Generation of Stratified Samples for B-Spline Pixel Filtering

Michael Stark Peter Shirley Michael Ashikhmin

April 8, 2003

Abstract

B-spline filter kernals have proved useful in many pixel sampling applications. A cubic B-spline filter kernal, having a width of four pixels, is particularly effective. In distribution ray tracing, pixel filters are evaluated implicitly by having the density of sampling proportional to the filter value. In this work we present a simple method to generate random samples having an underlying B-spline density function. To reduce error it is important to stratify the samples, akin to jittering for uniform sampling. We provide an algebraic and a numerical technique for doing this for B-spline kernals of degree 1, 2, and 3.

1 Introduction

Distribution ray tracing often uses a stochastic method (Monte Carlo or quasi-Monte Carlo integration) to evaluate the integrals of pixel filters for antialiasing. For reduced variance, the pixel filters are evaluated implicitly by choosing random samples so that the probability density function (PDF) of the samples is proportional to the filter. Doing so, however, can be difficult because typically all that is available to the programmer is a uniform random number generator.

Ordinarily, generating samples according to a particular (1D) PDF involves computing the inverse of the corresponding cumulative density function, (the integral of the PDF; see Section 5). This inverse function applied to a uniform random variable in $[0, 1)$, produces a random variable having the desired PDF. Convergence can be improved by stratifying the samples, i.e, by choosing n samples so that sample k is randomly chosen in the k^{th} subinterval of size $1/n$ in $[0, 1)$. Computing the inverse of the cumulative density can, however, be a difficult problem.

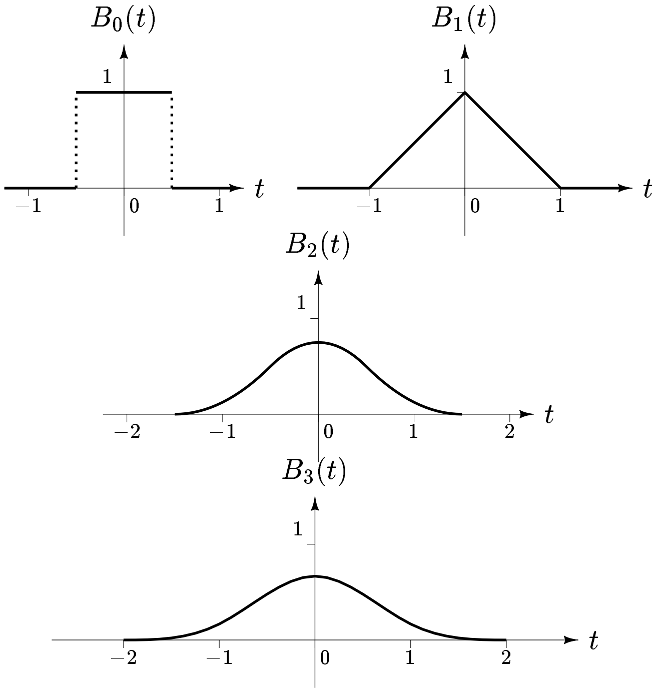


Figure 1: B-spline basis functions, centered at the origin.

2 B-Spline Filter Kernals

B-spline kernals for pixel filters are attractive for several reasons: they are piecewise polynomial with the maximum possible derivative continuity, and they are zero outside a closed interval. Furthermore, and perhaps most importantly, the filter kernals sum to unity across the entire image when a B-spline kernal is placed at each pixel. (This property is necessary to avoid temporal aliasing of moving bright subpixel details, such as sparks from an explosion.) Figure 1 illustrates B-spline functions of degree 0 through 3. Because of the piecewise polynomial character of B-splines, a B-spline PDF up to degree 3 can be integrated and inverted (piecewise) algebraically.

We advocate the use of a cubic B-spline filter kernal in particular because the it appears to produce somewhat better results than a quadratic or linear B-spline kernal. The development will therefore be done for a cubic filter; the corresponding results for linear and quadratic B-spline kernals are given in the Appendix.

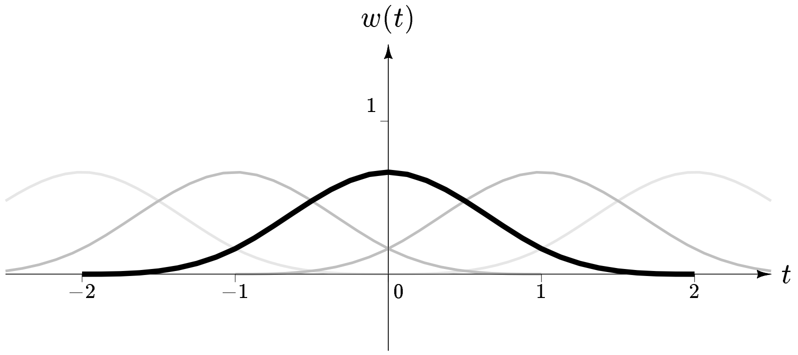


Figure 2: Cubic B-spline filter kernals centered at integer values. At each point t, at most four of the functions are nonzero, and the values sum to 1.

