A1-REPORT

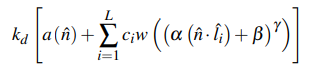
Evyn Brouwer – 100702629

Anthony Smiderle – 100695532

Daniel Hong – 100623669

MATH

TF2 Lighting (Unimplemented):



kd = albedo of the object sampled from the texture map

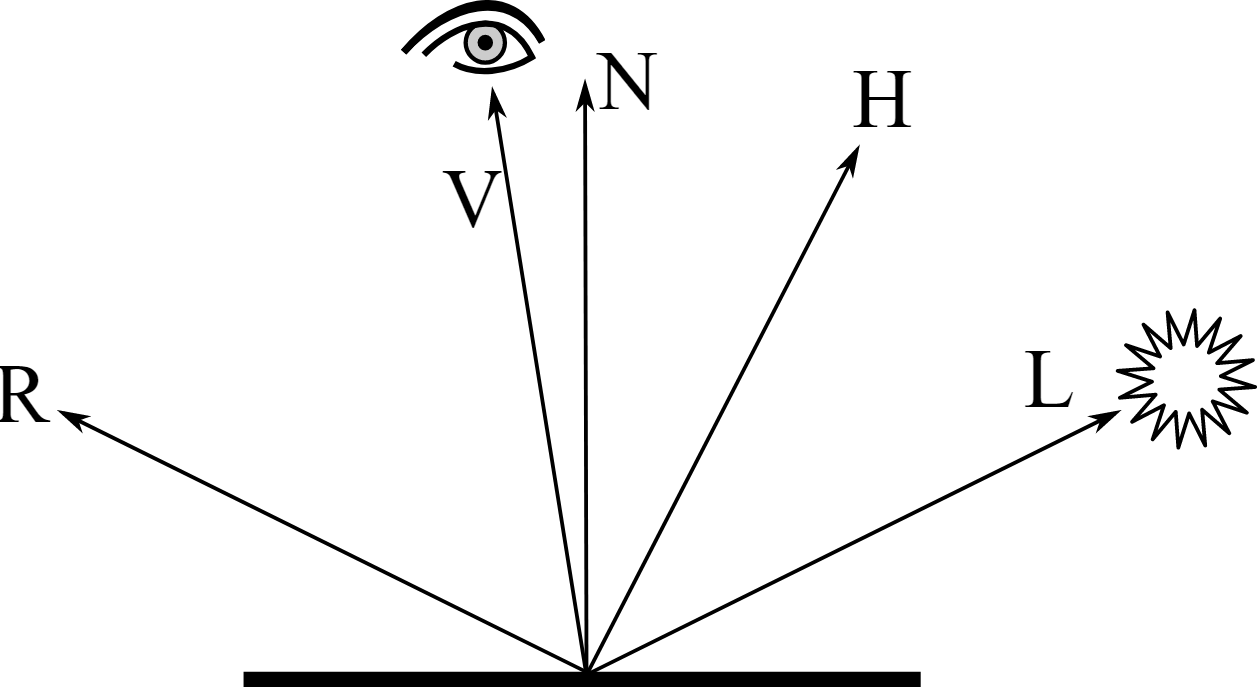
a(n) = directional ambient lighting on the object as a function of the per pixel normal n

∑Li=1 = the summation of the following function for L = number of lights and i = light index

ci = color of the light i

w = wrapping function that applies a 1D LUT and multiplies the color output by 2 before clamping to 0-1 range

n dot li = Lambert term found through (N dot L) based on the following diagram:

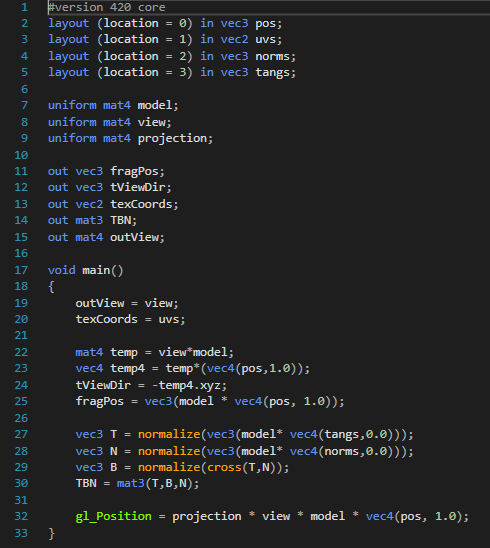


a, b, y = scale, bias, and exponentiation which are set to 0.5, 0.5, 1 for the purpose of TF2 lighting

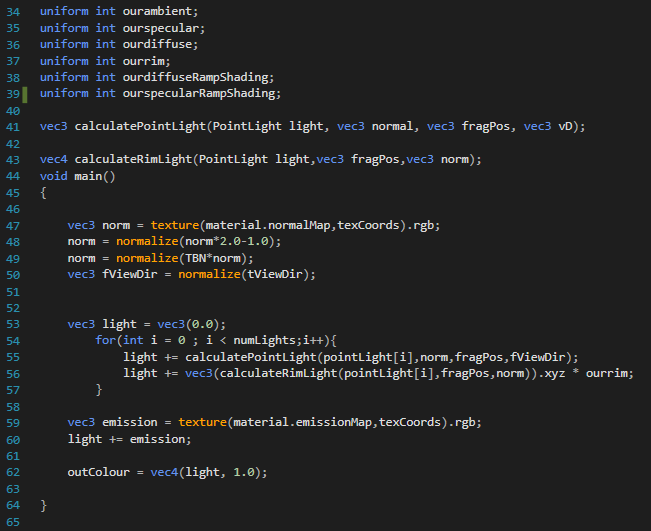
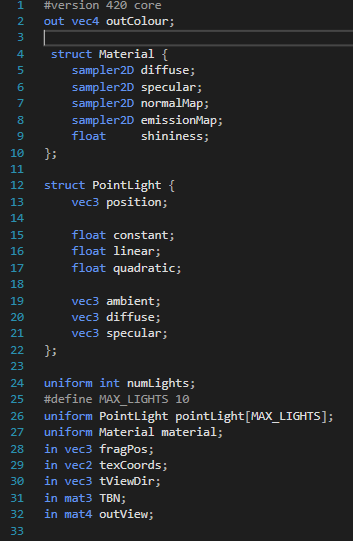
ALGORITHM

