

PENGEDITAN DETIL KURVA DENGAN METODE CURVE ANALOGIES MENGGUNAKAN PUSTAKA KURVA MULTIREOLUSI

Nanik Suciati, Rizky Yuniar Hakkun

Jurusan Teknik Informatika,

Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember

Kampus ITS, Jl. Raya ITS, Sukolilo – Surabaya 60111, Telp. + 62 31 5939214, Fax. + 62 31 5913804

Email : nanik@its-sby.edu, ryu_q2000@yahoo.com

ABSTRAK

Pembuatan gambar dua dimensi membutuhkan suatu perangkat lunak yang mendukung. Seringkali perangkat lunak tersebut membutuhkan kemampuan pengguna dalam hal menggambar, sehingga sulit bagi pengguna biasa untuk menghasilkan suatu bentuk sesuai keinginan. Penelitian ini membuat suatu perangkat lunak editor gambar yang dapat memudahkan pengguna untuk membentuk suatu gambar baru. Gambar ini direpresentasikan ke dalam bentuk kurva. Perangkat lunak dapat membuat kurva contoh dan membentuk kurva baru mirip dengan kurva contoh yang diberikan pada suatu bentuk dasar.

Metode pembentukan kurva baru dari kurva contoh yang diberikan menggunakan suatu metode yang dinamakan metode Curve Analogies. Metode Curve Analogies merupakan metode pembentukan kurva baru dari kurva contoh yang diberikan dengan mengikuti suatu bentuk tertentu, selain itu kurva yang dihasilkan mirip dengan kurva contoh. Kurva contoh yang diberikan merupakan kurva B-Spline Kubik multiresolusi sehingga dengan representasi multiresolusi akan mendapatkan beberapa bentuk kurva yang berbeda. Kurva B-Spline yang digunakan merupakan kurva B-Spline kubik sedangkan kurva polylines digunakan sebagai bentuk dasar dalam pembentukan kurva baru. Perangkat lunak dibuat dengan menggunakan bahasa pemrograman Borland Delphi 5 pada sistem operasi Windows XP.

Berdasarkan ujicoba yang dilakukan, perangkat lunak dapat bekerja dengan baik.. Perangkat lunak mampu membentuk pustaka kurva dan membuat kurva baru yang mirip dengan kurva contoh yang diberikan. Perangkat lunak mampu mensintesa kurva contoh pada bentuk dasar yang diberikan. Perangkat lunak juga mampu melakukan representasi multiresolusi pada kurva B-Spline yang dibuat.

Kata Kunci : Curve Analogies, B-Spline Curve, Multiresolution Curve.

1. PENDAHULUAN

Sering kali sulit mendesain sebuah gambar atau kurva yang membutuhkan banyak lekukan dan gerigi maupun untuk membuat berbagai karakter tulisan dengan berbagai macam gaya. Aplikasi pengolah gambar seperti editor gambar pada Microsoft Paint hanya dapat membentuk kurva secara manual. Aplikasi tersebut belum dapat membentuk kurva secara otomatis dari beberapa kurva contoh yang diberikan, sehingga penggambaran kurva membutuhkan waktu yang cukup lama dan seringkali tidak sesuai dengan yang diinginkan. Aplikasi tersebut memungkinkan untuk membentuk kurva yang panjang dan kontinu, tetapi akan menghabiskan waktu dan membutuhkan ketelatenan dalam menggambar.

Pembentukan kurva baru dari kurva contoh merupakan pekerjaan yang sulit karena kurva baru yang dibentuk harus sesuai dengan keinginan. Terdapat beberapa metode yang digunakan dalam menggambar kurva baru, tetapi masih memiliki

beberapa kelemahan. Sperti misal metode yang digunakan Adam Finkelstein *et al* [3] dan aplikasi yang dikembangkan oleh Nanik Suciati [10] merupakan metode pembentukan kurva yang sangat baik, tetapi bentuk kurva dapat berubah apabila bentuk *sweep* kurva baru sangat berbeda dari bentuk awal. Kurva yang dihasilkan tidak lagi mirip dengan kurva awal yang dibentuk sehingga untuk mengembalikan bentuk kurva baru yang mirip dengan kurva awal dibutuhkan inputan detail – detail kurva berupa matriks. Hal ini menyebabkan pengguna akan kesulitan dalam membentuk kurva yang diinginkan.