3 The Cubic B-spline Filter Kernal

The cubic B-spline filter kernal is a piecewise polynomial with four segments, each of which is a translation/reflection of one of the following two polynomials in $t \in [0, 1]$:

$$\begin{aligned} b_0(t) &= \frac{1}{6}t^3 \\ b_1(t) &= \frac{1}{6}\left(-3t^3 + 3t^2 + 3t + 1\right). \end{aligned}$$

The full filter function is

$$B_3(t) = \begin{cases} b_0(t+2) & \text{if } -2 \le t \le -1, \\ b_1(t+1) & \text{if } -1 < t \le 0, \\ b_1(1-t) & \text{if } 0 < t \le 1, \\ b_0(2-t) & \text{if } 1 < t \le 2, \\ 0 & \text{otherwise,} \end{cases}$$

and is thus zero off $[-2, 2]$. Figure 2 illustrates. The 2D B-spline filter kernal is simply the product $B_3(s)B_3(t)$, and as such is a separable 2D filter.

4 Generating Unstratified Samples

Computing samples corresponding to the a particular distribution generally requires the inverse of the integral of the cumulative distribution. However, if we are content with *unstratified* random samples for a B-spline distribution, there is a much simpler approach.

One view of the B-spline kernal is as the product of repeated convolution of the unit width box filter B_0. Each step of such a convolution spreads the "energy" of each point uniformly in a unit width region around the point. This has a probabilistic interpretation: the probability density function (PDF) of the sum of two random variables is the convolution of the PDFs of the random variables [2]. The PDF of the sum of two uniform random variables in $[-\frac{1}{2}, \frac{1}{2})$ is therefore the B-spline function B_1, and the PDF of the sum of three such random variables is B_2, etc.

If $r()$ is a call to a random number generator that returns uniformly random numbers in the range $[0, 1)$, then the summation

$$t = -\frac{n}{2} + \sum_{i=0}^{n} r() \tag{1}$$

generates a random t with a B-spline PDF of degree n ($n = 3$ for a cubic B-spline distribution).

To get an (x, y) pair with the appropriate B-spline density, we simply apply the summation of Equation (1) above separately for x and y.

5 Generating Stratified Samples

Generating stratified (or uniform) samples is a more challenging task. For a 1D function, the standard technique is to compute the inverse of the cumulative density function; that is, solve for r in the equation

$$r = \int_{-\infty}^{u} B_3(t)\, dt. \tag{2}$$

For a separable 2D function the process is done separately in each variable.

Because our underlying cubic B-spline function is composed of pieces with relative volumes $\frac{1}{24}$, $\frac{11}{24}$, $\frac{11}{24}$, $\frac{1}{24}$, each piece can be treated separately provided we account for those weightings. For example, it is straightforward to generate stratified samples with the underlying density proportional to b_0 by taking a uniformly stratified sample r and solving for u in

$$r = \int_{0}^{u} 24\, b_0(t)\, dt = \int_{0}^{u} 4t^3\, dt, \tag{3}$$

which yields:

$$u = r^{1/4}. \tag{4}$$

If $0 \leq r < \frac{1}{24}$, the corresponding sample lies in the segment of $B_3(t)$ where $-2 \leq t < -1$, and the value is therefore given by $(24r)^{1/4} - 2$. Similarly, if $\frac{23}{24} \leq r \leq 1$ the sample is $2 - (24(1-r))^{1/4}$.

For b_1 we have:

$$r = \int_0^u \frac{4}{11}(-3t^3 + 3t^2 + 3t + 1)\, dt \tag{5}$$

which after intergration becomes

$$11r = -3u^4 + 4u^3 + 6u^2 + 4u, \tag{6}$$

a quartic equation which is more difficult equation to solve. We could use a general numeric quartic solver to solve for u, but the code is messy and has many conditional expressions resulting in many branches (undesirable for a modern pipelined architecture). Instead we can use an algebraic solution (Section 5.1) or a numerical root finding method (Section 5.3).