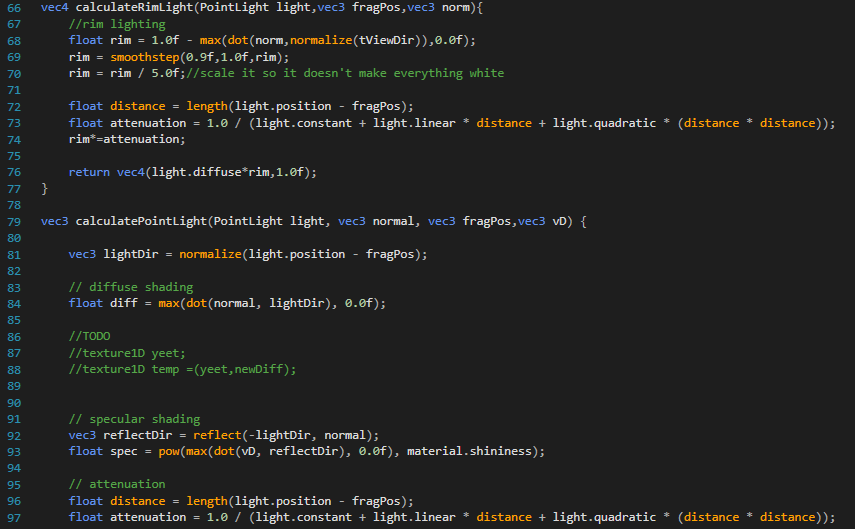
CODE

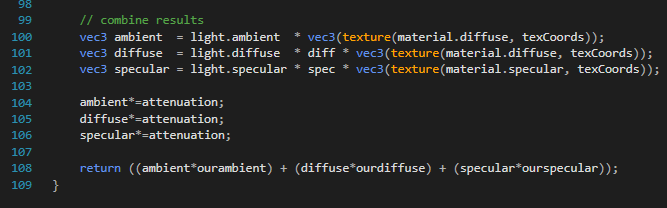
