Spline Curves
another view

CSE452A 17

[ Slides courtesy of Kavita Bala ]
Defining spline curves

- At the most general they are parametric curves

\[ S = \{ f(t) \mid t \in [0, N] \} \]

- For splines, \( f(t) \) is piecewise polynomial
  - for this lecture, the discontinuities are at the integers
Defining spline curves

- At the most general they are parametric curves

\[ S = \{ f(t) \mid t \in [0, N] \} \]

- For splines, \( f(t) \) is piecewise polynomial
  - for this lecture, the discontinuities are at the integers
Defining spline curves

• At the most general they are parametric curves

\[ S = \{ f(t) \mid t \in [0, N] \} \]

• For splines, \( f(t) \) is piecewise polynomial
  – for this lecture, the discontinuities are at the integers
Defining spline curves

- At the most general they are parametric curves

\[ S = \{ f(t) \mid t \in [0, N] \} \]

- For splines, \( f(t) \) is piecewise polynomial
  - for this lecture, the discontinuities are at the integers
Defining spline curves

- Generally $f(t)$ is a piecewise polynomial
  - for this lecture, the discontinuities are at the integers
  - e.g., a cubic spline has the following form over $[k, k + 1]$:  
    \[
    x(t) = a_x t^3 + b_x t^2 + c_x t + d_x \\
    y(t) = a_y t^3 + b_y t^2 + c_y t + d_y
    \]
  - Coefficients are different for every interval
Coordinate functions

2D spline
Coordinate functions

2D spline

coordinate function $x(t)$
Coordinate functions

2D spline

coordinate function $y(t)$

coordinate function $x(t)$

$t$ $0$ $1$ $2$

$y$

$2$

$1$

$0$
Coordinate functions

2D spline

coordinate function $y(t)$

coordinate function $x(t)$
Coordinate functions

2D spline

coordinate function $x(t)$

coordinate function $y(t)$
Coordinate functions

2D spline

coordinate function $x(t)$

coordinate function $y(t)$
Coordinate functions

2D spline

coordinate function \( x(t) \)

coordinate function \( y(t) \)
Coordinate functions

2D spline

coordinate function \( x(t) \)

coordinate function \( y(t) \)
Coordinate functions

2D spline

coordinate function $x(t)$

coordinate function $y(t)$
Control of spline curves

- Specified by a sequence of controls (points or vectors)
- Shape is guided by control points (aka control polygon)
  - interpolating: passes through points
  - approximating: merely guided by points
Control of spline curves

- Specified by a sequence of controls (points or vectors)
- Shape is guided by control points (aka control polygon)
  - interpolating: passes through points
  - approximating: merely guided by points
Control of spline curves

- Specified by a sequence of controls (points or vectors)
- Shape is guided by control points (aka control polygon)
  - interpolating: passes through points
  - approximating: merely guided by points
Control of spline curves

- Specified by a sequence of controls (points or vectors)
- Shape is guided by control points (aka control polygon)
  - interpolating: passes through points
  - approximating: merely guided by points
Spline Segments
Trivial example: piecewise linear

- This spline is just a polygon
  - control points are the vertices
- But we can derive it anyway as an illustration
- Each interval will be a linear function
  - \( x(t) = at + b \)
    - constraints are values at endpoints
  - \( b = x_0 \); \( a = x_1 - x_0 \)
    - this is linear interpolation
Trivial example: piecewise linear

- Vector formulation

\[ x(t) = (x_1 - x_0)t + x_0 \]
\[ y(t) = (y_1 - y_0)t + y_0 \]
\[ f(t) = (p_1 - p_0)t + p_0 \]

- Matrix formulation

\[ f(t) = \begin{bmatrix} t & 1 \end{bmatrix} \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \end{bmatrix} \]
Trivial example: piecewise linear

- Basis function formulation
  - regroup expression by \( p \) rather than \( t \)
    \[
    f(t) = (p_1 - p_0)t + p_0 \\
    = (1 - t)p_0 + tp_1
    \]
  - interpretation in matrix viewpoint
    \[
    f(t) = \left( \begin{bmatrix} t & 1 \end{bmatrix} \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix} \right) \begin{bmatrix} p_0 \\ p_1 \end{bmatrix}
    \]
Trivial example: piecewise linear

