COSC422 Advanced Computer Graphics
Programming Exercise 8
Particle Systems

This programming exercise deals with a shader based implementation of a method for rendering particle systems.

PS_Smoke.cpp:

The program `PS_Smoke.cpp` generates the display of a particle system consisting of 10 (variable `NPART`) particles (Fig. 1).

Fig. 1.

Point sprites are used for rendering particles. All particles share the same texture "Smoke1.tga". The parameters of the path for each particle are specified in 6 VBOs (position, velocity, angle, magntitude, frequency, start time). Each particle follows a sinusoidal path defined by the magnitude and frequency values.

The vertex shader (`PS_Smoke.vert`) computes the position of a particle based on the equation given on Slide [3]-34. The size of each particle (`gl_PointSize`) is specified as 50 pixels.

The fragment shader (`PS_Smoke.frag`) texture maps the image onto the point sprite, and assigns a transparency factor of 0.5 for blending the texture mapped images.

Please modify the program to load all four textures "Smoke1.tga"… "Smoke4.tga". Create one more VBO containing a texture index in the range 0..3 for each particle. Please make the necessary changes in the shaders so that all four textures are used in the rendering of the particle system.

Use a linear mapping function (Slide [3]-37) in the vertex shader to increase the size of a particle from 50 to 300 as its distance from the source increases from 0 to DMAX. Similarly, scale the output colour of the fragment shader by a distance based scale factor so that the particle’s colour fades into the white background at maximum distance.

Increase the number of particles to 1000. The program should produce an output similar to that given on Slide [3]-40.

[3]: COSC422 Lecture Slides “3. Sprites and Particle Systems”