2. REPRESENTASI MULTIREOLUSI

Representasi multiresolusi menggunakan fungsi berbasis *wavelet* untuk kurva B-Spline Kubik yang menginterpolasi titik – titik ujung. Pembentukan suatu kurva ditentukan oleh titik – titik kontrolnya, dimana dalam representasi multiresolusi, titik – titik kontrol dari kurva B-Spline dimanipulasi sedemikian

rupa dengan memanfaatkan fungsi berbasis *wavelet* sehingga membentuk hirarki titik kontrol.

Dalam representasi multiresolusi, akan disusun ruang B-Spline kubik seragam S^j dengan deretan knot berjarak 2^{-j} , dengan j menunjukkan tingkat resolusi dari kurva. Suatu kurva dengan tingkat resolusi j dibentuk dengan menggunakan $2^j + 3$ titik kontrol. Pada setiap resolusi, kurva yang sama dinyatakan dalam titik – titik kontrol yang berbeda.

Misalkan terdapat himpunan C^n yang berisi titik kontrol pada resolusi n yang dinyatakan dalam vektor kolom $\begin{bmatrix} c_0^n \\ c_1^n \\ c_2^n \\ \vdots \\ c_{m-1}^n \end{bmatrix}$ dan memiliki jumlah titik kontrol $m = 2^{n-1} + 3$. Untuk menyusun C^{n-1} , dimana memiliki resolusi yang lebih rendah yaitu $n-1$, yang memiliki jumlah titik kontrol $2^{n-1} + 3 = m'$ dengan $m' < m$ dilakukan *filtering* terhadap m titik kontrol dari C^n . Proses ini dilakukan dengan menggunakan persamaan : $C^{n-1} = A^n C^n$ dengan A^n adalah matriks berukuran $m' \times m$. Proses ini dinamakan proses dekomposisi. Untuk menyimpan detail yang hilang pada proses pemfilteran maka detail tersebut ditampung dalam D^{n-1} yang dihitung dengan persamaan matriks : $D^{n-1} = B^n C^n$ dengan B^n adalah matriks berukuran $(m - m') \times m$. Proses rekonstruksi C^n dari C^{n-1} dan D^{n-1} dapat dilakukan dengan menggunakan pasangan matriks P^n dan Q^n melalui persamaan : $C^n = P^n C^{n-1} + Q^n D^{n-1}$.

3. CURVE ANALOGIES

Algoritma sintesa kurva dimulai dengan menginisialisasi $B'(t)$ sebagai kurva kosong, kemudian mengambil secara acak segmen dari tiga titik kontrol berurutan pada kurva A' ke kurva B' . Kurva B' selanjutnya dibentuk dengan pengambilan contoh tetangga terdekat. Pembentukan kurva dimulai pada akhir kurva, pada $t_{new} = t_{max} + Dt$, dengan t_{max} merupakan nilai t maksimum dari B' dan Dt merupakan jarak antar parameter. Selanjutnya adalah memilih nilai $B'(t_{new})$ yang meminimalkan persamaan $E(B') = \min_i d(B', t_i, A', s_j)$. Masing-masing tetangga dari titik kontrol direpresentasikan sebagai set dari K contoh $\mathbf{a}'_k \circ A'(s_k)$ dan $\mathbf{b}'_k \circ B'(t_k)$, dengan $k = \{1..K\}$, yang diambil pada unit panjang busur meningkat sekitar $A'(s_j)$ dan $B'(t_i)$ berturut-turut. Untuk mendapatkan kehalusan kurva

digunakan fitur tangen yaitu $D\mathbf{a}'_k \circ \frac{(\mathbf{a}'_k - \mathbf{a}'_{k-1})}{\|\mathbf{a}'_k - \mathbf{a}'_{k-1}\|}$

dan $D\mathbf{b}'_k \circ \frac{(\mathbf{b}'_k - \mathbf{b}'_{k-1})}{\|\mathbf{b}'_k - \mathbf{b}'_{k-1}\|}$. Posisi titik-titik tetangga

dari kurva A' dan B' dibandingkan dengan menggunakan transformasi rigid untuk mendapatkan matriks rotasi dan translasi. Algoritma kemudian mencari nilai \mathbf{p}^* untuk $B'(t_{new})$ sehingga tetangga sekitar titik tersebut sesuai dengan beberapa tetangga j^* pada kurva A' . Untuk mencari nilai tersebut, algoritma akan menghitung setiap tetangga (dengan indeks j) pada kurva contoh A' . Untuk setiap j , algoritma mencari nilai kandidat \mathbf{p}_j untuk $B'(t_{new})$

yang meminimalkan nilai $d(B', t_{new}, A', s_j)$ dan biaya yang bersesuaian $d_j = \arg \min_{B'(t_{new})} d(B', t_{new}, A', s_j)$.