Vertex



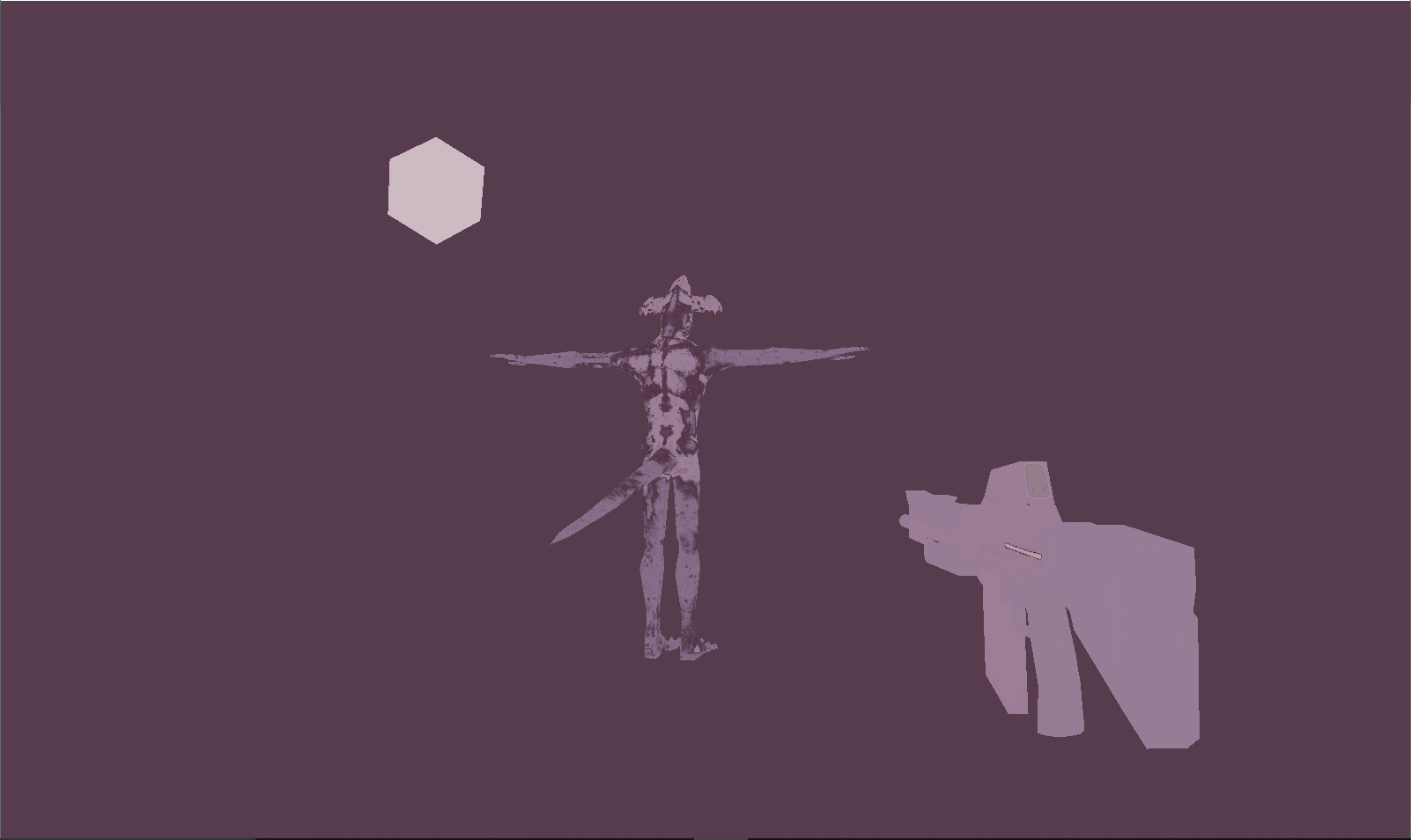
Fragment



