Using the Accumulation Buffer for Visualization

Mike Bailey
mjb@cs.oregonstate.edu

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License

The Framebuffers

Update
Depth
Accumulation
Back
Front
Refresh

Here’s how the Accumulation Buffer works:
1. You can multiply the RGBs in the Back Buffer by a constant
2. You can multiply the RGBs in the Accumulation Buffer by a constant
3. You can add the RGBs in the Back Buffer to the RGBs in the Accumulation Buffer
4. You can copy the Accumulation Buffer to the Back Buffer

Using the Accumulation Buffer to Achieve Motion Blur

1. Draw the new frame into the Back Buffer and multiply all its RGBs by A.
2. Multiple all the Accumulation Buffer’s RGBs by (1. – A) and add the Back Buffer into it (“GL_ACCUM”). Basically, you are blending the new animation frame with a collection of the old frames.
3. Copy (“GL_RETURN”) the Accumulation Buffer to the Back Buffer.
4. Swap the Front and Back Buffers (“glutSwapBuffers”).

The framebuffer starts out as: FB0 = Black
The first frame results in: FB1 = A*F1 + (1-A)*FB0 = A*F1 + (1 -A)*Black
The second frame results in: FB2 = A*F2 + (1.-A)*FB1 = A*F2 + (1.-A)*A*F1 + (1.-A)2*Black
The third frame results in: FB3 = A*F3 + (1.-A) *A*F2 + (1.-A) 2*A*F1 + (1.-A)3*Black

glAccum( GL_MULT, A );
glAccum( GL_ACCUM, 1.-A );
glAccum( GL_RETURN, 1.00 );

A = 0.900
Using the Accumulation Buffer to Achieve Motion Blur

The framebuffer starts out as: $FB_0 = \text{Black}$

The first frame results in: $FB_1 = 0.900*F_1 + 0.100*FB_0 = 0.900*F_1 + 0.100*\text{Black}$

The second frame results in: $FB_2 = 0.900*F_2 + 0.100*FB_1 = 0.900*F_2 + 0.090*F_1 + 0.010*\text{Black}$

The third frame results in: $FB_3 = 0.900*F_3 + 0.090*F_2 + 0.009*F_1 + 0.001*\text{Black}$