

HUBUNGAN ALGORITMA BEZIER DAN B-SPLINE PADA FUNGSI HARMONISNYA UNTUK MENCIPTAKAN BENTUK KURVA SESUAI KEINGINAN

Djunaidy Santoso; Zulfany Erlisa Rasjid; Edy Irwansyah

Computer Science Department, School of Computer Science, Binus University
Jl. K. H. Syahdan No. 9 Palmerah Jakarta Barat 11480
djunsan2002@yahoo.com; zulfany erlisa_rasjid@yahoo.com; edirwan@binus.ac.id

ABSTRACT

Computer graphics is a science widely used in any areas. Its up-to-date technique is applied to solve image problems related to needs of customer and decision-making. Images can be analyzed so as produce good form which its accuracy is still under researches using CAD and CAM. This study aims to associate algorithm Hermite, Bezier, B-Spline on its harmony function by creating a proper image to a form that can be used in drawing engineering for decision making. This study produces an adequate algorithm for image design.

Keywords: computer graphics, algorithm, harmonic function, curve, picture, image, CAD, CAM

ABSTRAK

Grafika komputer adalah ilmu yang sebenarnya banyak digunakan di berbagai bidang. Tekniknya yang sangat mutakhir diperuntukkan untuk pemecahan masalah gambar atau citra yang sangat erat dengan kebutuhan customer maupun pengambilan keputusan. Impian gambar ataupun citra dapat di analisis sehingga menghasilkan bentuk yang baik yang masih terus diteliti ketepatannya menggunakan CAD maupun CAM. Tujuan dari penelitian yaitu menghubungkan algoritma Hermite, Bezier, B-Spline pada fungsi harmonisnya dengan menciptakan gambar atau citra yang tepat untuk suatu bentuk yang dapat dipakai dalam rekayasa gambar untuk pengambilan keputusan. Hasil penelitian ini menghasilkan suatu algoritma yang memadai untuk perancangan gambar atau citra.

Kata kunci: grafika komputer, algoritma, fungsi blending/harmonis, kurva, gambar, citra, CAD, CAM.

PENDAHULUAN

Dengan semakin banyaknya algoritma dalam bidang grafika komputer, semakin banyak waktu untuk pengambilan keputusan dalam menentukan pendekatan pemakaian rumusan/algoritma tersebut. Dari data statistik industri/perusahaan atau institusi, banyak kurva/citra yang dibuat tidak mulus atau terdapat zig-zag, sehingga tidak bisa dipergunakan di industri/institusi/organisasi. Untuk itu perlu adanya penelitian algoritma/rumusan yang baik dalam menghadapi rancangan bentuk gambar (kurva) yang disesuaikan dengan keinginan.

Tujuan dan manfaat dari penelitian ini yaitu: (1) untuk merancang kurva/gambar/citra bagi industri mobil (gambar mesin), kapal terbang/laut, dan desain rumah; (2) untuk menganalisis dengan baik atau merancang/membuat kurva /citra secara mulus dari gambar satellite (GIS); (3) untuk dipergunakan dalam bidang statistik dan Rumah Sakit serta industri percetakan dan lain-lain.

Landasan Teori Interpolasi

Kurva Hermite – Bentuk polynom Hermite diterangkan dengan dua titik akhir dan dua tangen vektor. Bentuk umum segment kurva yaitu:

$$q(t) = (x(t), y(t)) \rightarrow 2D$$

dimana

$$x(t) = a_x t^3 + b_x t^2 + c_x t + d_x$$

$$y(t) = a_y t^3 + b_y t^2 + c_y t + d_y \rightarrow \text{untuk dua dimensi (2D)}$$

$$z(t) = a_z t^3 + b_z t^2 + c_z t + d_z \rightarrow \text{untuk tiga dimensi (3D)} \quad (1)$$

Koeffisien dalam bentuk matrix C:

$$x(t) = a_x t^3 + b_x t^2 + c_x t + d_x = Ct = G_x M_H t$$

$$\text{untuk 2D tambah } G_y (a_y, b_y, c_y, d_y) \quad (2)$$

begitupun sama dengan 3D dibutuhkan G_x , G_y dan G_z ,

G_x adalah x-komponen, G_y adalah y-komponen dan G_z adalah z-komponen dari geometry matrix.

