Making Faces – Eve Online's New Portrait Rendering

Bert Peers
CCP Games, Reykjavik, Iceland
bpeers@ccpgames.com

1. Introduction

Eve Online is a single shard MMO with tens of thousands of spaceships battling it out. Recently, its character customization system has seen a major upgrade, providing the players with a high definition avatar rendering system to sculpt, customize and tweak their in-game characters.

This customization runs at interactive rates in the client, with a static portrait rendered using the same system with higher settings at the online render farm.

In this presentation we survey the array of tricks and techniques used to sculpt and render high quality skin, hair, eyes and shadows. We break down a final render into its individual components, show what worked and what didn't, and point out a few engine gems. Even if the techniques are relatively low key in order to get good results out of commodity hardware, we hope to offer one or two new tricks to everyone in the audience.

2. Contributions

Eve Online uses an established engine that was initially geared towards rendering massive space battles. In this 20 minute talk we give an overview of the changes and additions made to introduce high quality avatar rendering into this system.

2.1 Elaboration

We start with a quick recap of proper linear rendering [GRITZ08], filmic tonemapping [HABLE10], and the skin specific specular model we used [KELEMEN01].

Next, we discuss our implementation of diffuse subsurface scattering [D'EON08] with a Poisson-sampled texture-space light diffusion [HABLE09]. So called *renderjobs* automatically translate rendering algorithms expressed in Python into a *display-list*-like recording [OPENGL]; this gives us very quick prototyping and iteration capabilities at minimal loss in runtime performance.

Subtle facial animation requires a good wrinkling system. Python code monitors animated bone positions to independently activate any of 12 facial zones per frame. An active zone will call on the shader to apply a normal offset and skin recoloring to simulate proper skin deformation.

Turning to shadow rendering, we survey a few techniques that were tested, including Variance Shadow Mapping [LAURITZEN08, CHEN09] and Fourier Opacity Mapping [JANSEN10], before discussing our PCF solution. A tightening of the scene OBB to the view frustum leads to a low complexity but satisfying improvement in shadow quality. For hair and glasses, a screendoor technique is used to emulate transparent shadows [REEVES87].

Wrapping up the shader part with a few small but effective tricks hiding in the hair and the eyes, we turn to our use of blendshapes to improve customizability of the player-sculpted faces.

Using more than 130 morphtargets rules out a slider-based approach, so we opt for *triangle-fields* instead. We augment the face with multiple 2D triangle meshes, where each vertex represents a predefined set of morphtarget weights. As the user moves the mouse, barycentric coordinates are used to interpolate a new set of blendweights, creating the illusion of directly sculpting the face. A similar approach using animation-blendtree weights provides the ability to pose the face and change the facial expression. Finally, we use blendshapes to push vertices behind obstructing geometry, offering the player easy-to-author tucking options for shirts and pants.

Figure 1. *Player created portrait, rendered on the server.*

3. Results

Using these techniques we achieve high quality character customization on commodity hardware, using DirectX9. By simply increasing a few basic settings such as input texture resolution and shadow map resolution, the same rendering system works as a render farm producing high definition player portraits such as Figure 1.

References

[GRITZ08] "THE IMPORTANCE OF BEING LINEAR", LARRY GRITZ ET AL, *GPU Gems 3.*

[HABLE10] "UNCHARTED 2 : HDR LIGHTING", JON HABLE, *GDC 2010.*

[KELEMEN01] "A MICROFACET BASED COUPLED SPECULAR MATTE BRDF MODEL WITH IMPORTANCE SAMPLING", KELEMENT ET AL, *Eurographics 2001.*

[D'EON08] "ADVANCED TECHNIQUES FOR REALISTIC REAL-TIME SKIN RENDERING", EUGENE D'EON, *GPU Gems 3.*

[HABLE09] "FAST SKIN SHADING", JON HABLE, *ShaderX7.*

[OPENGL] "DISPLAY LISTS", OPENGL 1.1 SPECIFICATION

[LAURITZEN08] "SUMMED-AREA VARIANCE SHADOW MAPS", ANDREW LAURITZEN, *GPUGems3.*

[CHEN09] "LIGHTING RESEARCH AT BUNGIE", CHEN ET AL, *Siggraph 2009 Course.*

[JANSEN10] "FOURIER OPACITY MAPPING", JANSEN ET AL, *I3D 2010.*

[REEVES87] "RENDERING ANTIALIASED SHADOWS WITH DEPTH MAPS", REEVES ET AL, *Siggraph 1987.*

SIGGRAPH 2011, Vancouver, British Columbia, Canada, August 7 – 11, 2011.
ISBN 978-1-4503-0921-9/11/0008