

Studi Pembentukan Huruf Font Dengan Kurva Bezier

Agung Haryono, S.Kom., M.Kom.
Jurusan Teknologi Informasi
Institut Sains Terapan dan Teknologi Surabaya
Jl. Ngagel Jaya Tengah 73-77, Surabaya
andreasagung88@gmail.com

ABSTRAK

Kurva bezier (*baca : bes-ee-ay*) merupakan sebuah kurva parametrik yang sering diterapkan pada komputer grafis dan dalam bidang-bidang berhubungan lainnya. Kurva bezier sangat dikenal karena kemampuannya untuk menghasilkan kurva yang halus.

Penelitian ini bertujuan untuk mengetahui peran kurva bezier dalam pembentukan huruf-huruf *Font* dan mencari perbedaan peran kurva bezier dalam pembentukan *Font* TrueType dan *Font* PostScript.

Sedangkan manfaat dari penelitian ini adalah mengenalkan pada masyarakat cara kerja dan peran kurva bezier dalam pembentukan huruf-huruf *Font*.

Font TrueType dan *font* PostScript merupakan *font* vektor yang sama-sama menggunakan kurva bezier untuk menggambarkan kontur-kontur *glyph*. *Font* TrueType menggunakan kurva bezier kuadratik yaitu kurva bezier berordo 2, *Font* PostScript menggunakan kurva bezier kubik yaitu kurva bezier berordo 3.

Kata kunci : Kurva bezier, font TrueType, font PostScript.

1. PENDAHULUAN

Dalam dunia tipografi, *Font* adalah sebuah kumpulan karakter lengkap yang mempunyai satu ukuran dan gaya. Sedangkan dalam dunia komputer, *Font* merupakan file data elektronik yang mengandung sebuah kumpulan dari elemen penulisan (*glyph*), karakter-karakter atau simbol-simbol. Saat ini terdapat lebih dari ratusan ribuan jenis *Font* di dunia. Sistem operasi ternama di dunia, Microsoft Windows, yang mampu menampung hingga 4000 *Font* dalam sistemnya, menggunakan TrueType sebagai format *Font* digital.

2. METODOLOGI PENELITIAN

Telah diketahui pada latar belakang bahwa kurva bezier sangat berperan penting dalam pembentukan *Font outline* pada TrueType dan PostScript. Untuk mengenal lebih lanjut apa itu kurva bezier, *Font TrueType* dan PostScript, maka akan dijabarkan pada sub bab -sub bab dibawah ini.

A. Kurva Bezier

Pada akhir tahun 1950, komputer sudah mampu digunakan untuk membuat kalkulasi dan desain automobil. Sayangnya waktu itu tidak ada program yang mampu untuk membuat dan mewakili penggambaran kurva pada sebuah automobil secara efektif. Seorang insinyur asal Perancis yang bekerja pada Renault, Pierre Bezier, menerapkan sebuah kurva khusus untuk menyelesaikan masalah ini. Meskipun kurva ini dinamakan seperti itu, bukan berarti Pierre Bezier adalah orang yang menemukannya karena pada tahun yang sama P.De Castelju juga menerapkan kurva yang sama dalam perancangan tubuh automobil. Kurva parametrik ini telah diketahui sejak lama, namun tidak banyak orang yang mengenalnya. Baru pada tahun 1962, Pierre Bezier mempublikasikan kurva ini.

Kurva bezier terdiri dari beberapa titik pembentuknya yang berjumlah $n+1$ sehingga terdapat titik-titik P_0 sampai dengan P_n . Titik P_0 merupakan titik awal kurva dimana terdapat kurva akan berangkat dari titik tersebut dan melengkung ke arah P_n dan berhenti di titik tersebut. Semua ini dilakukan dengan t sebagai presentase waktu tempuh kurva dari titik P_0 dan P_n yaitu dari $t=0$ hingga $t=1$. Jadi ketika $t=0$ garis kurva berada tepat pada titik P_0 dan ketika $t=1$ garis kurva berada tepat pada titik P_n .