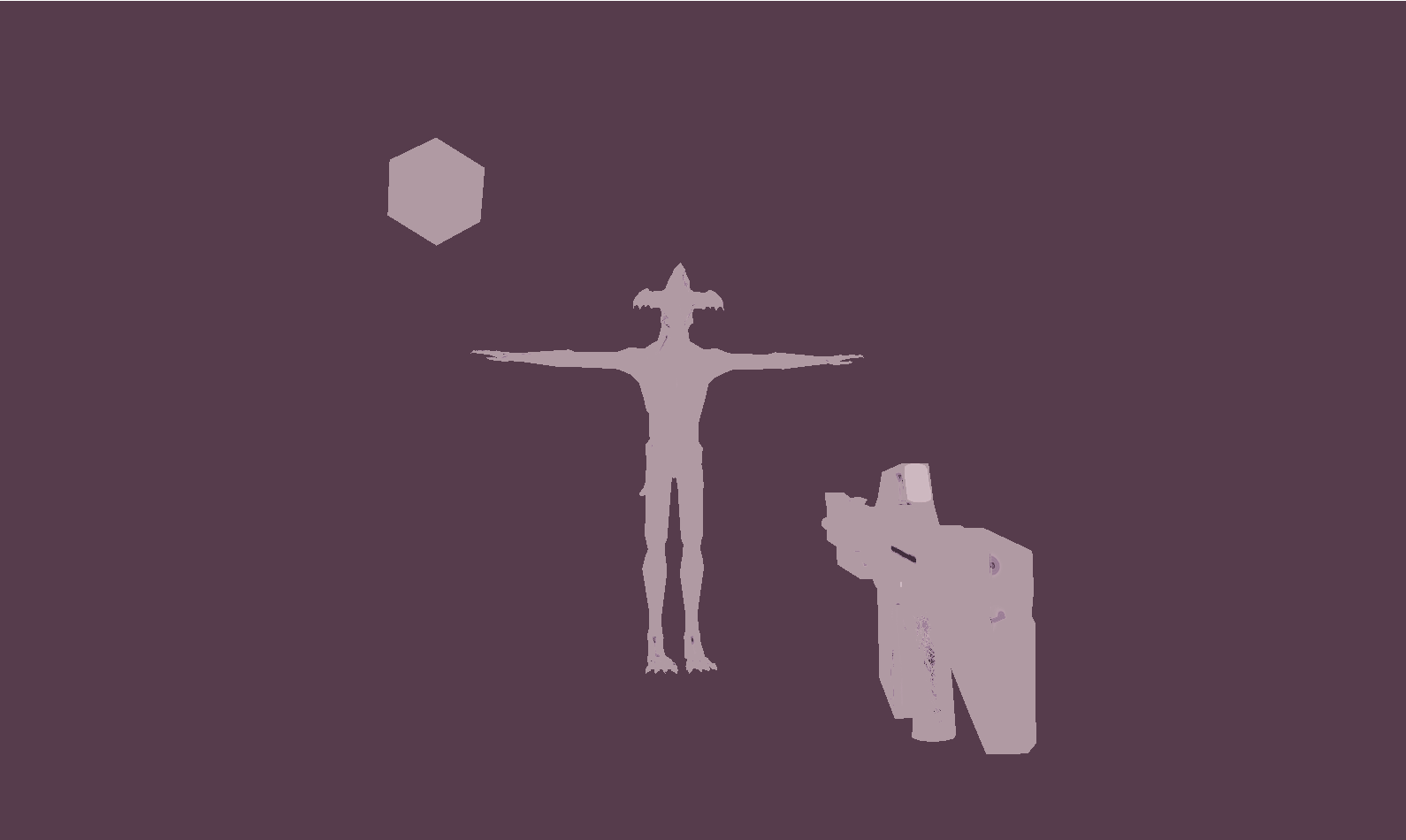
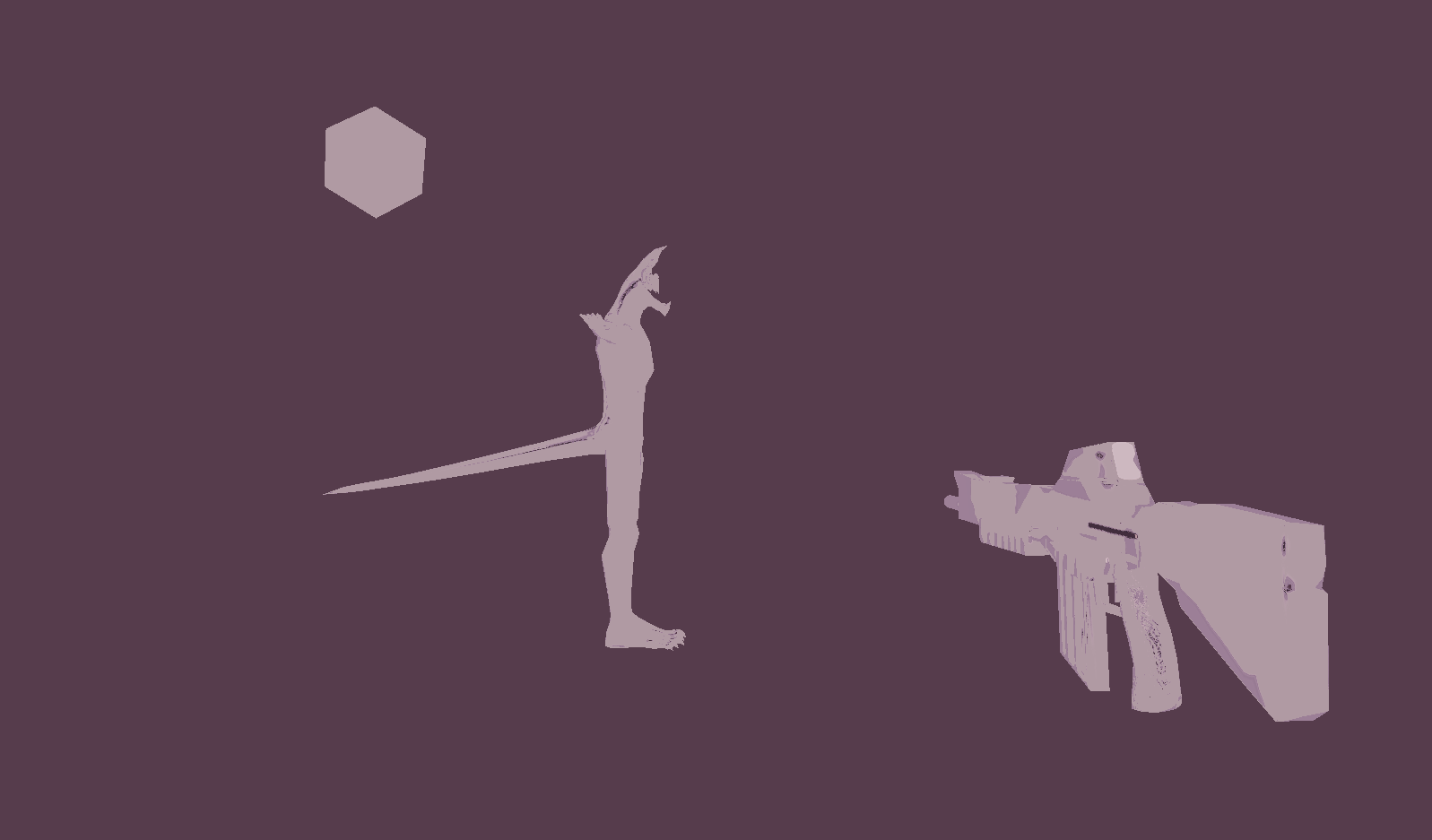