M_H adalah Hermite basis matrix.

Sekarang kita dapat menulis $x(t)$ dalam dua cara:

$$x(t) = a_x t^3 + b_x t^2 + c_x t + d_x$$

atau

$$G_x B_H(t)$$

Dimana

$B_H(t) = M_H t$ adalah Hermite blending function (fungsi harmonis Hermite).

Kemudian geometri matrix G dan M_H dapat kita tulis menjadi

$$C = G M_H \quad (3)$$

Untuk menggambar kurva tersebut, perlu dievaluasi dengan sederhana:
 $x(t)$ and $y(t)$ --> untuk 2D; untuk 3D tambahkan $z(t)$.

Matrix berbasis Hermite M_H (Fungsi Blending/Harmonis Hermite):

$$M_H = \begin{bmatrix} 2 & -3 & 0 & 1 \\ -2 & 3 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix}$$

Baris atas matrix hermite (Gambar 1):

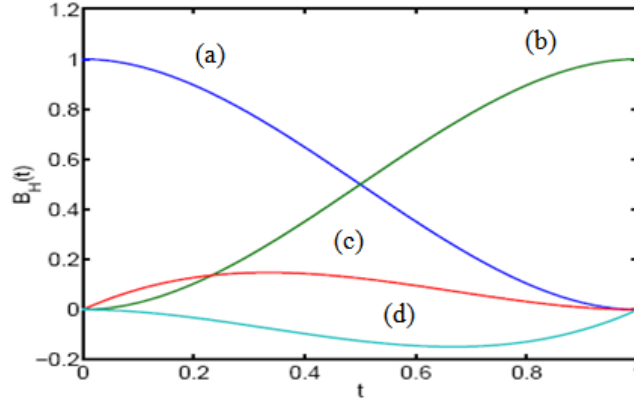
$$b_0 = 2t^3 - 3t^2 + 1 \quad (a)$$

$$b_1 = -2t^3 + 3t^2 \quad (b)$$

$$b_2 = t^3 - 2t^2 + t \quad (c)$$

$$b_3 = t^3 - t^2 \quad (d)$$

(4)



Gambar 1. Baris atas matrix Hermite

Approksimasi

Kurva Bezier sama dengan Kurva Hermite. Bedanya Bezier menggunakan empat titik kontrol.

$$P(t) = \sum_i^n P_i * B_{i,n}(t); \text{ Polynom Bezier} \quad (5)$$

$$B_{i,n} = C(n,i) * t^i * (1-t)^{n-i}; \text{ Fungsi Blending} \quad (6)$$

$$C(n,i) = n! / (i! * (n-i)!); \text{ Binomial Koeffisien} \quad (7)$$

Dengan demikian matrix berbasis Bezier (fungsi blending/harmonis Bezier) menjadi:

$$M_B = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Baris bawah matrix Bezier (Gambar 2):

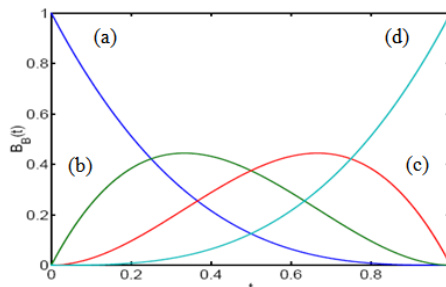
$$b_3 = t * t * t$$

$$b_2 = 3 * t * t * (1-t) \text{ atau } -3t^3 + 3t^2$$

$$b_1 = 3 * t * (1-t) * (1-t) \text{ atau } 3t^3 - 6t^2 + 3t$$

$$b_0 = (1-t) * (1-t) * (1-t) \text{ atau } -t^3 + 3t^2 - 3t + 1$$

(8)



Gambar 2. Matriks Bezier.