Perlu diketahui bahwa t bukanlah lamanya waktu yang ditempuh untuk membuat sebuah kurva. Variabel t menunjukkan seberapa jauh kurva bezier yang diwakili oleh fungsi $B(t)$ bergerak dari P_0 hingga P_1 . Sebagai contoh, ketika $t=0,25$ fungsi $B(t)$ berada pada seperempat jalan dari titik P_0 ke titik P_1 . Adapun persamaan untuk kurva bezier ordo satu (bezier linier) adalah :

$$B(t) = P_n + t(P_1 - P_n) = (1-t) P_n + tP_1 \quad t \in [0,1]$$

Jika diperhatikan rumus-rumus kurva bezier yang berurutan dari ordo satu sampai tiga menganut kaidah segitiga pascal. Setiap tingkat ordo satu pada kurva bezier berbanding lurus dengan baris-baris segitiga Pascal yaitu setiap ordo n akan memiliki koefisien-koefisien pada baris $n+1$ pada segitiga Pascal.

B. *Font*

Font merupakan salah satu bentuk seni yang dibuat hak ciptanya untuk dibeli atau dijual. Saat ini terdapat lebih dari ratusan ribuan jenis *Font* didunia. Seringkali dalam komputersasi orang-orang mengatakan *Font* ketika istilah *typeface* lebih dikatakan. Contoh dari *typeface* antara lain Times, Courier dan Arial karena memasukkan banyak ukuran dan banyak gaya. Sedangkan contoh *Font* antara lain bold (cetak tebal), italic (cetak miring), bold italic (cetak miring dan tebal), underline (bergaris bawah), *outline* (hanya mempunyai garis luar), dan shadow (berbayangan). Dalam definisi, *Font* merupakan keluarga *typeface*, sebagai contoh Arial 12-point italic, maka itu adalah dengan ukuran 12-point bercetak miring dari *typeface* Arial.

Typeface telah diketahui sendiri merupakan karakter-karakter khusus dan unik yang mempunyai banyak ukuran dan banyak gaya. Dengan demikian *typeface* memiliki banyak *Font* dengan berbagai ukuran dan didesain dengan kesatuan gaya.

Terdapat tiga macam format *Font* pada komputer antara lain *Font* bitmap, *outline* dan stroke. Perbedaan format tersebut bergantung pada komponen-komponen dalam pembentukan *Font*.

- ***Font* TrueType**

TrueType merupakan *Font* vektor atau *Font* berformat outline yang digunakan pada sistem operasi Microsoft Windows dan Apple Computer. Pada *Font* vektor, sebuah seri dari koordinasi menggambarkan sebuah kontur karakter, jadi perubahan penskalaan sederhana secara efektif mengecilkan atau memperbesar karakter. Sejak dikenalkan pada publik pada tahun 1991, *Font* TrueType telah menjadi teknologi *Font* yang dominan untuk digunakan sehari-hari. Karena *Font* TrueType tidak membutuhkan bahasa pencetakan khusus, *Font* TrueType menggantikan peran *Font* PostScript pada dunia percetakan.

Terdapat ribuan *Font* TrueType yang tersedia di dunia dan cukup banyak diantaranya yang disediakan cuma-cuma pada web. Banyak dari *Font* ini telah ditinjau dan diubah ke dalam sumber-sumber lain. Sementara sebagian *Font* dapat diproses dengan baik, pembuatan *Font* TrueType yang kurang baik dapat mengandung error yang dapat mengakibatkan kerusakan pada komputer. *Font* yang didesain secara profesional dapat menghabiskan banyak biaya namun biasanya membutuhkan hint yang cukup berat dan telah diuji pada berbagai ukuran dan sudut pada kualitas optimal.

