Per-fragment Effects
Part II: Texture Based Animation

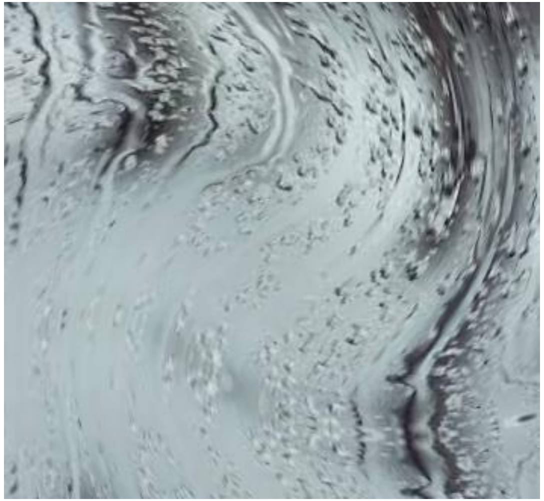
Basic Idea

- Each texel is associated with a fragment and the change of the texture coordinates associated with a fragment will change the colour value associated with the vertex

- By just altering the sampling texture coordinates, it is possible to animate certain dynamic behaviours of an object using a texture
  - Depending on the use of a texture, the animation can be:
    - Texture colour based
    - Texture coordinates based

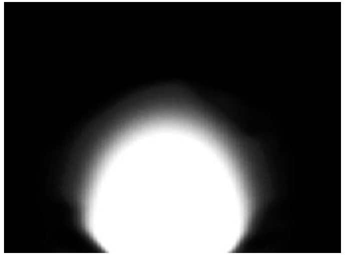
Image Transformation

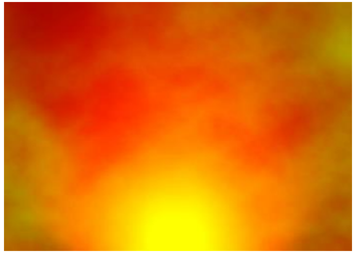
- An image can be transformed by transforming texture coordinates, either uniformly or locally.
  - Translation
  - Rotation
  - Scaling
  - … …

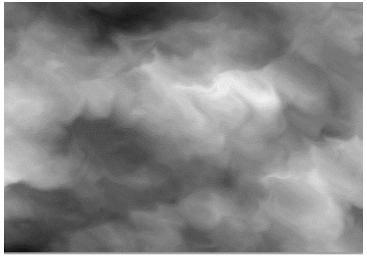


Example: Texture Based Fire Animation

- Use THREE textures respectively to specify:
  - Fire shape
  - Fire intensity
  - Noise
- Noise texture is used to alter fire shape and fire intensity







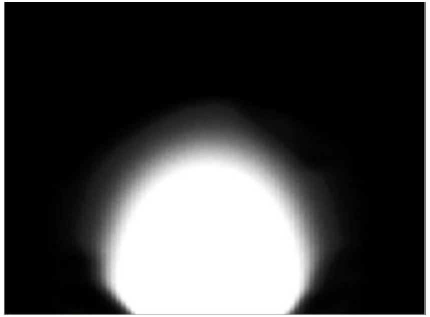
Example: Image Based Fire Animation
—Specify Basic Fire Shape

- Introduce a timer

```
float time;
```

- Specify the overall shape of the fire using a texture

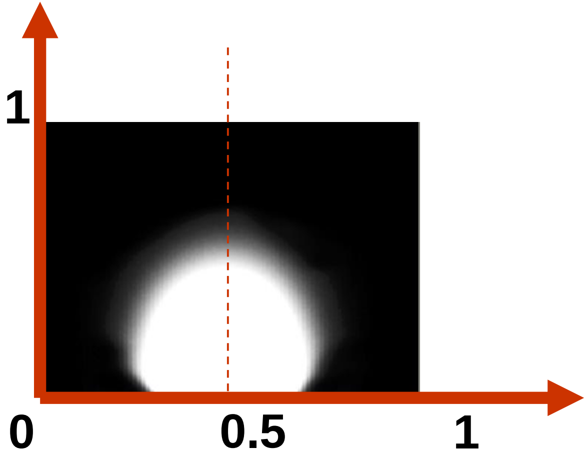
```
    vec4 shape = texture2D(fire_shape, tex);
```

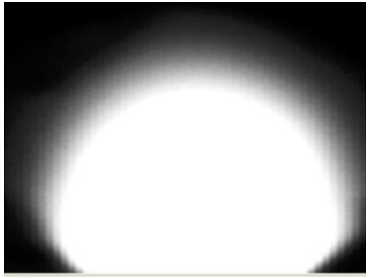


Example: Image Based Fire Animation —Control Fire Shape

- Control the size of the fire shape by scaling the fire shape texture in the following way:

```
vec2 tex = TexCoord;
tex.y = 1.0 - tex.y;
tex.x = Scale.x*(TexCoord.x-0.5) + 0.5;
 tex.y= Scale.y*tex.y;
```



