Charles Hancock

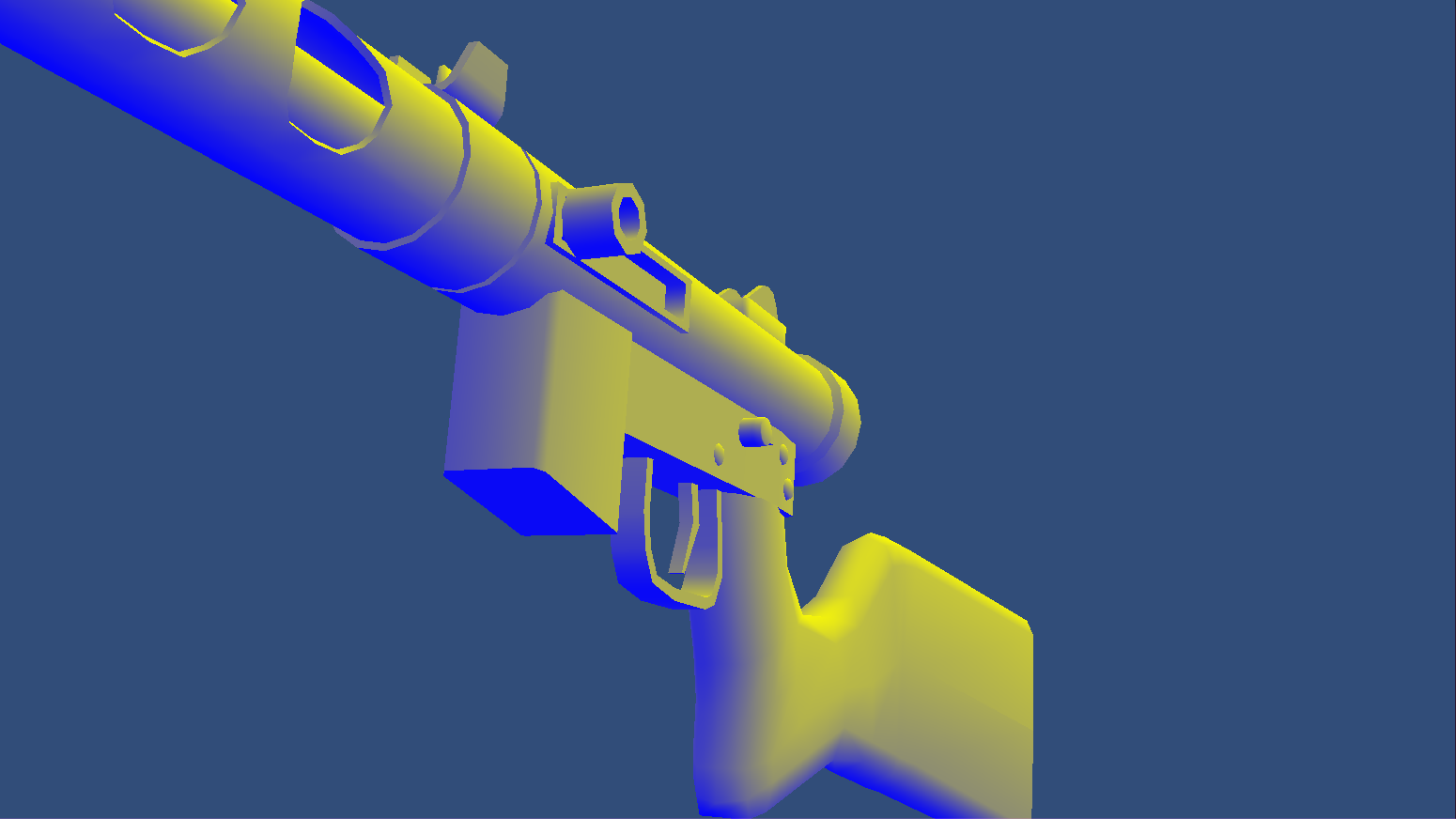
Homework 2

CS 4803

Aaron Lanterman

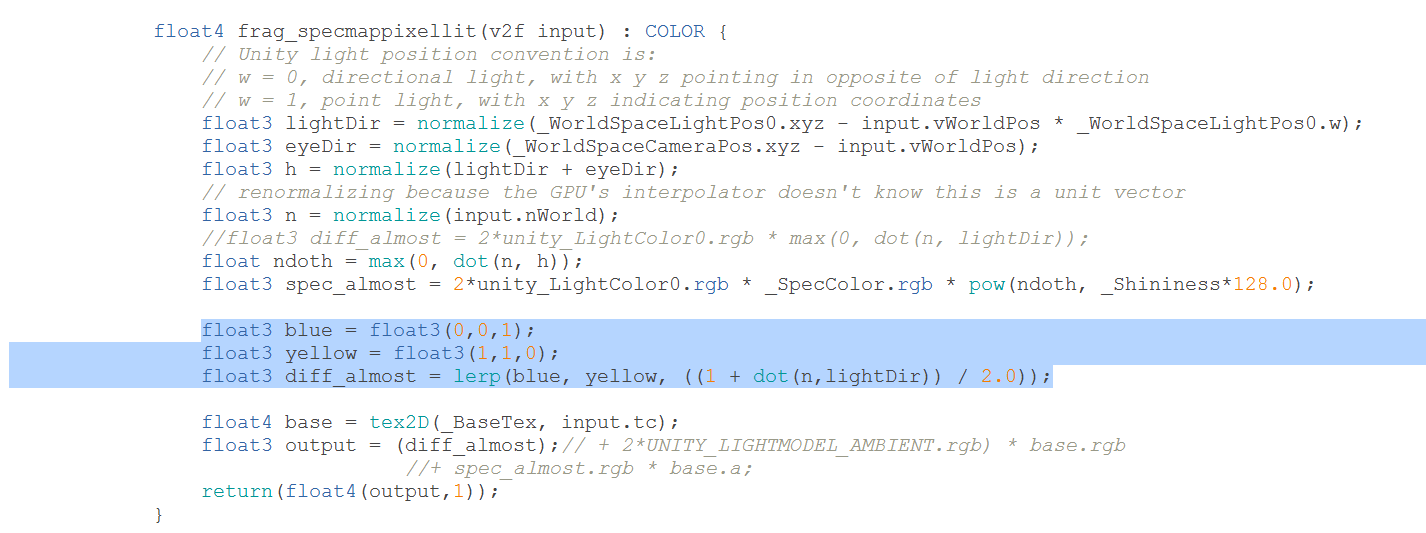
6/20/2014

PROBLEM 1

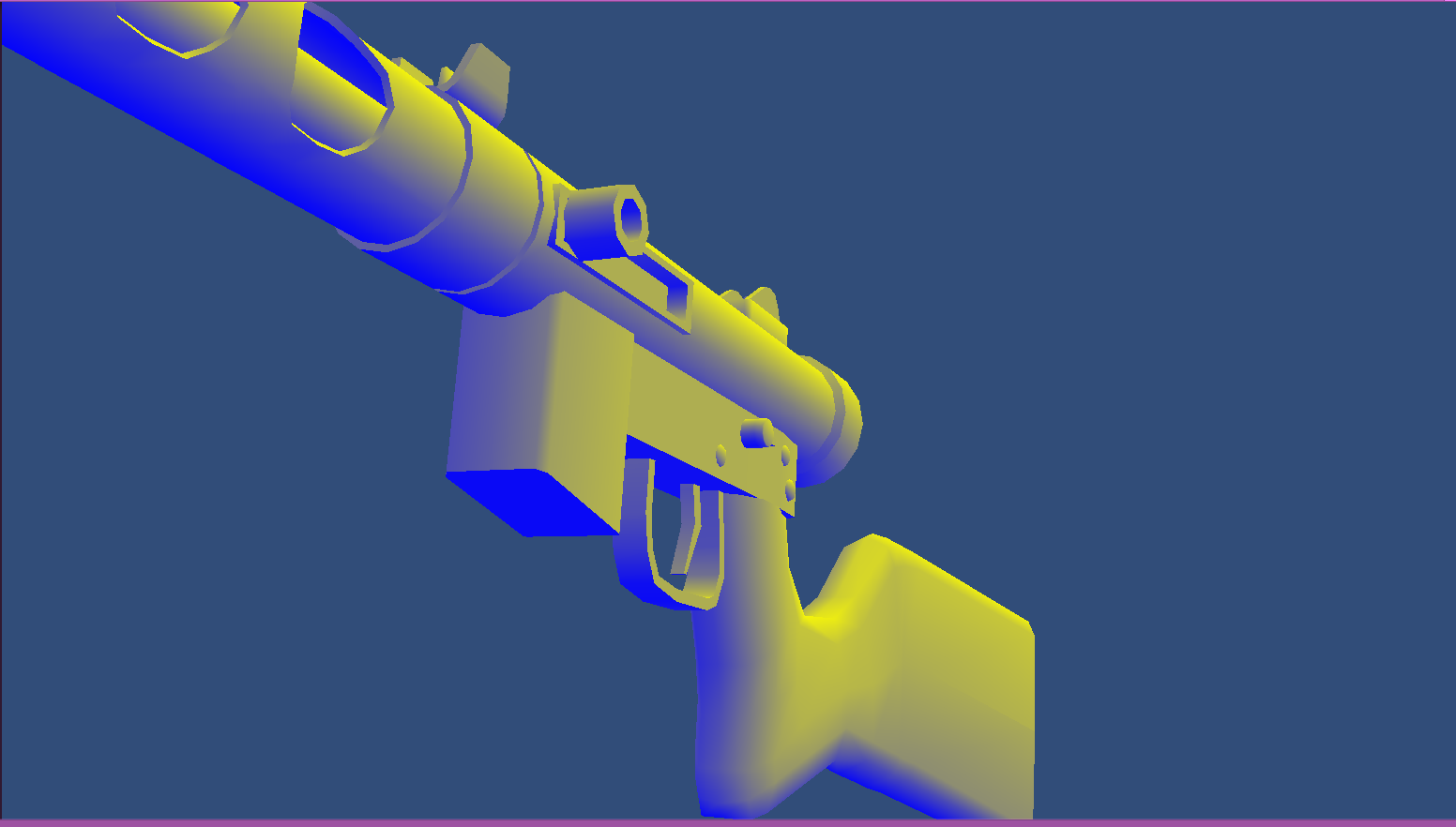


Pixel-lit

**GPUXXSpecmapPixelLit.shader**:

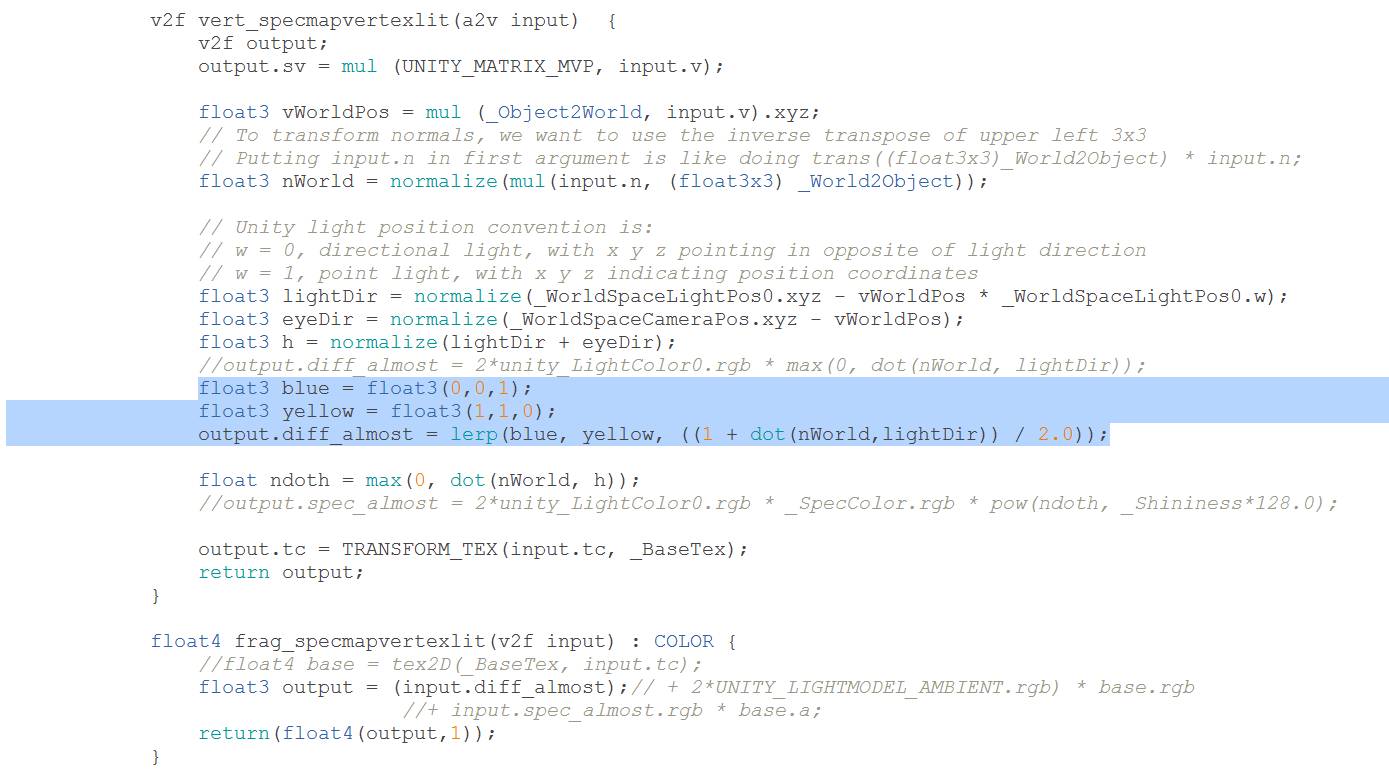


Note the commented out specular and ambient code at the bottom



