

Tarea9

Fire with UV displacement

Instrucciones:

- Leer:
https://docs.gameloft.org/3d-training/#UV_Displacement
- Trabajaras con:
 - Los archivos de geometría (file.nfg)
 - Texturas
 - Fire3.tga
 - Fire_mask.tga
 - DisplacementMap.tga

En general, hay dos formas de generar una animación:

- a) **Animación por vertex**: Se requieren archivos de animación específicos, donde los valores de los parámetros dinámicos son especificados por key frame y una interpolación es hecha entre ellos para obtener el valor de cada frame.
- b) **Animación por material**: Un método menos costoso para obtener una animación, basado en el desplazamiento de las coordenadas uv con un offset diferente en cada frame y usando estas nuevas coordenadas para hacer el sampling.

Material = instancia de un shader.

Por lo tanto, **Material = Shader** (incluyendo los estados del shader) + **valores de las uniform estáticas** (como las texturas y luces, al contrario de las uniforms dinámicas como la world matrix).

- Nosotros **utilizaremos Animación por Materia** para generar el efecto de fuego.

Guía:

1. Como has visto en el mundo real, una fogata de campamento tiene mayormente un movimiento vertical. La física explica este comportamiento debido a la elevación de las partículas incandescentes.

Por lo tanto, nosotros necesitaremos generar este tipo de animación. Para esto necesitaremos la textura de desplazamiento:

Vec2 disp = texture2D(Displ_tex, vec2(Tex_coord_original.u, Tex_coord_original.v + u_Time)).xy

Dónde:

- **disp** da el desplazamiento para u y v. Por el hecho que el mapa de desplazamiento se está moviendo sobre la superficie del objeto (el movimiento es dado por `Text_coords_original.v + u_Time`), se creara la ilusión de movimiento de las partículas del objeto en la misma dirección (en este caso la ilusión de movimiento en la dirección "v").
 - **u_Time** es el tiempo de la aplicación, se acumula el `deltaTime` de `NewTrainingFramework.cpp::Update()`. Se aplica solo sobre "v" porque la mayoría del movimiento será vertical.
 - **disp.x** y **disp.y** están en [0,1] (la lectura desde una textura siempre producirá un valor entre 0 y 1)
2. El movimiento del fuego es aleatorio (movimiento browniano), por lo que el movimiento en cualquier dirección es igualmente probable.

Para obtener este movimiento, necesitamos a disp entre los valores [-1,1]

Por lo tanto, disp cambiara desde [0,1] → [-1,1] con esta fórmula:

$$2.0 * \text{disp} - 1.0$$

3. El movimiento del fuego será en cada dirección con una amplitud (dMax)

Por lo que, el offset será:

$$\text{offset} = (2.0 * \text{disp} - 1.0) * \text{dMax}$$

$$\text{new_uv} = \text{v_uv} + \text{offset}$$

4. Usa **new_uv** para hacer el sampling de fire3.tga
5. Para tener un fuego más vistoso usaremos la textura fire_mask (una máscara en escala de grises).

```
vec4 AlphaValue = texture2D(FireMask, Text_coods_original);
```

```
glFragColor = fire_color *(1.0, 1.0, 1.0, AlphaValue.r);
```

- Necesitamos la uv original para hacer el sampling desde fire_mask por que la cúpula debe de estar en la misma posición.
- No reemplazaremos el canal alpha de fire3.tga con el alpha de fire_mask, solo los combinaremos, porque la información de los dos es relevante.

6. El Wrapping Mode de las texturas:

- GL_CLAMP_TO_EDGE para fire3.tga y fire_mask.tga (estas texturas no son tiliables, si usas GL_REPEAT se verán cosas mal en pantalla)
- GL_REPEAT para Displacement.tga

7. El fuego es un objeto con transparencia:

- Tendrá que renderizarse después de que todos los objetos opacos hayan sido renderizados, por lo tanto sitúa tu fuego hasta el final de tu SM.txt
- Necesitaras activar el blending cuando vayas a renderizar el fuego y poner la función de blending a alpha blending

```
glEnable(GL_BLEND);
```

```
glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA);
```

Resultado:

