* pada Generator dan *convolution layer* pada Diskriminator.

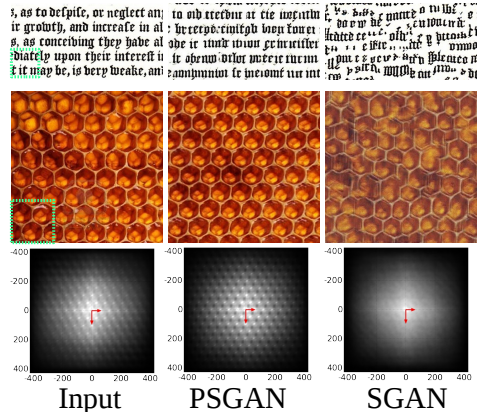
Selama proses pelatihan, data latih yang digunakan adalah sampel acak dari target tekstur yang sama, Panjang dan lebar matriks data acak akan diatur sedemikian rupa sehingga hasil keluaran Generator (dan gambar yang diterima Diskriminator) lebih kecil dari ukuran tekstur yang dimiliki. Pada umumnya kita ingin ukuran sampel tekstur cukup kecil hingga model bisa melihat variasi dalam data latih, namun cukup besar sehingga model bisa melihat struktur yang ada dalam tekstur. Setelah proses pelatihan selesai, model Generator bisa digunakan dengan masukan matriks data acak dengan panjang dan lebar yang bebas, menghasilkan gambar berukuran bebas yang mirip dengan tekstur data latih.

### *Periodic-Spatial-GAN*

Salah satu kelemahan SGAN adalah karena hanya terdiri dari *convolution layer*, SGAN akan kesulitan membuat model dari tekstur yang memiliki struktur global, seperti sarang lebah atau dinding batu bata. Karena setiap *perceptron* pada setiap *layer* hanya akan terhubung ke beberapa keluaran pada *layer* sebelumnya, hasil tekstur pada bagian tertentu akan bersifat independen terhadap tekstur bagian lain yang jauh. *Periodic-Spatial-GAN* (Bergmann dkk., 2017) dapat digunakan untuk mengatasi hal ini.

PSGAN melakukan beberapa perubahan terhadap matriks / tensor 3-dimensi acak yang digunakan pada SGAN untuk membuat model dapat memproses tekstur terstruktur dengan baik. Selain itu, PSGAN juga membuat beberapa perubahan untuk membuat model dapat melakukan ekstraksi dan interpolasi tekstur. Pembahasan pada thesis ini hanya akan fokus pada perubahan yang relevan dalam membuat model dapat memproses tekstur dengan struktur global.





DAFTAR PUSTAKA

Bergmann, U., Jetchev, N., and Vollgraf, R. (2017): Learning Texture Manifolds with the Periodic Spatial GAN, *34th International Conference on Machine Learning, ICML 2017*, retrieved October 12, 2021from internet: https://arxiv.org/abs/1705.06566v2, **1**, 722–730.

Cohen, M. F., Shade, J., Hiller, S., and Deussen, O. (2003): Wang Tiles for image and texture generation, *ACM SIGGRAPH 2003 Papers, SIGGRAPH ’03*, 287–294. https://doi.org/10.1145/1201775.882265

Difrancesco, D., Francisco, S., Assignee, C., Pixar, :, and Richmond, C. (1993): METHOD FOR BORDERLESS MAPPNG OF TEXTURE IMAGES, **130**, 134.

Efros, A. A., and Freeman, W. T. (2001): Image quilting for texture synthesis and transfer, *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2001*, 341–346. https://doi.org/10.1145/383259.383296

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014): Generative Adversarial Networks, *Communications of the ACM*, retrieved November 5, 2021from internet: https://arxiv.org/abs/1406.2661v1, **63**(11), 139–144.

Jetchev, N., Bergmann, U., and Vollgraf, R. (2016): Texture Synthesis with Spatial Generative Adversarial Networks, retrieved October 12, 2021from internet: https://arxiv.org/abs/1611.08207v4.

Radford, A., Metz, L., and Chintala, S. (2015): Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*, retrieved November 5, 2021from internet: https://arxiv.org/abs/1511.06434v2.

Wei, L.-Y. (2005): Chapter 12. Tile-Based Texture Mapping | NVIDIA Developer, retrieved October 12, 2021, from internet: https://developer.nvidia.com/gpugems/gpugems2/part-ii-shading-lighting-and-shadows/chapter-12-tile-based-texture-mapping.

Yueran, Z., and Huichuan, D. (2013): Texture synthesis using diagonal quilting wang tiles, *Proceedings - 9th International Conference on Computational Intelligence and Security, CIS 2013*, 836–840. https://doi.org/10.1109/CIS.2013.182

Zhang, X., and Kim, Y. J. (2008): Efficient texture synthesis using strict Wang Tiles, *Graphical Models*, **70**(3), 43–56. https://doi.org/10.1016/J.GMOD.2007.10.002