Vertex Lit (they are ever so slightly different)

**GPUXXSpecmapVertexLit.shader**:



Also note the commented-out ambient/specular code at the bottom

PROBLEM 2



GPUXXTexturedTileCorrectly.shader:

struct v2f { // vertex to fragment

float4 sv: SV\_POSITION;

float2 tc: TEXCOORD0; // not same as TEXCOORD0 above

float depthFactor: TEXCOORD1;

};

v2f vert\_texturedstruct(a2v input) {

v2f output;

output.sv = mul(UNITY\_MATRIX\_MVP, input.v);

float e = 2.8;

float s = 2;

output.depthFactor = max(0,min(1,(e-output.sv.z)/(e-s)));

// Make sure you TRANSFORM\_TEX the vertex shader, not the

// fragment shader!

output.tc = TRANSFORM\_TEX(input.tc, \_BaseTex);

return output;

}

PROBLEM 3



Before



After

GPUXXTexturedStruct.shader:

v2f vert\_texturedstruct(a2v input) {

v2f output;

output.sv = mul(UNITY\_MATRIX\_MVP, input.v);

output.tc = input.tc;

float AA = 1;

float B = 10;

float C = .5;

float D = 1;

float E = 10;

float F = .5;

output.tc.x += AA\*sin(B\*input.tc.y)\*sin(C\*\_Time.x);

output.tc.y += D\*sin(E\*input.tc.x)\*sin(F\*\_Time.x);