5.1 Algebraic Solution For Second Curve Segment

The solution in u to the quartic equation

$$-3u^4 + 4u^3 + 6u^2 + 4u - 11r = 0, \tag{7}$$

as a function of r can be solved analytically by known methods (e.g., [1]). There are generally four roots; after considerable manipulation, we found the following formula for the desired root. Given r in $[0, 1]$, compute [1]

$$\begin{aligned} t &= 32 + 352r & (8)\\ v &= \sqrt{t^2 + \left(\frac{4 - 132r}{3}\right)^3} & (9)\\ w &= 3\left(\sqrt[3]{t+v} + \sqrt[3]{t-v} - 2\right) & (10)\\ s &= \sqrt{22 + w} & (11) \end{aligned}$$

and the desired solution is then

$$u = \frac{1}{6}\left(2 + s - \sqrt{26 - w + \frac{160}{s}}\right). \tag{12}$$

Under the proper transformations, this completes the result for the two inside curve segments.

---

[1] The value of $t - v$ can be negative. In the absence of a cube root function, the cube root of x with the proper sign can be computed in C using

```
(x < 0.0 ?  -pow(-x, 1.0/3.0) :  pow(x, 1.0/3.0));
```

5.2 Generation of the Full Filter

Combining the results for the two curve segments, the following function transforms a uniform distribution on $[0,1)$ to a cubic B-spline distribution:

$$D(r) = \begin{cases} (24r)^{1/4} - 2, & \text{if } 0 \le r < \frac{1}{24}, \\ \mathit{distb}_1\left(\frac{24}{11}\left(r - \frac{1}{24}\right)\right) - 1, & \text{if } \frac{1}{24} \le r < \frac{1}{2}, \\ 1 - \mathit{distb}_1\left(\frac{24}{11}\left(\frac{23}{24} - r\right)\right), & \text{if } \frac{1}{2} \le r < \frac{23}{24}, \\ 2 - \left(24\left(1 - r\right)\right)^{1/4}, & \text{if } \frac{23}{24} \le r \le 1. \end{cases} \tag{13}$$

A collection of stratified samples on $[0,1)$, or a set of uniformly chosen samples on the same interval, are mapped to a cubic B-spline distribution via the function D. If the collection of samples is on the unit square $[0,1)^2$, then D is applied separately to each coordinate of each sample.

In Equation (13), the distb_1 function refers to the solution for u in Equation (6). This can be computed algebraically, using Equations (8) through (12), or numerically, as will be shown in the next subsection.

5.3 Numeric Solution for the Second Curve Segment

Using the algebraic solution for the middle curve segments provides a certain satisfaction, but the square and cube roots make it costly to evaluate. The underlying quartic polynomial function

$$f(u) = -3u^4 + 4u^3 + 6u^2 + 4u \tag{14}$$

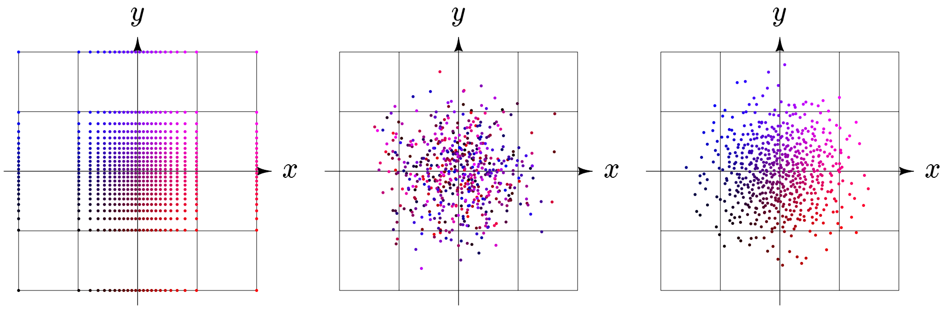
is convex in the interval $[0,1]$ and is in the range $[0,11]$, so that the equation $f(u) = 11r$ can be solved numerically by Newton's iterative root-finding method.

For the first approximation, $u_0 = r$. Then the Newton iterant $u_{i+1} = u_i - f(u_i)/f'(u_i)$ is

$$\begin{aligned} u_{i+1} &= u_i - \frac{11r - 4u_i - 6u_i^2 - 4u_i^3 + 3u_i^4}{-4 - 12u_i - 12u_i^2 + 12u_i^3} \\ &= \frac{11r + u_i^2[6 + u_i(8 - 9u_i)]}{4 + 12u_i[1 + u_i(1 - u_i)]} \end{aligned}$$

(the second equation has the u_i term combined and the numerator and denominator associated for efficient evaluation).