- Vector blending formulation: “average of points”
  - blending functions: contribution of each point as $t$ changes

\[ b_0(t) = 1 - t \]
\[ b_1(t) = t \]
Hermite splines

- Less trivial example
- Form of curve: piecewise cubic
- Constraints: endpoints and tangents (derivatives)
Hermite splines

- Solve constraints to find coefficients

\[ x(t) = at^3 + bt^2 + ct + d \]
\[ x'(t) = 3at^2 + 2bt + c \]
\[ x(0) = x_0 = d \]
\[ x(1) = x_1 = a + b + c + d \]
\[ x'(0) = x'_0 = c \]
\[ x'(1) = x'_1 = 3a + 2b + c \]

\[ d = x_0 \]
\[ c = x'_0 \]
\[ a = 2x_0 - 2x_1 + x'_0 + x'_1 \]
\[ b = -3x_0 + 3x_1 - 2x'_0 - x'_1 \]
Matrix form of spline

\[ f(t) = at^3 + bt^2 + ct + d \]

\[
\begin{bmatrix}
  t^3 & t^2 & t & 1
\end{bmatrix}
\begin{bmatrix}
  \times & \times & \times & \times \\
  \times & \times & \times & \times \\
  \times & \times & \times & \times \\
  \times & \times & \times & \times \\
\end{bmatrix}
\begin{bmatrix}
  p_0 \\
  p_1 \\
  p_2 \\
  p_3 \\
\end{bmatrix}
\]

\[ f(t) = b_0(t)p_0 + b_1(t)p_1 + b_2(t)p_2 + b_3(t)p_3 \]
Matrix form of spline

\[ f(t) = at^3 + bt^2 + ct + d \]

\[
\begin{bmatrix}
t^3 & t^2 & t & 1
\end{bmatrix}
\begin{bmatrix}
x & x & x & x & x
x & x & x & x
x & x & x & x
x & x & x & x
\end{bmatrix}
\begin{bmatrix}
p_0 \\
p_1 \\
p_2 \\
p_3 \\
\end{bmatrix}
\]

\[ f(t) = b_0(t)p_0 + b_1(t)p_1 + b_2(t)p_2 + b_3(t)p_3 \]
Matrix form of spline

\[ f(t) = at^3 + bt^2 + ct + d \]

\[
\begin{bmatrix}
  t^3 & t^2 & t & 1 \\
\end{bmatrix}
\begin{bmatrix}
  \times & \times & \times & \times \\
  \times & \times & \times & \times \\
  \times & \times & \times & \times \\
  \times & \times & \times & \times \\
\end{bmatrix}
\begin{bmatrix}
  p_0 \\
p_1 \\
p_2 \\
p_3 \\
\end{bmatrix}
\]

\[ f(t) = b_0(t)p_0 + b_1(t)p_1 + b_2(t)p_2 + b_3(t)p_3 \]
Hermite splines

- Matrix form is much simpler

\[ f(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ t_0 \\ t_1 \end{bmatrix} \]

- coefficients = rows
- basis functions = columns
Hermite splines

- Hermite blending functions
Hermite splines

- Hermite basis functions
Hermite to Bézier

- Mixture of points and vectors is awkward
- Specify tangents as differences of points
Hermite to Bézier

- Mixture of points and vectors is awkward
- Specify tangents as differences of points
Hermite to Bézier

- Mixture of points and vectors is awkward
- Specify tangents as differences of points
Hermite to Bézier

- Mixture of points and vectors is awkward
- Specify tangents as differences of points

- note derivative is defined as 3 times offset
  - reason is illustrated by linear case
Hermite to Bézier

\[
\begin{align*}
\mathbf{p}_0 &= \mathbf{q}_0 \\
\mathbf{p}_1 &= \mathbf{q}_3 \\
t_0 &= 3(\mathbf{q}_1 - \mathbf{q}_0) \\
t_1 &= 3(\mathbf{q}_3 - \mathbf{q}_2)
\end{align*}
\]