Nilai dari satuan jarak yang akan dihitung adalah :

$$d(B', t_i, A', s_j) = \min_{R, t} \sum_k w_k (\|R' \mathbf{a}'_k + \mathbf{t}' - \mathbf{b}'_k\|^2 + w_D \|R' D\mathbf{a}'_k - D\mathbf{b}'_k\|^2) \mathbf{V}$$

ariabel j^* menyimpan indeks dari tetangga yang memiliki biaya terkecil d_j dan \mathbf{p}^* menyimpan indeks dari posisi \mathbf{p}_j yang bersesuaian. Pada akhirnya, nilai $B'(t_{new})$ merupakan nilai dari \mathbf{p}^* sedangkan indeks sumber $s(i) = j^*$. Pemilihan lokasi sumber j^* dilakukan secara acak dari beberapa nilai biaya yang masih lebih kecil dari nilai biaya terkecil dikalikan dengan suatu error yang dinotasikan sebagai $d_j(1 + e)$, dengan e merupakan toleransi error yang diperbolehkan.

Metode pencarian nilai \mathbf{p}_j dan d_j dengan membandingkan tetangga pada kedua kurva menggunakan transformasi rigid. Algoritma mencari transformasi rotasi R'_j dan translasi \mathbf{t}'_j yang optimal dari $\{\mathbf{a}'_k, D\mathbf{a}'_k\}$ ke $\{\mathbf{b}'_k, D\mathbf{b}'_k\}$ dengan menggunakan metode pencocokan titik Berthold K.P. Horn [7] dan jika hanya variasi translasi saja yang terjadi maka R'_j merupakan matriks identitas.

Nilai optimal \mathbf{p}_j didapatkan dengan mentransformasi posisi $A'(s_j)$ pada tetangga B' dan

$$\text{dapat dirumuskan sebagai } \mathbf{p}_j = (R'_j A'(s_j) + \mathbf{t}'_j).$$

Nilai d_j dihitung berdasar hasil dari \mathbf{p}_j beserta koherenitas dari hasil tersebut jika ada. Hasil kurva tersebut dikatakan koheren bila $s(i-1) < s(i)$ dan

$$\|A'(s(i)) - A'(s(i-1))\| < \frac{3}{2} \|B'(t_i) - B'(t_{i-1})\|, \text{ dengan nilai } 3/2$$

merupakan nilai batas toleransi yang diperbolehkan.

Sintesa kurva dengan memberikan suatu nilai *constraint*, yaitu *soft constraint* maka nilai titik

$$\text{kandidat adalah } \mathbf{p}_j = \frac{w_{\text{new}} (R'_j A'(s_j) + \mathbf{t}'_j) + w_c \mathbf{q}_c}{w_{\text{new}} + w_c}$$

Bila nilai w_c sangat besar maka *constraint* mendekati *hard constraint*.

Proses sintesa analog kurva dilakukan dengan menghitung transformasi pada masing – masing *sweep* kurva contoh dan sketsa. Hasil dari transformasi diterapkan pada sintesa kurva.

