Controlling the Dark Side in Toon Shading

Ole Gulbrandsen*

1 Introduction
Sharply separating a diffuse surface into a light and dark side often results in unwanted details. Combining normals from the actual surface with the normals from a simplified surface we get better control of the dark side.

2 Theory
Our solution is to create a shell covering the actual geometry (as in figure 1), and store the shell normals in a point cloud.

figure 1: Sampling normals from an outer shell

The surface shader samples the point cloud and calculates a new normal (1) which is then used as the new shading normal. The weighting between the shell normals and the actual surface normal can easily be controlled by storing $\beta$ in a texture map.

$$n_{avg} = \frac{1}{N+1} \left( \beta n_{surface} + (1-\beta) \sum_{i=1}^{N} n_{ptc_i} \right)$$ (1)

3 Implementation

```cpp
surface bakeNormals(string Filename=""; string CoordSys="")
{
    bake3d( Filename, "", P, n,
        "coordsystem", CoordSys,
        "interpolate", 1,
    );
}
surface readNormals(string Filename=""; float b=0.5;
    string CoordSys="")
{
    uniform string category = concat(        "pointcloud",";",
        Filename );

    point p_ptc = transform(CoordSys,P);
    normal n = normalize(N);
    normal Nsample,Navg,n_avg,n_shade=0;
    float Samples=8;
    float MaxDist=1;
    gather(category, Pworld, n, PI/2,Samples,
        "maxdist",MaxDist,"point:normal", Nsample ){
        Navg += Nsample;
    }
    n_avg = ntransform(CoordSys,"current",Navg);
    n_shade = b*n+(1-b)*n_avg;
    //pass n_shade on to the toon shader
}
```

*e-mail: olegul@hotmail.com*