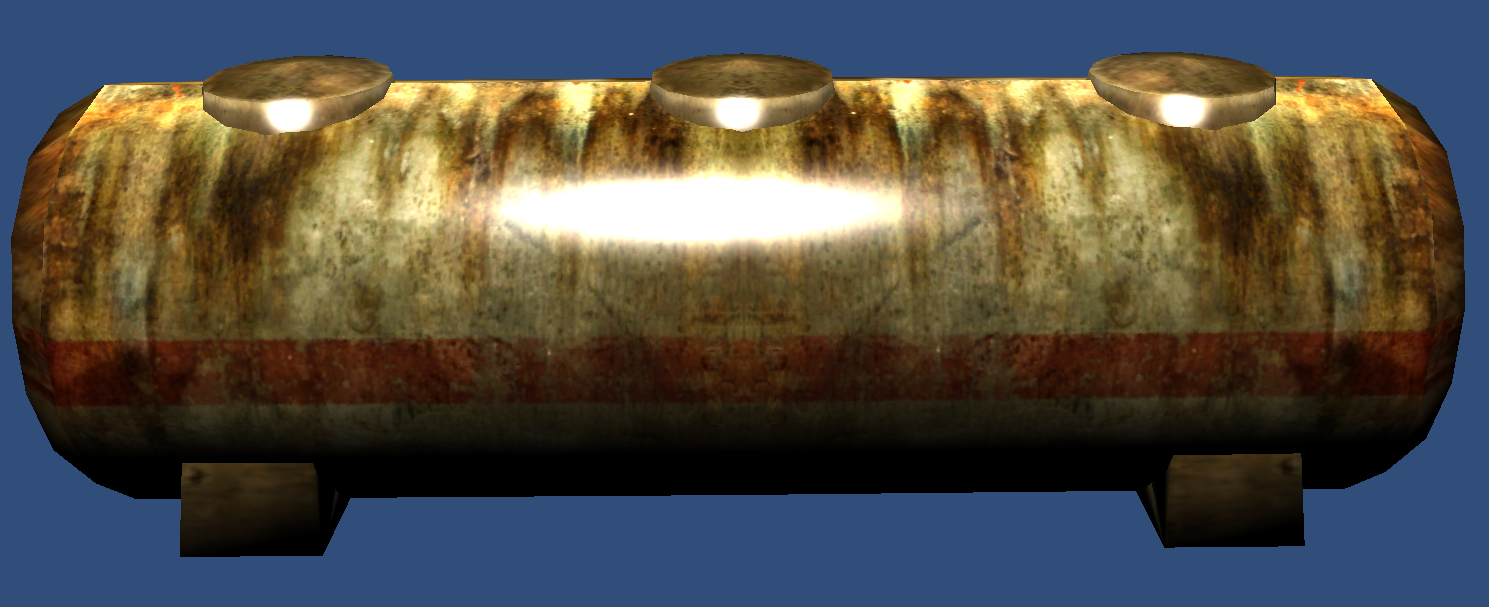
return output;

}

**PROBLEM 4**

****

**Before**

****

**After**

**GPUXXSpecmapPixelLit.shader:**

v2f vert\_specmappixellit(a2v input) {

v2f output;

float AA = 1;

float BB = 10;

input.v += AA \* float4(input.n.x, input.n.y, input.n.z, 0) \* (1+sin(BB\*\_Time.x));

output.sv = mul(UNITY\_MATRIX\_MVP, input.v);

// To transform normals, we want to use the inverse transpose of upper left 3x3

// Putting input.n in first argument is like doing trans((float3x3)\_World2Object) \* input.n;

output.vWorldPos = mul(\_Object2World, input.v).xyz;

output.nWorld = normalize(mul(input.n, (float3x3) \_World2Object));

output.tc = TRANSFORM\_TEX(input.tc, \_BaseTex);

return output;

}