4. DESKRIPSI SISTEM

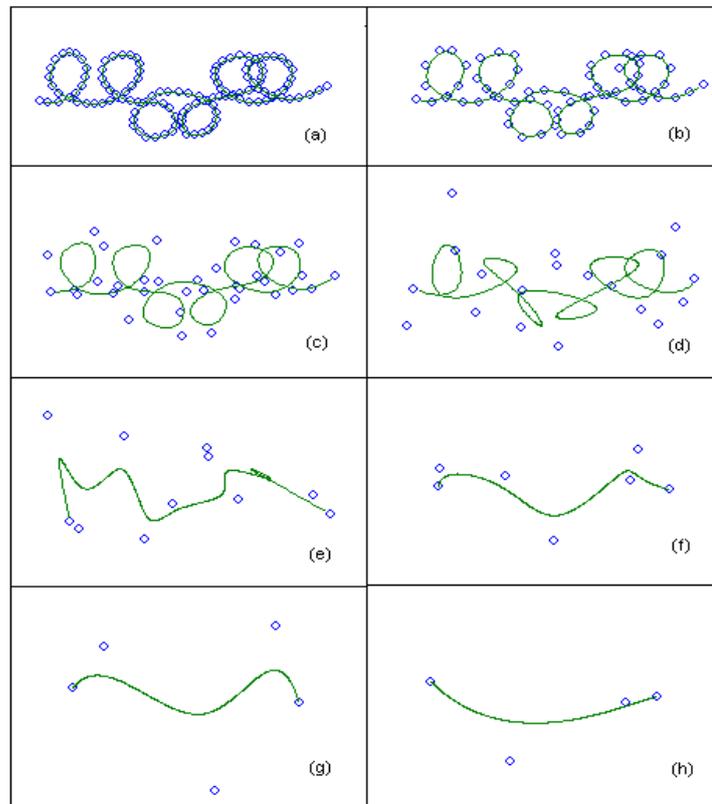
Sistem memiliki dua proses utama yaitu pembentukan pustaka kurva dan pembentukan kurva baru dengan metode *curve analogies*. Proses pembentukan pustaka dilakukan dengan membuat suatu kurva contoh. Kurva yang dibentuk merupakan kurva *B-Spline Kubik* multiresolusi sehingga dengan representasi multiresolusi akan didapatkan beberapa kurva dengan resolusi yang berbeda. Kurva contoh yang telah dibuat tersebut disimpan dalam pustaka kurva untuk digunakan dalam proses pembentukan kurva baru.

Proses pembentukan kurva baru dari kurva contoh dimulai dengan membentuk gambar kasar atau sketsa pada aplikasi. Pada sketsa yang telah dibentuk dapat diterapkan satu kurva contoh dari pustaka kurva untuk membentuk suatu kurva baru yang mirip dengan kurva contoh yang diberikan dan kurva baru yang dibentuk mengikuti pola dari sketsa yang dibuat. Proses pembentukan kurva baru ini menggunakan metode *curve analogies* yang telah dijelaskan sebelumnya.

5. UJI COBA

Uji coba yang akan dilakukan terdiri dari uji representasi multiresolusi, uji sintesa kurva dan uji sintesa analog kurva. Ujicoba dilakukan dengan menggunakan kurva contoh multiresolusi pada pustaka kurva yang telah dibuat.

Dalam uji representasi multiresolusi ini digunakan kurva B-Spline Kubik yang telah dibuat dan disimpan dalam pustaka kurva. Kurva contoh yang digunakan memiliki resolusi maksimum 7. Hasil representasi multiresolusi salah satu kurva contoh dapat dilihat pada Gambar 1.