- **Font PostScript**

PostScript merupakan *Font* outline atau *Font* berbasis vektor yang mendeskripsikan karakter dengan konstruksi matematis yaitu kurva bezier. Karena dikonstruksi oleh rumus matematika, seperti *Font* TrueType, *Font* PostScript juga memungkinkan dilakukan perubahan ukuran, penggambaran karakter dalam berbagai ukuran dan dalam resolusi tinggi. Karakter-karakter disimpan pada tabel bitmap yang merepresentasikan setiap karakter untuk setiap ukuran. PostScript sendiri adalah bahasa pemrograman yang digunakan untuk mengubah instruksi pencetakan grafis yang rumit ke printer digital.

PostScript dikenalkan pada tahun 1984 oleh Adobe. Pada waktu itu terdapat dua tipe *Font* yang berbeda yaitu tipe 1 dan tipe 3. Dari kedua tipe tersebut, Tipe 1 merupakan tipe *Font* yang memiliki format yang lengkap.

C. Metode Analisis

Penelitian tentang fungsi kurva bezier dalam pembuatan *Font* TrueType dan PostScript ini dilakukan dengan cara mengumpulkan beberapa paper dari internet. Selain itu, pengambilan data juga dilakukan dengan mengumpulkan contoh-contoh glyph pada suatu *Font* beserta titik-titik koordinatnya. Hal ini dilakukan dengan cara membuka file *Font* TrueType yang berekstensi ttf dan file *Font* PostScript yang berekstensi pfb dengan dua program file *Font* editor berbeda. Penggunaa dua program ini hanyalah ditujukan untuk mempermudah memperoleh data untuk analisis. Batas-batas dalam penelitian ini adalah sebagai berikut :

- a. Peran kurva bezier ordo tertentu dalam pembentukan *Font* TrueType dan PostScript.
- b. Analisa titik-titik kurva bezier pada rancangan glyph karakter pada *Font* TrueType dan PostScript.
- c. File-file *Font* TrueType dan file-file *Font* PostScript yang hanya berlaku pada sistem operasi Microsoft Windows yaitu file ttf untuk *Font* TrueType dan file pfb untuk *Font* PostScript.
- d. *Font* PostScript yang dijabarkan dan dianalisis pada penelitian in hanyalah *Font* PostScript Tipe 1.

D. Prosedur Pengumpulan Data

Penelitian ini menggunakan prosedur tertentu dalam proses pengumpulan data. Prosedur pengumpulan data pada penelitian ini dijabarkan sebagai berikut :

- a. Mencari file-file yang diperlukan pada penelitian ini yaitu file ttf dan file pfb. Hal ini dilakukan dengan mencari file-file kemudian mengunduh tersebut pada web-web internet. Namun pencarian dengan internet tidak perlu dilakukan jika sistem operasi yang digunakan telah memiliki cukup banyak *Font* untuk dijadikan data. Sebagai keterangan, khusus untuk file berekstensi pfb pencairan dilakukan dengan internet karena Windows tidak menyediakan file-file tersebut.
- b. Membuka file –file berekstensi ttf dan pfb dengan program file *Font* editor yaitu *Font Creator Professional Editor 5.6* dan *FontLab Studio 5*. Kedua program tersebut memiliki perbedaan dalam kapasitas file yang dapat dibuka. *Font Creator* hanya mampu membuka file-file berekstensi ttf sedangkan *FontLab Studio* mampu membuka file-file baik berekstensi ttf maupun file berekstensi pfb.
- c. Mendapatkan titik-titik koordinat pada glyph karakter untuk setiap contoh file *Font* baik yang berekstensi ttf dan pfb. Setelah mengetahui titik-titik koordinat yang menunjukkan titik-titik penting pada kurva, langkah selanjutnya adalah mencatat titik-titik tersebut pada sebuah tabel. Setiap tabel hanya mewakili satu glyph karakter pada *Font* tertentu.