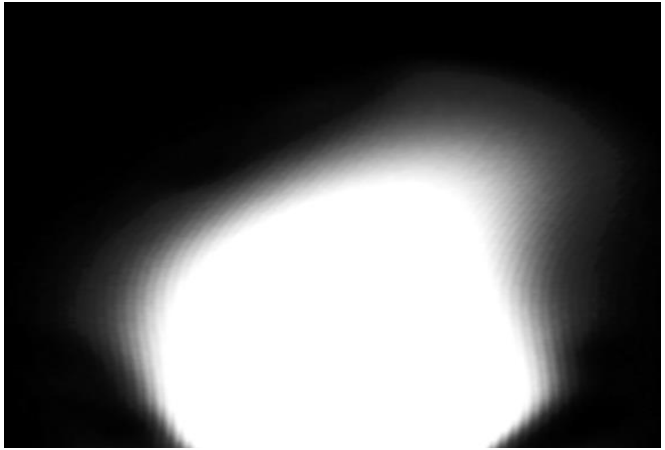


Example: Image Based Fire Animation

—Make Fire Shape Wavy

- Perturb fire shape along x-axis and y-axis:

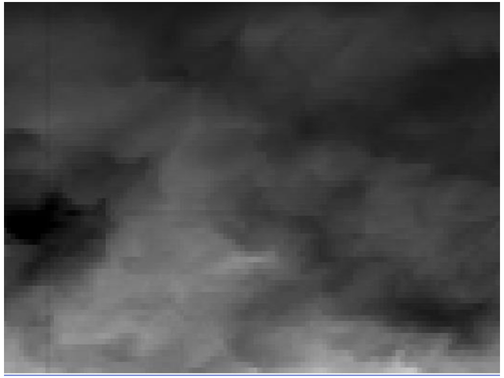
```
tex.x += 0.1*tex.y*sin( freq.x *(TexCoord.y + time));
tex.y += 0.1*tex.y*sin( freq.y *(TexCoord.x + time));
```



Example: Image Based Fire Animation —Add some noise

- shooting the texel upward:

```
vec4 noise = texture2D(fire_noise, TexCoord+speed*time);
```

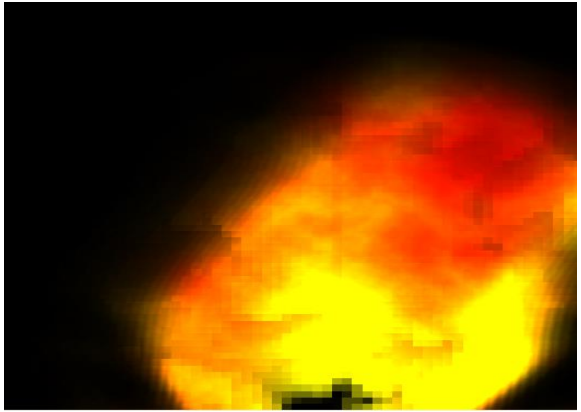


Example: Image Based Fire Animation

—Control Fire Intensity

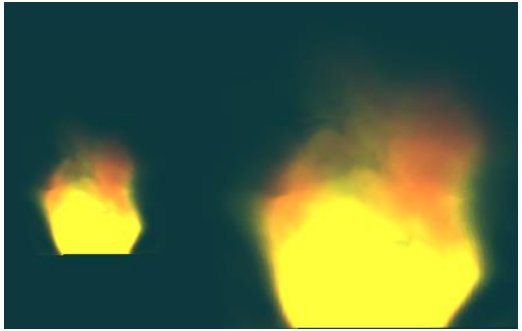
- Modify fire intensity using the noise and shape textures

```
gl_FragColor = 2.0*TexCoord.y*shape*fireColor*noise;
```