\[
\begin{bmatrix}
\mathbf{p}_0 \\
\mathbf{p}_1 \\
\mathbf{v}_0 \\
\mathbf{v}_1
\end{bmatrix}
= 
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 \\
-3 & 3 & 0 & 0 & 0 \\
0 & 0 & -3 & 3 & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{q}_0 \\
\mathbf{q}_1 \\
\mathbf{q}_2 \\
\mathbf{q}_3
\end{bmatrix}
\]
\[
\begin{bmatrix}
p_0 \\ p_1 \\ v_0 \\ v_1 
\end{bmatrix} = 
\begin{bmatrix}
1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 
\end{bmatrix} 
\begin{bmatrix}
q_0 \\ q_1 \\ q_2 \\ q_3 
\end{bmatrix}
\]

\[
f(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} 
\begin{bmatrix}
2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 
\end{bmatrix} 
\begin{bmatrix}
p_0 \\ p_1 \\ t_0 \\ t_1 
\end{bmatrix}
\]

\[
f(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} 
\begin{bmatrix}
2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 
\end{bmatrix} 
\begin{bmatrix}
1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 
\end{bmatrix} 
\begin{bmatrix}
q_0 \\ q_1 \\ q_2 \\ q_3 
\end{bmatrix}
\]
Bézier matrix

\[ f(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} \]

- note that these are the Bernstein polynomials

\[ b_{n,k}(t) = \binom{n}{k} t^k (1 - t)^{n-k} \]

and that defines Bézier curves for any degree
Bézier basis
Rendering the curve

- Option 1: uniformly sample in \( t \)
- Problem
  - may oversample smooth regions: slow
  - may undersample highly curved regions: faceted rendering
Evaluating by subdivision

- Recursively split spline
  - stop when polygon is within epsilon of curve
De Casteljau algorithm

- Adaptive subdivision!
Recursive algorithm

void DrawRecBezier (float eps) {
    if Linear (curve, eps)
        DrawLine (curve);
    else
        SubdivideCurve (curve, leftC, rightC);
        DrawRecBezier (leftC, eps);
        DrawRecBezier (rightC, eps);
}
Evaluating by subdivision

- Recursively split spline
  - stop when polygon is within epsilon of curve
- Termination criteria
  - distance between control points
  - distance of control points from line
  - angles in control polygon
Continuity

- Smoothness can be described by degree of continuity
  - zero-order ($C^0$): position matches from both sides
  - first-order ($C^1$): tangent matches from both sides
  - second-order ($C^2$): curvature matches from both sides
Continuity
Continuity

C0 continuous
Continuity

C1 continuous
Affine invariance

- Transforming the control points is the same as transforming the curve
  - true for all commonly used splines
  - extremely convenient in practice...
Affine Invariance Revisited

Bézier matrix

\[
f(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix}
\]
Another interpolating spline
Another interpolating spline
Another interpolating spline

Look at Hermite Spline
Another interpolating spline

Look at Hermite Spline
Another interpolating spline

Look at Hermite Spline
Another interpolating spline

Look at Hermite Spline
Another interpolating spline

Look at Hermite Spline
Another interpolating spline

\[ f(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ t_0 \\ t_1 \end{bmatrix} \]
Another interpolating spline

\[ f(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ t_0 \\ t_1 \end{bmatrix} \]

\[ p_0 = q_k \]
\[ p_1 = q_k + 1 \]
\[ v_0 = 0.5(q_{k+1} - q_{k-1}) \]
\[ v_1 = 0.5(q_{k+2} - q_K) \]

\[
\begin{bmatrix} p_0 \\ p_1 \\ t_0 \\ t_1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & -0.5 & 0 & 0.5 \end{bmatrix} \begin{bmatrix} q_{k-1} \\ q_k \\ q_{k+1} \\ q_{k+2} \end{bmatrix}
\]
Catmull-Rom Curve

\[
\begin{bmatrix}
  p_0 \\
p_1 \\
t_0 \\
t_1
\end{bmatrix} =
\begin{bmatrix}
t^3 & t^2 & t & 1
\end{bmatrix}
\begin{bmatrix}
  2 & -2 & 1 & 1 \\
-3 & 3 & -2 & -1 \\
0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
  0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
-0.5 & 0 & 0.5 & 0 \\
0 & -0.5 & 0 & 0.5
\end{bmatrix}
\begin{bmatrix}
  q_{k-1} \\
q_k \\
q_{k+1} \\
q_{k+2}
\end{bmatrix}
\]
Catmull-Rom basis