E. Prosedur Penganalisaan Data

Setelah melakukan pengumpulan data, langkah selanjutnya adalah penganalisaan data. Penelitian ini menggunakan prosedur tertentu dalam proses penganalisaan data. Prosedur penganalisaan data pada penelitian ini dijabarkan sebagai berikut :

- a. Menginputkan pada program sederhana beberapa contoh titik yang merupakan bagian kurva pada glyph. Program sederhana merupakan program open source yang didapatkan dari internet yang kemudian dimodifikasi untuk keperluan penelitian ini.
- b. Menganalisis gambar kurva yang terbentuk pada program.
- c. Mendapatkan gambar kurva pada *glyph* yang sebenarnya. Hal ini dilakukan dengan cara memperbesar gambar *glyph* yang merupakan output dari program

- Font* editor. Pembesaran gambar difokuskan hanya pada beberapa bagian kurva dari contoh *glyph* yang sesuai dengan bagian kurva yang diujikan pada program.
- d. Membandingkan gambar kurva yang didapatkan dari program dan gambar kurva yang didapatkan dari hasil pembesaran gambar *glyph* yang sebenarnya.
 - e. Menyimpulkan perbandingan untuk *Font TrueFont* dan *Font PostScript* secara keseluruhan berdasarkan hasil penelitian yang dilakukan.

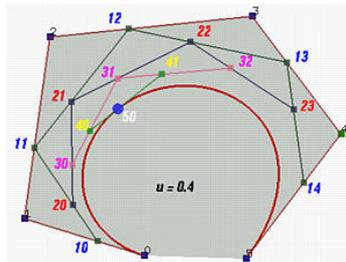
3. HASIL PEMBAHASAN

A. Algoritma De Casetljau

Tujuan dari algoritma ini adalah menentukan titik tengah dari dua titik terdekat kemudian menyatukan beberapa titik tengah satu sama lain. Berikut ini adalah algoritma singkatnya :

- a. Tentukan nilai t untuk setiap bilangan real dari 0 sampai 1. t adalah bilangan konstan.
- b. Tentukan P_i untuk setiap nilai t dengan $i=0, \dots, n$
- c. Untuk setiap $j=0$ lakukan perhitungan rumus kurva bezier.
- d. Hubungkan menjadi kurva $g(t)=P[n](t)$.

Berikut adalah gambar dari analisa algoritma de castlejau pada kurva bezier :



Gambar Analisa algoritma de Casteljau pada kurva bezier

Pada gambar, nilai t adalah 0,4. 10 merupakan segmen garis dari 00 dan 01, 11 merupakan segmen garis dari 01 dan 02, dan seterusnya hingga 14 adalah segmen garis dari 04 dan 05. Titik-titik baru tersebut diberi nomor dengan ketentuan $1i$.

B. *Font TrueType*

Garis tepi dari sebuah karakter atau biasa disebut *glyph* dalam *Font TrueType* dibuat dari bagian-bagian garis lurus dan kurva-kurva kuadratik Bezier. Kurva-kurva matematik yang lebih simpel ini sedikit efisien dalam prosesnya daripada kurva kubik bezier yang lazimnya terdapat pada PostScript dan digunakan pada *Font Type 1*.

Meskipun demikian, sebagian besar bentuk membutuhkan lebih banyak titik untuk menggambarkan dengan kurva kuadratik daripada kurva kubik.

TrueType *Collection* (TTC) adalah ekstensi dari format TrueType yang memungkinkan sejumlah *Font* dikombinasikan menjadi satu *file*, menciptakan ruang penyimpanan yang kokoh untuk koleksi-koleksi *Font* yang hanya menggunakan *glyph-glyph* yang berbeda pada beberapa karakter. Format *Font* TrueType dasar tersusun dari beberapa tabel yang ditentukan pada header. Sebuah nama tabel dapat tersusun atas 4 kata. Dalam *file* TTC, mengandung sebuah *ttcf* tabel yang menyatakan berapa jumlah *Font* di dalam koleksi. Sejumlah *Font* di dalam koleksi menggunakan tabel *glyph* yang sama. *Font* TrueType regular berekstensi *ttf*, sementara TrueType *Collection* berekstensi *ttc*.