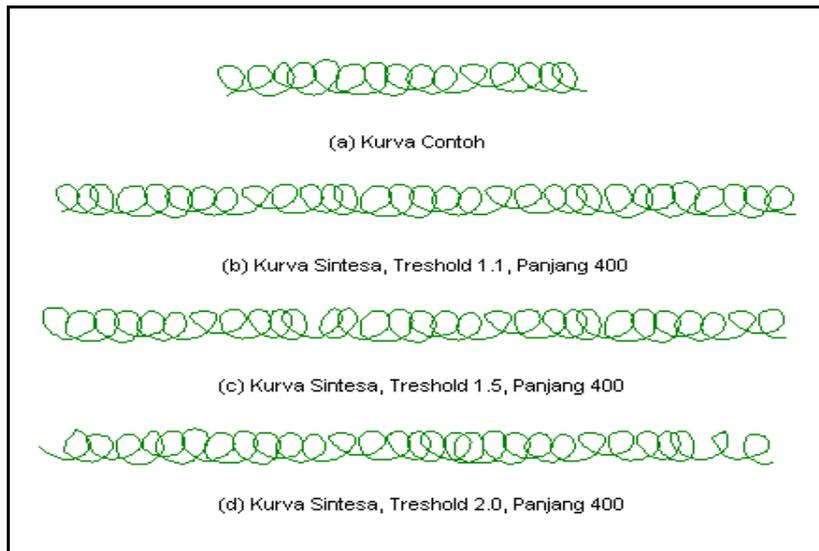


Gambar 1. Uji proses dekomposisi dan rekonstruksi

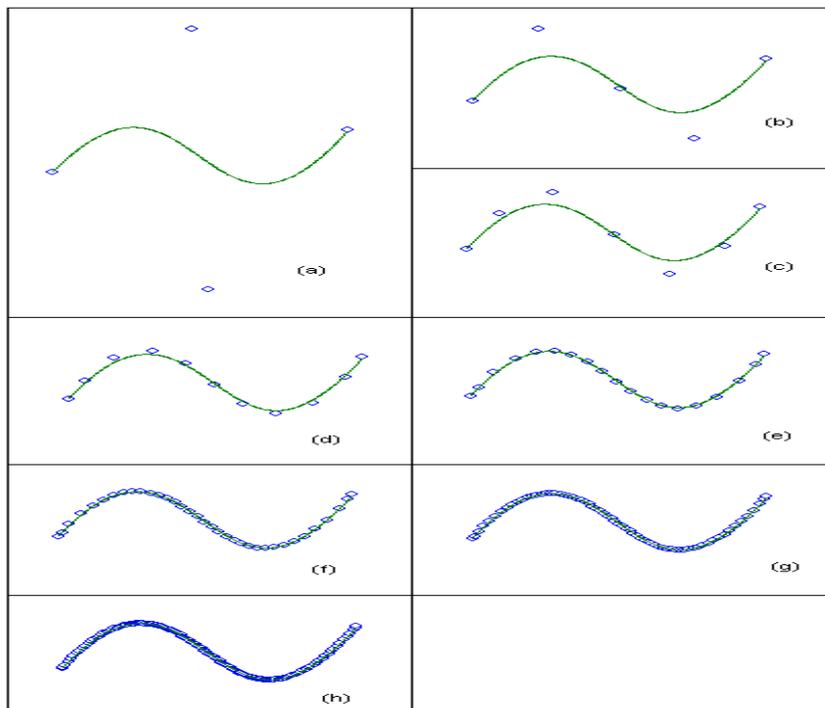
Representasi Multiresolusi dapat digunakan untuk melakukan penghalusan kontinu kurva. Hasil uji coba penghalusan kurva dapat dilihat pada Gambar 2 dengan kurva awal berada pada resolusi 0 (Gambar 2(a)), kemudian dengan menghilangkan detail kurva dilakukan penghalusan kurva secara kontinu sampai pada resolusi 7 (Gambar 2(h)). Jumlah titik kontrol pada kurva semakin banyak pada

tingkat resolusi yang lebih tinggi dengan bentuk kurva yang tetap.

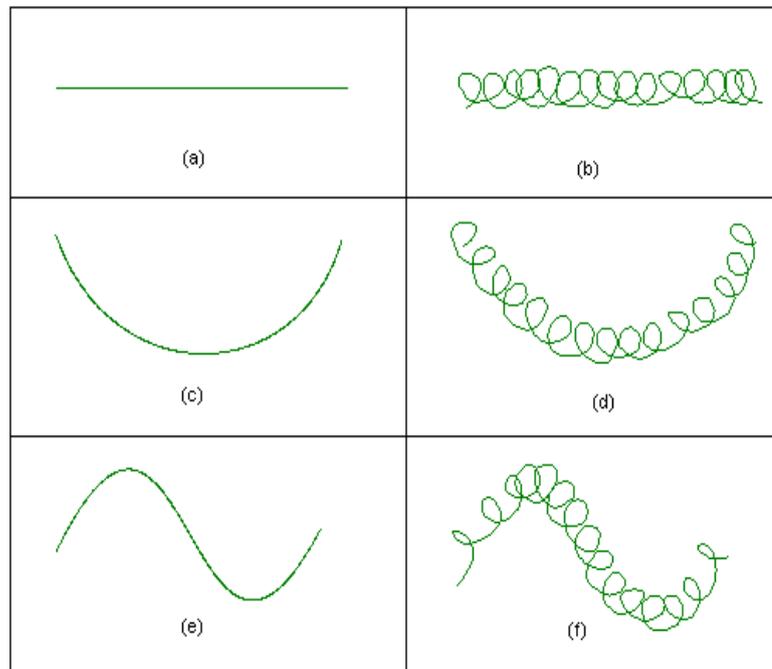
Hasil uji coba pengeditan langsung pada titik kontrol dengan mempertahankan detail kurva dapat dilihat pada Gambar 3. Uji sintesa kurva dilakukan dengan memberikan beberapa nilai *threshold* yang berbeda. Gambar 4 merupakan hasil sintesa pada *threshold* 1.1, 1.5 dan 2.0.



