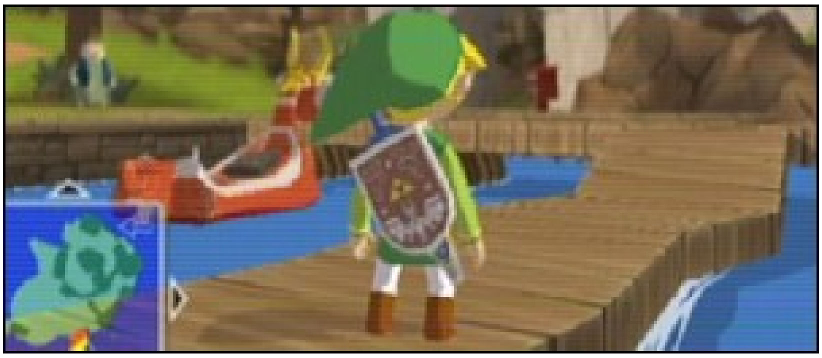
Rendering Style: Toon Shading

- **Toon Shading**

In computer graphics, rendering is generally defined as “p rocess of generating an image from a model, by means of software programs.” Among many rendering styles, in this paper, we will particularly focus on Toon Shading. In brief, Toon Shading, also known as Cel Shading, is a simplest non-photorealistic rendering technique to give mesh a cartoon look and the impression of hand drawn models in the final render. In animation field, Toon Shading technique has been used for years and recently applied to video game field. Many Japanese animated movies in modern time used this technique, while in the video game field, only few non-animation games applied Toon Shading such as Legend of Zelda: The Wind Waker, Viewtiful Joe, and Jet Set Radio Future.

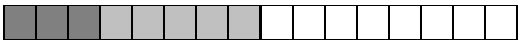


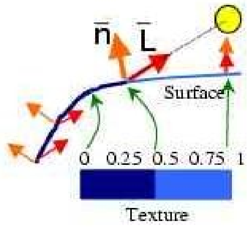
- **Pros and Cons of Toon Shading**

Toon Shading has advantage of giving more cartoon effect than flat shading, and better FPS and less complex texture so that GPU does not need to process high VRAM using textures. However, Toon Shading has little disadvantage of when it renders sophisticated objects such as human characters. It gives glossy feature for rendering human characters since shading itself is too simple as well as it uses only 1D image texture. In addition, all parts of the meshes are rendered by same parameters so that it’s hard to stress and express certain parts in detail. In fact, it's more suitable rendering simple design objects such as robots and cars.

- **Toon Shading Technique**

1. First of all, creating a 1x32 pixel grey scale texture map which determines the shading of the mesh.





2. Secondly, compute the dot product of the light vector and normal vector of each vertex. Dot product of light vector and normal vector produces a value between 0 and 1, which is used as a texture coordinate to map to the grey scale texture map produced above. If the value is less than zero, we clamp the value to zero. When the normal vector is close to light vector, the dot product of two vectors is close to 1 and uses brighter color. On the other hand, when the angle between normal vector and light vectors are far, the dot product of two vectors yields to zero and uses darker color.

- **Toon Shading Using GLSL**

-Vertex Shader: Normal transformation per vertex.

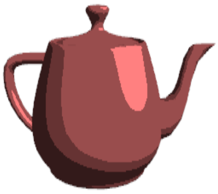
```
varying vec3 normal;
void main()
{
   normal = gl_Normal;
   gl_Position = ftransform();
}
```

-Fragment Shader: Compute intensity by computing dot product of light vector and normal vector.

```
uniform vec3 lightDir;
```

```
varying vec3 normal;
void main()
{
  float intensity;
  vec4 color;
  intensity = dot(lightDir,normalize(normal));
  if (intensity > 0.95) color = vec4(1.0,0.5,0.5,1.0);
  else if (intensity > 0.5) color = vec4(0.6,0.3,0.3,1.0);
  else if (intensity > 0.25) color = vec4(0.4,0.2,0.2,1.0);
  else color = vec4(0.2,0.1,0.1,1.0);
  gl_FragColor = color;
}
```

- Examples of Toon Shading



References:

http://en.wikipedia.org/wiki/Rendering_(computer_graphics)

http://www.lighthouse3d.com/opengl/glsl/index.php?toon2

http://www.hitlabnz.org/fileman_store/TR-2005-4.pdf

http://www.sfu.ca/~ysl/NPR/cel_shading.html

http://www.postulate.org/silhouette.php

http://www.urbandictionary.com/define.php?term=cel-shade