Font TrueType disusun dengan kurva bezier kuadratik. Kurva bezier kuadratik merupakan kurva bezier berordo 2 yang memiliki satu titik *off-curve* sebagai titik kontrol. Namun ada beberapa area kurva yang memiliki 2 titik *off-curve*. Hal ini disebabkan oleh dua kurva kuadratik yang saling berurutan sehingga terlihat seperti kurva bezier kubik atau kurva bezier ordo 3.

Meskipun dipenuhi oleh titik-titik yang berbeda, setiap kontur pada *glyph* dapat dibagi menjadi beberapa segmen kurva. Setiap segmen kurva diawali dengan titik *on-curve* dan diakhiri juga dengan titik *on-curve*. Setiap segmen bisa memiliki 1 titik *off-curve* maupun 2 titik *off-curve*. Jika terdapat 2 titik *on-curve* yang berurutan maka segmen tersebut membentuk kurva bezier linier yang berupa garis lurus.

Perbedaan tempat dari titik-titik kontrol sangat berpengaruh pada tingkat kelengkungan kurva. Jika titik kontrol tidak jauh dari kurva maka kurva terbentuk tidak terlalu melengkung, jika titik kontrolnya terletak lebih jauh dari kurva sehingga kurva yang terbentuk terlihat lebih melengkung.

C. *Font* PostScript

Jika *Font* TrueType menggunakan kurva bezier kuadratik, *glyph-glyph* pada *Font* PostScript digambarkan dengan kurva bezier kubik. Satu set *glyph* pada *Font* PostScript dapat diubah-ubah ukurannya melalui transformasi matematika sederhana yang kemudian dapat dikirim ke printer PostScript. Karena data dari *Font* PostScript khususnya pada Tipe 1 merupakan deskripsi dari garis-garis tepi (*outline*) dari sebuah *glyph* maka *Font* ini juga disebut *Font outline*.

Pada sistem operasi Macintosh, *Font* PostScript terdiri dari *file Suitcase* dan *file Printer*. Hanya ada boleh satu *file suitcase* untuk seluruh keluarga *Font* yang menyediakan hingga 4 file printer yaitu *reguler*, *italic*, *bold* dan *bold italic*. Pada sistem operasi Windows, *Font* PostScript dapat terdiri dari 2 atau 3 file. Satu set 3 file terdiri dari file PFB yang mengandung data garis tepi, AFM yang mengandung informasi tentang *spacing* dan *kerning*, INF yang mengandung informasi tambahan yang berguna untuk instalasi. Selama instalasi berlangsung, dengan program Adobe Type Manager (ATM), Windows menghasilkan *file* PFM yang berbasis pada *file* AFM dan *file* INF.

Adobe Type Manager merupakan program yang harus dipasang pada Windows terlebih dahulu sebelum melakukan instalasi *Font* PostScript. Namun tidak pada semua tipe Windows harus melakukan instalasi ini. ATM dibutuhkan pada sistem operasi Windows 95, 98 NT dan Windows ME. Namun pada sistem operasi Windows 2000, XP dan Vista tidak perlu melakukan instalasi ATM.

File PFB tidak jauh beda dengan file TTF. File PFB juga terdiri dari beberapa tabel yang mengandung informasi tersimpan pada header. *Glyph-glyph* karakter tersusun pada tabel memiliki informasi sendiri-sendiri. Jumlah *glyph* juga sangat banyak, bermacam-macam dan berlainan bentuk, sama halnya seperti *file* TTF pada *Font* TrueType.