Gambar 2. Penghalusan kontinu

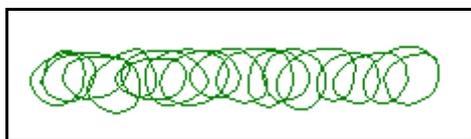


Gambar 3. Pengeditan kurva

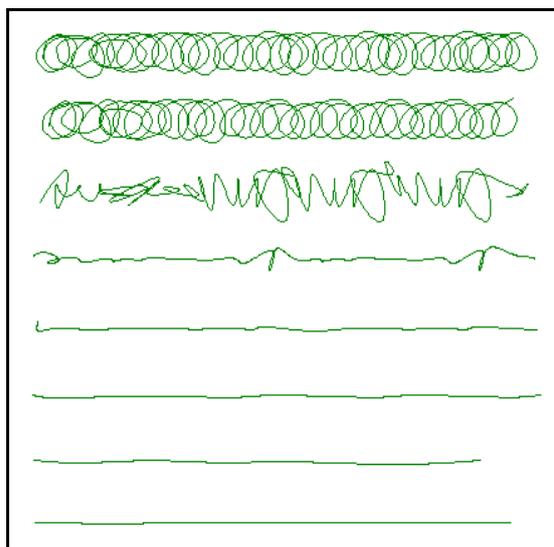


Gambar 4. Pengeditan kurva

Uji coba pada gambar 5 dan 6 merupakan uji sintesa kurva dengan menerapkan representasi multiresolusi pada kurva contoh.

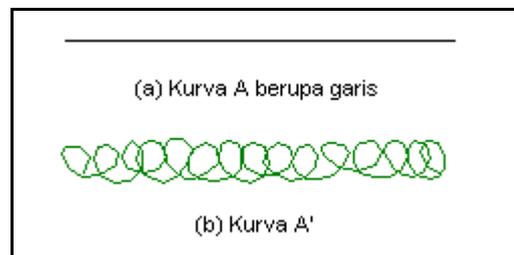


Gambar 5. Kurva Contoh pada resolusi 7

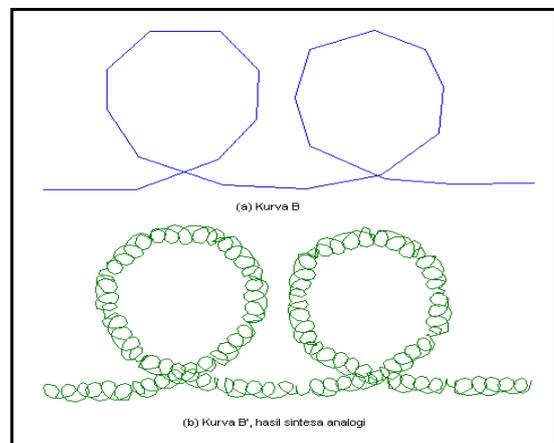


Gambar 6. Hasil sintesa kurva contoh pada resolusi yang berbeda

Uji sintesa analog kurva akan dilakukan untuk membentuk kurva baru yang mirip dengan kurva contoh yang diberikan pada *sweep*. Perhatikan gambar 7 dan 8.

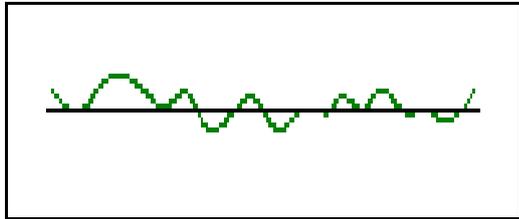


Gambar 7. Kurva contoh dan *sweep*



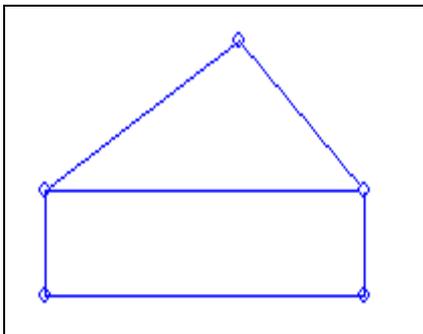
Gambar 8. Kurva hasil dan *sweep*

Uji coba berikut adalah uji pembentukan kurva dengan menerapkan *constraint* dalam membentuk kurva baru pada *sweep*. Kurva pada Gambar 9 digunakan sebagai contoh dalam pembentukan kurva baru.