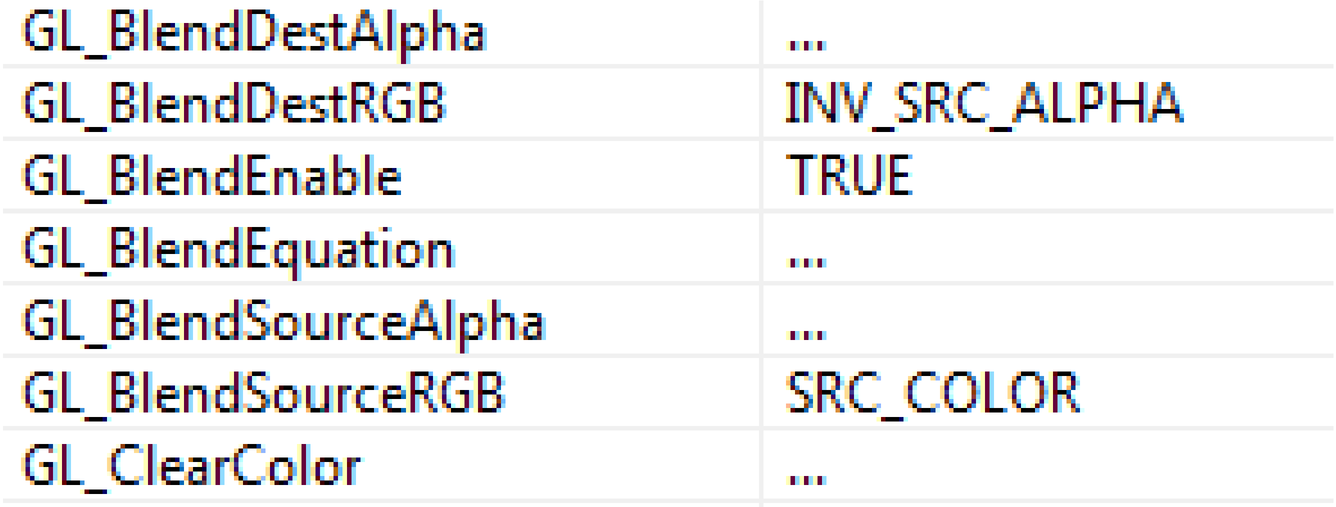


Instancing

- Render fire into texture
- Mapping the rendered fire texture on to a quad texture
- Enable colour blending



Configure Render States

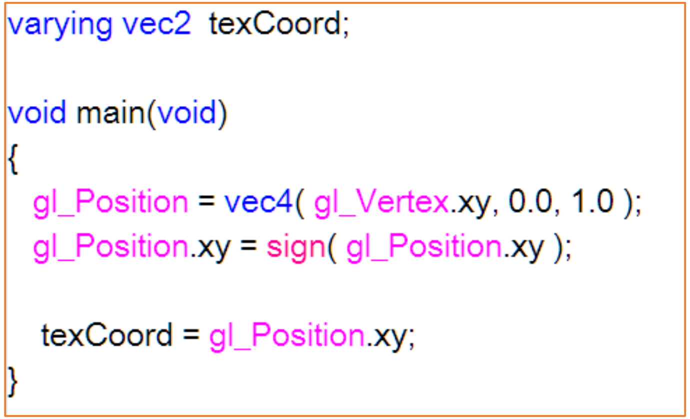


GL_BlendDestAlpha	...
GL_BlendDestRGB	INV_SRC_ALPHA
GL_BlendEnable	TRUE
GL_BlendEquation	...
GL_BlendSourceAlpha	...
GL_BlendSourceRGB	SRC_COLOR
GL_ClearColor	...

Procedural Textures

Create a Canvas

- Vertex shader
  - Specify a quad of size [-1,1]x[-1, 1]

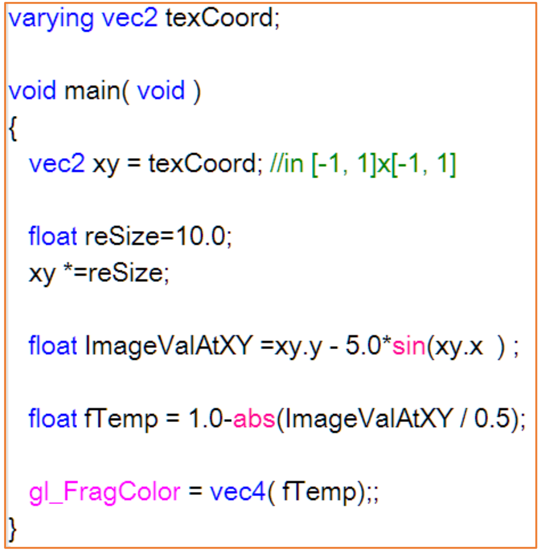


```
varying vec2  texCoord;

void main(void)
{
   gl_Position = vec4( gl_Vertex.xy, 0.0, 1.0 );
   gl_Position.xy = sign( gl_Position.xy );

    texCoord = gl_Position.xy;
}
```

- Fragment shader
  - Resize the square to a required canvas size



```
varying vec2 texCoord;

void main( void )
{
   vec2 xy = texCoord; //in [-1, 1]x[-1, 1]

   float reSize=10.0;
   xy *=reSize;

   float ImageValAtXY =xy.y - 5.0*sin(xy.x  ) ;

   float fTemp = 1.0-abs(ImageValAtXY / 0.5);

   gl_FragColor = vec4( fTemp);;
}
```