Kurva-kurva bezier pada *Font* PostScript tidaklah sama dengan kurva-kurva bezier pada *Font* TrueType karena *Font* PostScript menggunakan kurva bezier kubik sedangkan *Font* TrueType menggunakan kurva bezier kuadratik. Selain itu kurva-kurva pada *glyph* *Font* PostScript cukup konsisten. Pada *glyph* *Font* PostScript, kurva hanya terdiri dari kurva bezier linier yang berupa garis lurus dan kurva bezier kubik yang memiliki 2 titik kontrol.

D. Perbandingan *Font* TrueType dengan *Font* PostScript

Banyak hal yang telah dijabarkan dan dianalisis mengenai kedua *Font* tersebut sehingga dapat disimpulkan adanya beberapa perbedaan dan persamaan di antara *Font* TrueType dan *Font* PostScript.

Persamaan dari kedua *Font* adalah keduanya sama-sama merupakan *Font outline*. *Glyph-glyph* yang dibentuk pada kedua *Font* tersebut disusun berdasarkan garis-garis tepi atau kontur yang dibuat berdasarkan kurva matematika. Kurva matematika yang dipakai adalah kurva bezier. Hal ini berlaku untuk keduanya. Jadi

tidak hanya *Font TrueType* saja yang menggunakan kurva bezier tetapi juga *Font PostScript*.

Meskipun keduanya sama-sama menggunakan kurva bezier, jenis kurva bezier yang digunakan pada kedua *Font* tersebut berbeda. *Font TrueType* menggunakan kurva bezier kuadratik yang menggunakan 3 titik untuk membentuk kurva, sedangkan *PostScript* menggunakan kurva bezier kubik yang menggunakan 4 titik untuk membentuk kurva. Perbandingan *Font TrueType* dan *Font PostScript* dapat dilihat pada tabel dibawah.

	<i>Font TrueType</i>	<i>Font PostScript</i>
Format <i>Font</i>	Outline / Vektor	Outline / Vektor
Pengembang	Microsoft dan Apple	Adobe System
Ekstensi file pada Windows	.TTF	.PFB dan .PFM
Jenis Kurva Bezier	Kuadratik (ordo 2), membutuhkan 3 titik yaitu 2 titik <i>on-curve</i> dan 1 titik <i>off-curve</i>	Kubik (ordo 3), membutuhkan 4 titik yaitu 2 titik <i>on-curve</i> dan 2 titik <i>off-curve</i>

4. KESIMPULAN DAN SARAN

A. Kesimpulan

Hasil dari penelitian ini dapat disimpulkan sebagai berikut :

1. *Font TrueType* dan *font PostScript* merupakan *font* vektor yang sama-sama menggunakan kurva bezier untuk menggambarkan kontur-kontur *glyph*.
2. *Font TrueType* menggunakan kurva bezier kuadratik yaitu kurva bezier berordo 2 yang membutuhkan dua titik yang merupakan titik ujung kurva dan satu titik kontrol.
3. *Font PostScript* menggunakan kurva bezier kubik yaitu kurva bezier berordo 3 yang membutuhkan dua titik yang merupakan titik ujung kurva dan dua titik kontrol kurva.

B. Saran

Saran untuk penelitian lanjutan di masa mendatang adalah perlu adanya program yang lebih kompleks dalam penggambaran kurva-kurva bezier sehingga dapat membentuk kontur-kontur *glyph* secara keseluruhan.

REFERENSI

- [1] Computer, Apple, Inc. 1993. *Inside Macintosh : Text*, Addison Wesley.
- [2] Prautzsch, Harmut, Wolfgang Boehm, Marco Paluszny, *Bezier and B-Spline Techniques*, Springer-Verlag Berlin Heidelberg New York, 2002
- [3] Solomon, David, *Curves and Surfaces for Computer Graphics*, Springer Verlag, August 2006
- [4] Kennedy, Jhon. *A Brief Introduction To Bezier Curves*. Mathematics Department. Santa Monica College.
- [5] Garfield, Simon. *Just My Type : A Book about Fonts*, Profile Books LTD ,Great Britain, 2010