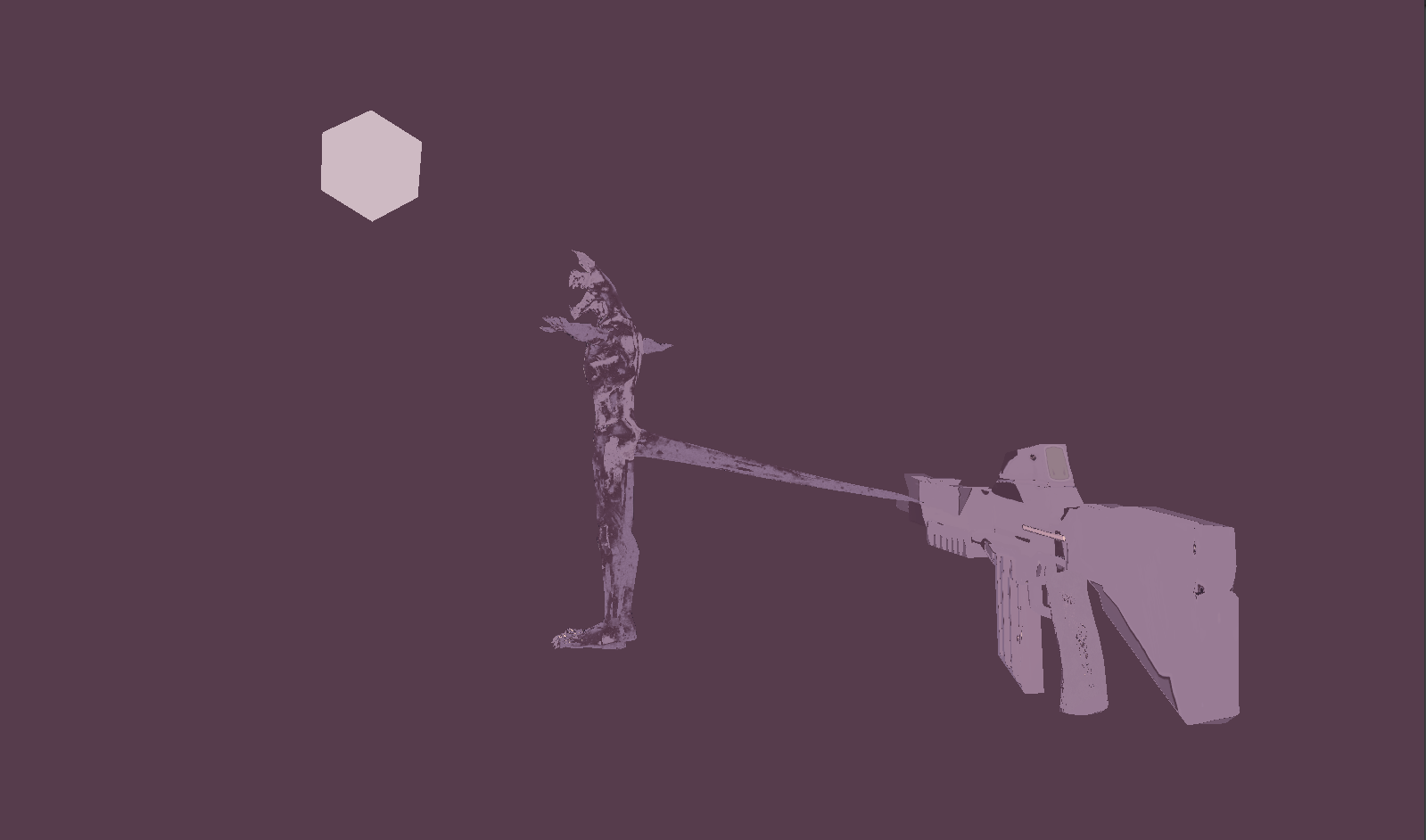