We have found that five iterations is sufficient for double-precision arithmetic. Thus for r in $[0,1)$, the following C code computes the solution u of Equation (14).



(a) Uniform samples (b) Unstratified random samples (c) Stratified random samples

Figure 3: Plots of sampling approaches; there are 24^2 samples in each image.

```
int i;
double u = r;
for (i = 0; i < 5; i++)
  u = (11*r + u*u*(6 + u*(8 - 9*u)))/(4 + 12*u*(1 + u*(1 - u)));
```

Indeed, only four iterations produces a value about as accurate as the algebraic solution. In fact, for this particular problem, our experiments have shown that using Newton's method is simpler, faster, can be implemented with less code, and produces a more accurate result than the algebraic solution. The function f, however, happens to be well suited for Newton's method on $[0, 1]$ in this case.

6 Results

Figure 3 illustrates the sample dsitrubutions of the sampling strategies discussed in this paper. As expected, the stratified samples provide a much better distribution than the unstratified samples. Figure 4 shows a "difficult" image containing very high frequencies sampled using a traditional box filter, compared with a cubic B-spline filter.

(a) Box filter

(b) Cubic B-spline filter

Figure 4: Grayscale renderings of the function $1 + \sin((x^2 + y^2)/100)$ (512×384 pixels, 900 samples per pixel) using (a) a box filter, and (b) a cubic B-spline filter. Notice the cubic B-spline image does not exhibit the interference patterns produced by the box filter.

Appendix: Quadratic and Linear B-Spline Filter Kernals

Quadratic B-Spline Filter

The quadratic B-spline filter kernal is composed of three quadratic polynomial pieces:

$$B_2(t) = \begin{cases} b_0(t + \frac{3}{2}) & \text{if } -\frac{3}{2} \le t \le -\frac{1}{2}, \\ b_1(t + \frac{1}{2}) & \text{if } -\frac{1}{2} < t \le \frac{1}{2}, \\ b_0(\frac{1}{2} - t) & \text{if } \frac{1}{2} < t \le \frac{3}{2}, \\ 0 & \text{otherwise,} \end{cases}$$

where

$$b_0(t) = \frac{1}{2}t^2$$
$$b_1(t) = \frac{1}{2}\left(-2t^2 + 2t + 1\right).$$

Unstratified samples can be produced using Equation (1), with $n = 2$.

Generating stratified samples requires a procedure similar to that of Section 5. The details are omitted, but the following function maps a uniform distribution on $[0, 1]$ to the quadratic B-spline distribution:

$$D_2(r) = \begin{cases} (6r)^{1/3} - \frac{3}{2}, & \text{if } 0 \le r < \frac{1}{6}, \\ \sqrt{3}\cos\left(\frac{\pi}{3} + \frac{1}{3}\arccos\left(\frac{4r-2}{\sqrt{3}}\right)\right) & \text{if } \frac{1}{6} \le r < \frac{5}{6}, \\ \frac{3}{2} - (6\,(1-r))^{1/3} & \text{if } \frac{5}{6} \le r \le 1. \end{cases} \tag{15}$$

The Newton iterant for the middle function is

$$u_{i+1} \leftarrow \frac{8u_i^3 - 12r + 6}{12u_i^2 - 9} \tag{16}$$

with initial approximation $u_0 = (6r-3)/4$. Four iterations in this case is sufficient. Again, Newton's method appears to be faster and more stable than the algebraic solution.

Linear B-Spline Filter

The linear B-spline filter kernal amounts to a common "separated triangle filter", but the details are given here for completeness. The filter kernal is

$$B_1(t) = \begin{cases} 1 + t & \text{if } -1 \le t \le 0, \\ 1 - t & \text{if } 0 < t \le 1, \\ 0 & \text{otherwise.} \end{cases}$$

Inversion is straightforward. The following function produces a linear B-spline distribution from a uniform distribution on $[0, 1]$.

$$D_1(r) = \begin{cases} \sqrt{2r} - 1, & \text{if } 0 \le r < \frac{1}{2}, \\ 1 - \sqrt{2(1-r)}, & \text{if } \frac{1}{2} \le r \le 1. \end{cases} \tag{17}$$

References

[1] William H. Beyer, editor. *CRC Standard Mathematical Tables*. CRC Press, Inc., Boca Raton, FL, 28^{th} edition, 1987.

[2] Murray R. Spiegel. *Schaum's Outline of Theory and Problems of Probability and Statistics*. McGraw-Hill, New York, 1997.