Regard a Texture as Function Image(x,y)

Fragment shader

```
varying vec2 texCoord;

vec3 ImageColorAt( vec2 pixelXY)
{
            … …
            … …
}
```

Fragment shader

```
void main( void )
{
 vec2 xy = texCoord;

   float reSize=10.0;
   xy *=reSize;
   vec3 color = ImageColorAt(xy);
   gl_FragColor = vec4(color, 1.0);
}
```

Example 1: Mandelbrot Fractal

- For each point **P**(x,y) on the plane, define a complex number

  $$\mathbf{C} = x + y\,\mathbf{i}$$

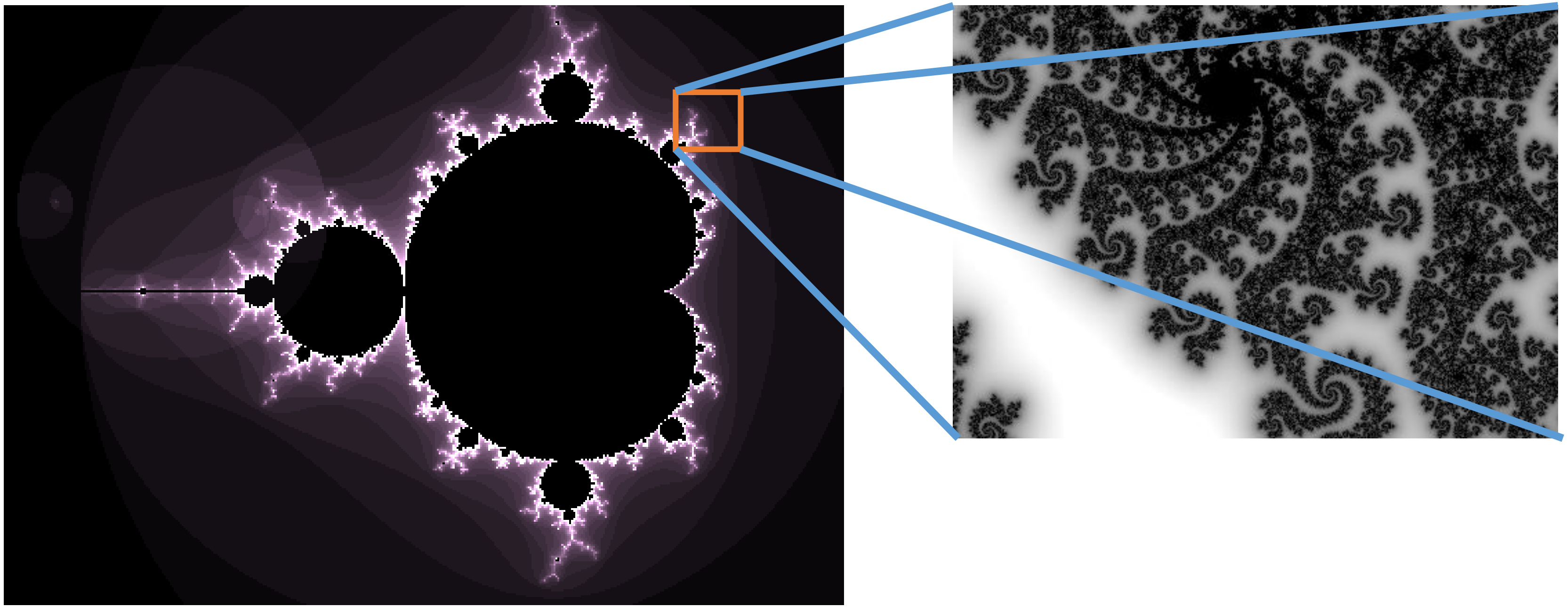
- Let $Z_0=(0, 0)$, then a sequence of complex numbers Z_1, Z_2, … can be generated in the following way:

  $$Z_{n+1}=(Z_n)^2 + C$$

Example 1: Mandelbrot Fractal

- Can also be described using transformations:
  - For each point (x,y) in the plane
    - set vector **C**=(x,y)
    - Starting from vector $\mathbf{P_0}$=(0,0)
    - For n>0, find vector $\mathbf{P_{n+1}}$ in the following way:
      - Rotate $\mathbf{P_n}$ about the origin by an angle formed between $\mathbf{P_n}$ and the x-axis
      - Then Scale $\mathbf{P_n}$ by : $|\mathbf{P_n}|$
      - Finally translate $\mathbf{P_n}$ by vector **C**

Example 1: Mandelbrot Fractal



Questions?