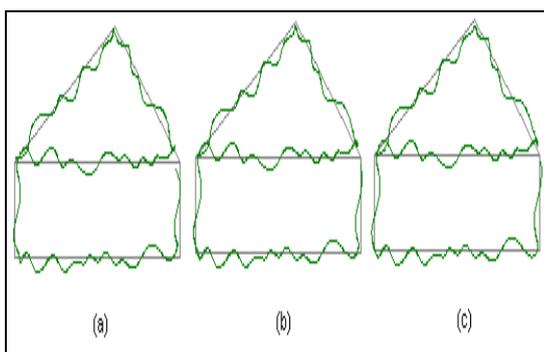
Gambar 9. Kurva Contoh

Kurva baru akan dibentuk pada *sweep* seperti pada Gambar 10.



Gambar 10. Kurva *Sweep*

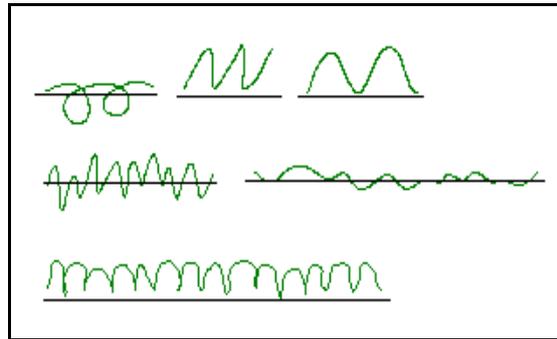
Hasil dari sintesa kurva pada *sweep* dengan *constraint* dapat dilihat pada Gambar 11.



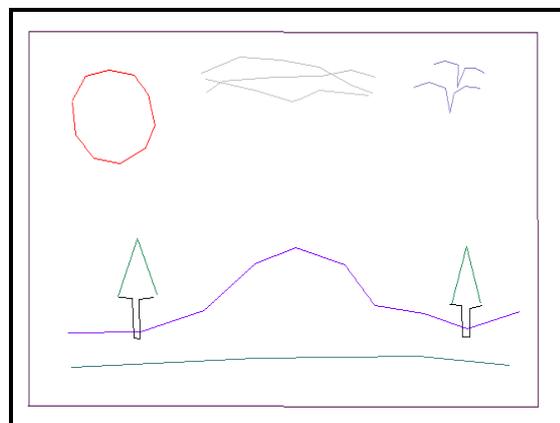
Gambar 11. Kurva *Sweep*

Uji coba berikut dilakukan untuk menunjukkan kemampuan aplikasi dalam membentuk suatu komposisi gambar dari beberapa sketsa yang dibuat. Gambar 12 merupakan kurva contoh yang digunakan untuk membentuk kurva baru pada sketsa yang diberikan. Gambar 13 merupakan sketsa yang

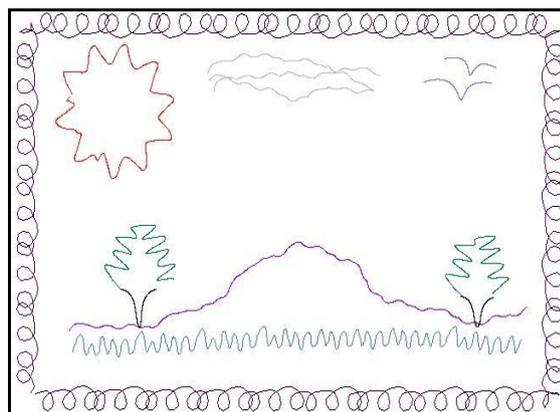
diberikan dan kurva baru dapat dilihat pada Gambar 14.



Gambar 12. Kurva contoh yang digunakan dalam pembentukan gambar



Gambar 13. Sketsa gambar



Gambar 14. Gambar hasil sintesa analog dari sketsa

6. KESIMPULAN

Beberapa kesimpulan yang dapat diambil dari Penelitian ini adalah:

- Aplikasi yang dibangun memudahkan pengguna untuk membuat kurva baru sesuai dengan keinginan tanpa membutuhkan keahlian pengguna dalam menggambar, karena aplikasi yang dibangun telah menyediakan pustaka kurva yang dapat digunakan untuk membuat kurva baru. Pengguna hanya menggambar sketsa sederhana saja dan aplikasi yang akan melakukan penggambaran untuk pengguna.
 - Aplikasi yang dibangun mampu memanipulasi kurva contoh dengan menggunakan representasi multiresolusi sehingga kurva contoh yang dihasilkan menjadi lebih banyak dan bervariasi.
 - Proses pembentukan kurva baru bergantung dari kurva contoh yang diberikan. Bila kurva contoh yang diberikan memiliki bentuk yang tidak beraturan maka hasil gambar yang baru juga tidak dapat diperkirakan. Bila kurva contoh yang diberikan memiliki suatu pola dan beraturan maka kurva baru yang dibentuk juga akan mengikuti pola dari kurva contoh tersebut.
 - Kurva baru yang dibentuk dengan memberikan suatu nilai *constraint* tertentu memiliki bentuk yang mengikuti pola dari bentuk dasar (*sweep*) yang diberikan. Kurva baru yang akan dibentuk tersebut akan melewati titik – titik kontrol dari bentuk dasar yang diberikan.
- Beberapa kemungkinan pengembangan yang dapat dilakukan dari Penelitian ini adalah :
- Representasi dari kurva yang digunakan dapat berupa kurva lain selain *B-Spline*.
 - Gambar yang dihasilkan bukan saja dalam bentuk dua dimensi, tetapi mungkin dalam bentuk tiga dimensi.
 - Gambar yang dihasilkan dapat dikembangkan dengan memadukan antara representasi multiresolusi dengan metode *curve analogies* dalam satu gambar hasil.
- 7. DAFTAR PUSTAKA**
1. A. Hertzmann, N. Oliver, Brian Curless dan Steven M. Seitz, “Curve Analogies”, *Thirteenth Eurographics Workshop on Rendering*, 2002.
 2. Aaron Hertzmann, Charles E. Jacobs, Nuria Oliver, Brian Curless dan David H. Salesin, “Image Analogies”, *Proceedings of SIGGRAPH 2001*, halaman 327–340, 2001.
 3. Adam Finkelstein dan David H. Salesin, “Multiresolution Curve”, *Proceedings of SIGGRAPH*, 1994.
 4. Alexei A. Efros dan Thomas K. Leung, “Texture Synthesis by Non-parametric Sampling”, *IEEE International Conference on Computer Vision*:1033–1038, September 1999.
 5. Alexei A. Efros dan William T. Freeman, “Image Quilting for Texture Synthesis and Transfer”, *Proceedings of SIGGRAPH 2001*, halaman 341–346, 2001.
 6. B. Kolman, David R. Hill, “Introductory Linear Algebra With Applications”, *Prentice Hall International, Inc*, 2001.
 7. Berthold K.P. Horn, Hugh M. Hilden dan Sahriar Negahdaripour, “Closed-Form Solution of Absolute Orientation Using Orthonormal Matrices”, *Journal of the Optical Society of America*, 5(7):1127-1135, July 1988.
 8. Donald Hearn dan M.Pauline Baker, “Computer Graphics”, *Prentice Hall International, In.*, 1990.
 9. Lee Markosian, Barbara J. Meier, Michael A. Kowalski, Loring S. Holden, J. D. Northrup dan John F. Hughes, “Art-based Rendering with Continuous Levels of Detail”, *NPAR 2000 :First International Symposium on Non Photorealistic Animation and Rendering*, halaman 59–66, June 2000.
 10. Nanik Suciati, “Kurva B-Spline Kubik Multiresolusi Berbasis Wavelet”, Thesis: Program Pasca Sarjana Universitas Indonesia, 1999.
 11. Paul Harrison, “A Non-Hierarchical Procedure for Re-Synthesis of Complex Textures”, *Proc. WCSG*, 2001.
 12. Rational Rose, “Rational University: Professional Education and Product Training”.
 13. Robert D. Kalnins, Lee Markosian, Barbara J. Meier, Michael A. Kowalski, Joseph C. Lee, Philip L. Davidson, Matthew Webb, John F. Hughes dan Adam Finkelstein, “WYSIWYG NPR: Drawing Strokes Directly on 3D Models”, *Proceedings of SIGGRAPH 2002*, Juli 2002.
 14. Siu Chi Hsu dan Irene H. H. Lee, “Drawing and Animation Using Skeletal Strokes”, *Proceedings of SIGGRAPH '94 (Orlando, Florida, Juli 24–29, 1994)*, Computer Graphics Proceedings, Annual Conference Series, halaman 109–118, Juli 1994.
 15. W. T. Freeman, E. C. Pasztor dan O. T. Carmichael, “Learning Low-Level Vision”, *Intl. J. Computer Vision*, 40(1):25–47, 2000.
 16. William T. Freeman, Joshua B. Tenenbaum dan Egon Pasztor, “An example-based approach to style translation for line drawings”, Technical Report TR99-11, MERL, Februari 1999.