- (a) b_0 , erat dengan P_0
- (b) b_1 , erat dengan P_1
- (c) b_2 , erat dengan P_2
- (d) b_3 , erat dengan P_3

B-Spline mempunyai sejarah yang panjang dalam grafika Komputer. Natural Kubic Spline mempunyai C^2 (koefisien= tangen arah) yang berkelanjutan dan lebih halus daripada kurva Hermite dan kurva Bezier.

$$P(t) = \sum_{k=0}^L P_k * N_{k,m}(t); \quad N_{k,m}(t); \text{ fungsi Blending} \tag{9}$$

$$N_{k,m}(t) = \left(\frac{t - t_k}{t_{k+m} - t_k} \right) * N_{k,m-1}(t) + \left(\frac{t_{k+m} - t}{t_{k+m} - t_{k+1}} \right) * N_{k+1,m-1}(t) \tag{10}$$

Matrix berbasis B-spline dinyatakan sebagai berikut:

$$M_{Bs} = \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 0 & 4 \\ -3 & 3 & 3 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Ini dapat dijabarkan sebagai fungsi blending/harmonis b-spline $N_{i,k}(t)$ sebagai berikut:

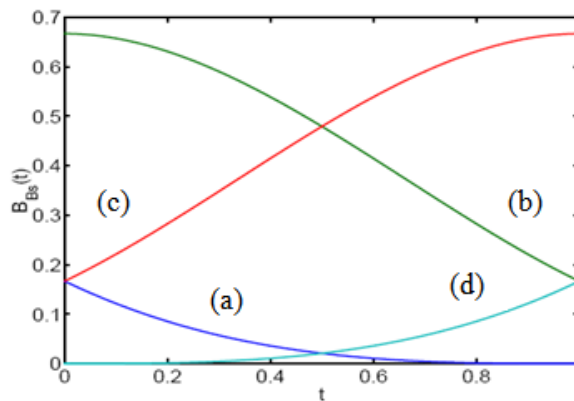
Baris atas matrix B-Spline (Gambar 3):

$$b_0 = \frac{(1-t)^3}{6} \quad \text{atau} \quad \frac{(-t^3 + 3t^2 - 3t + 1)}{6}$$

$$b_1 = \frac{(3t^3 - 6t^2 + 4)}{6}$$

$$b_2 = \frac{(-3t^3 + 3t^2 + 3t + 1)}{6}$$

$$b_3 = \frac{t^3}{6} \tag{11}$$

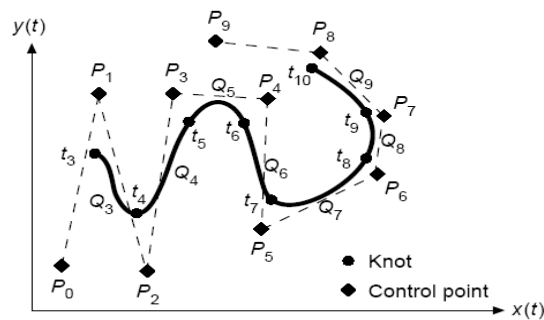


Gambar 3. Baris atas matrix B-Spline.

Fungsi blending/Harmonis B-spline.

- (a) b_0
- (b) b_1
- (c) b_2
- (d) b_3

B-Spline mempunyai karakteristik yaitu: (1) Bila jumlah knot(Q_i) banyak, semakin mulus kurva tersebut. (jumlah titik bantu alias Q_i dari P_0 sampai dengan P_n); (2) Jumlah titik P_i (control point) = jumlah kurva yang bisa dihasilkan.



METODE

Sebagai perbandingan antara kurva aproksimasi dengan kurva Interpolasi misalnya: Interpolasi kurva Hermite, tidak diteliti hanya dibahas sebagai perbandingan dengan kurva aproksimasi. Semua titik-titik statistic penelitian akan dilalui kurva tersebut baik dari titik awal sampai dengan titik akhir. Bedanya dengan kurva aproksimasi Bezier dan B-Spline hanya titik awal dan titik akhir saja yang dilaluinya. Titik-titik tengah antara titik awal dan titik akhir hanya dirata-ratakan saja (pada kurva Interpolasi Lagrange, Newton maupun Hermite dan lain-lain di daerah titik-titik awal dan di daerah titik-titik akhir terdapat alunan kurva yang sangat terjal). Ciri-ciri lain pada kurva Bezier dan B-Spline yaitu bila kurva diinginkan menjadi mulus harus dilakukan tangen arah berkelanjutan serta titik balik maximum dan titik balik minimum perlu diperhatikan.

Yang perlu diteliti pertama adalah fungsi Bezier. Dengan mengambil titik:

$$P_0 = (2/6) ; P_1 = (4/12)$$

$$P_2 = (7/3) ; P_3 = (9/10)$$

Sehingga $n=3$.

Setelah binomial koefisien dihitung, kemudian dihitung fungsi blending dan didapat:

$$b_{0,3}(t) = (1-t)^3$$

$$b_{1,3}(t) = 3*t*(1-t)^2$$

$$b_{2,3}(t) = 3*t^2*(1-t)$$

$$b_{3,3}(t) = t^3$$

Setelah fungsi blending didapat, baru dihitung fungsi polynomial Bezier $P(t)$ dengan (7), (6), (5). Maka didapat:

$$P_{0(t=0)} = (2.0000 / 6.0000)$$

$$P_{1(t=0.15)} = (2.9586 / 7.7902)$$

$$P_{2(t=0.35)} = (4.3860 / 8.1250)$$

$$P_{3(t=0.5)} = (5.5750 / 7.6250)$$

$$P_{4(t=0.65)} = (6.6250 / 7.2080)$$

$$P_{5(t=0.85)} = (8.0374 / 7.8242)$$

$$P_{6(t=1)} = (9.0000 / 10.0000)$$

Penelitian yang kedua menggunakan fungsi B-Spline. Dengan titik yang sama seperti penelitian fungsi Bezier yaitu: pertama menghitung knotvektor dan dimasukan nilai tersebut ke fungsi blending B-Spline didapat fungsi polynom B-Spline yaitu:

Dengan order $k=2$, didapat:

$$P_{0(t=0)} = (2 / 6)$$

$$\begin{aligned}
P_{1(t=0.5)} &= (3 / 9) \\
P_{2(t=1)} &= (4 / 12) \\
P_{3(t=1.5)} &= (5.5 / 7.5) \\
P_{4(t=2.0)} &= (7 / 3) \\
P_{5(t=2.5)} &= (8.0 / 6.5) \\
P_{6(t=3.0)} &= (9 / 10)
\end{aligned}$$

Dengan order k=3, didapat:

$$\begin{aligned}
P_{0(t=0)} &= (2 / 6) \\
P_{1(t=0.5)} &= (3.875 / 9.375) \\
P_{2(t=1)} &= (5.5 / 7.5) \\
P_{3(t=1.5)} &= (7.125 / 5.875) \\
P_{4(t=2.0)} &= (9 / 10)
\end{aligned}$$

Dengan order k=4, didapat:

$$\begin{aligned}
P_{0(t=0)} &= (2 / 6) \\
P_{1(t=0.5)} &= (5.5 / 7.625) \\
P_{2(t=1)} &= (9 / 10) \quad \text{dengan (10) dan (9)}
\end{aligned}$$

HASIL DAN PEMBAHASAN

Dari hasil perhitungan fungsi Bezier didapat:

$$\begin{aligned}
P_{0(t=0)} &= (2 / 6) \\
P_{1(t=0.15)} &= (2.9586 / 7.7902) \\
P_{2(t=0.35)} &= (4.386 / 8.125) \\
P_{3(t=0.5)} &= (5.575 / 7.625) \\
P_{4(t=0.65)} &= (6.625 / 7.208) \\
P_{5(t=0.85)} &= (8.0374 / 7.8242) \\
P_{6(t=1)} &= (9 / 10) \quad [2],[25,28]
\end{aligned}$$

Pembahasan untuk Bezier

Kurva yang pertama menghasilkan kurva polygon dan yang kedua menghasilkan kurva yang sama seperti pada kurva terakhir dari B-Spline, titik P_0 , P_3 dan P_6 sama dengan Kurva B-Spline dengan order k=4(selalu order k terakhir sama dengan jumlah titik). Hasil perhitungan fungsi B-Spline didapat:

Menurut Order k=2:

$$\begin{aligned}
P_{0(t=0)} &= (2 / 6) \\
P_{1(t=0.5)} &= (3 / 9) \\
P_{2(t=1)} &= (4 / 12) \\
P_{3(t=1.5)} &= (5.5 / 7.5) \\
P_{4(t=2.0)} &= (7 / 3) \\
P_{5(t=2.5)} &= (8.0 / 6.5) \\
P_{6(t=3.0)} &= (9 / 10)
\end{aligned}$$

Order k=3:

$$\begin{aligned}
P_{0(t=0)} &= (2 / 6) \\
P_{1(t=0.5)} &= (3.875 / 9.375) \\
P_{2(t=1)} &= (5.5 / 7.5) \\
P_{3(t=1.5)} &= (7.125 / 5.875)
\end{aligned}$$

$$P_{4(t=2.0)} = (9 / 10)$$

Order k=4, didapat:

$$P_{0(t=0)} = (2 / 6)$$

$$P_{1(t=0.5)} = (5.5 / 7.625)$$

$$P_{2(t=1)} = (9 / 10)$$

Dari hasil data perhitungan Bezier, bahwa pada titik-titik:

$$P_{0(t=0)} = (2.0000 / 6.0000), P_{3(t=0.5)} = (5.5750 / 7.6250), P_{6(t=1)} = (9.0000 / 10.0000)$$

hampir sama dengan B-Spline pada k=2 di titik-titik:

$$P_{0(t=0)} = (2.0000 / 6.0000), P_{3(t=1.5)} = (5.5000 / 7.5000), P_{6(t=3.0)} = (9.0000 / 10.0000)$$

dan pada k=3 di titik titik:

$$P_{0(t=0)} = (2.0000 / 6.0000), P_{2(t=1)} = (5.5000 / 7.5000), P_{4(t=2.0)} = (9.0000 / 10.0000)$$

serta pada k=4 di titik titik:

$$P_{0(t=0)} = (2.0000 / 6.0000), P_{1(t=0.5)} = (5.5000 / 7.6250), P_{2(t=1)} = (9.0000 / 10.0000)$$

Kalau diperhatikan, bahwa hasil kurva Bezier = Kurva B-Spline pada k =4 selalu yang terakhir.

Pembahasan pada B-Spline:

Pada order k=2 selalu terjadi kurva polygon. Kurva terakhir B-Spline sama dengan kurva Bezier. Dari kedua algoritma diatas didapat kurva yang mulus pada pembuatan angka 0 – 9, huruf A – Z (a –z) dan peta perbatasan wilayah suatu Negara serta pembuatan gambar gambar elektronika maupun gambar-gambar hasil analisis di bidang kedokteran, Bila kedua algoritma yang diteliti dikembangkan pada fungsi blending, akan ada hasil yang lebih baik.

PENUTUP

Untuk membuat citra yang tepat untuk sebuah bentuk, algoritma yang diusulkan adalah B-Spline. Dari hasil yang didapat, algoritma B-Spline lebih luas dan memungkinkan lebih banyak pilihan dari order k-nya yang sangat cocok dengan aplikasi pada bidang sipil, arsitek, elektronika, industri automobil, industri kapal terbang maupun laut, teknik industri, kedokteran dan pemetaan, dan lain-lain. Jadi, semakin banyak titik bantu (Q_i) diantara titik awal dan titik akhir semakin mulus kurva tersebut, tetapi semakin lama prosesnya karena banyak titik yang harus diproses. Walaupun Bezier tak ada pilihan untuk kurva yang cocok untuk aplikasi, penggunaannya sama seperti yang sudah dijelaskan pada kesimpulan di atas.

DAFTAR PUSTAKA

- Giloi, Wolfgang K. (1999). *Interactive Computer Graphics: Data Structures, Algorithms, Languages*. New Jersey: Prentice Hall.
- Newman, William M., Sproull, Robert F. (2000). *Principles of Interactive Computer Graphics*. New York: Tata-McGraw Hill.
- Rogers, David F., Adams, J. Alan. (1976). *Mathematical Elements for Computer Graphics*. New York: McGraw-Hill.