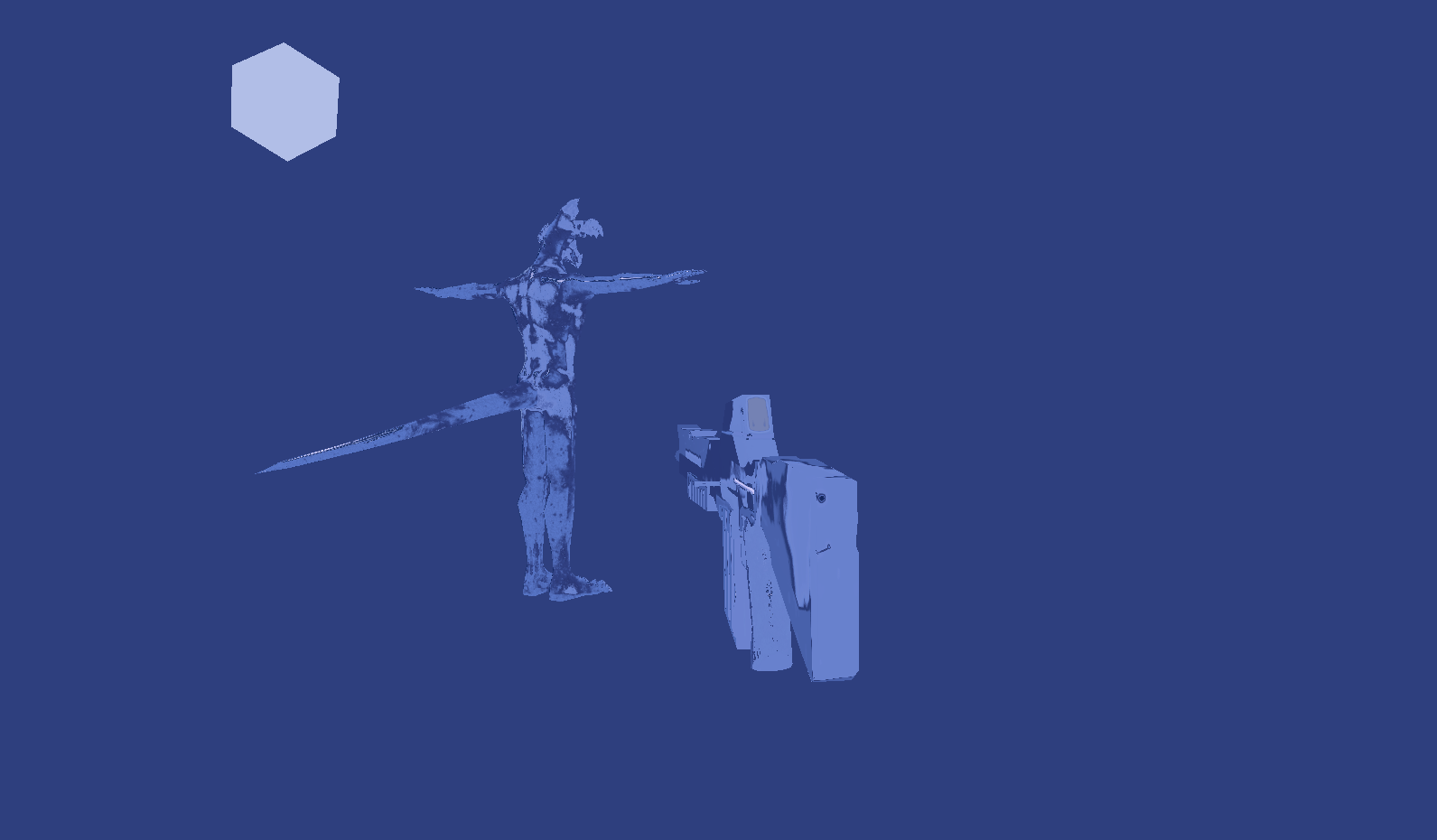
Screenshots of Modes

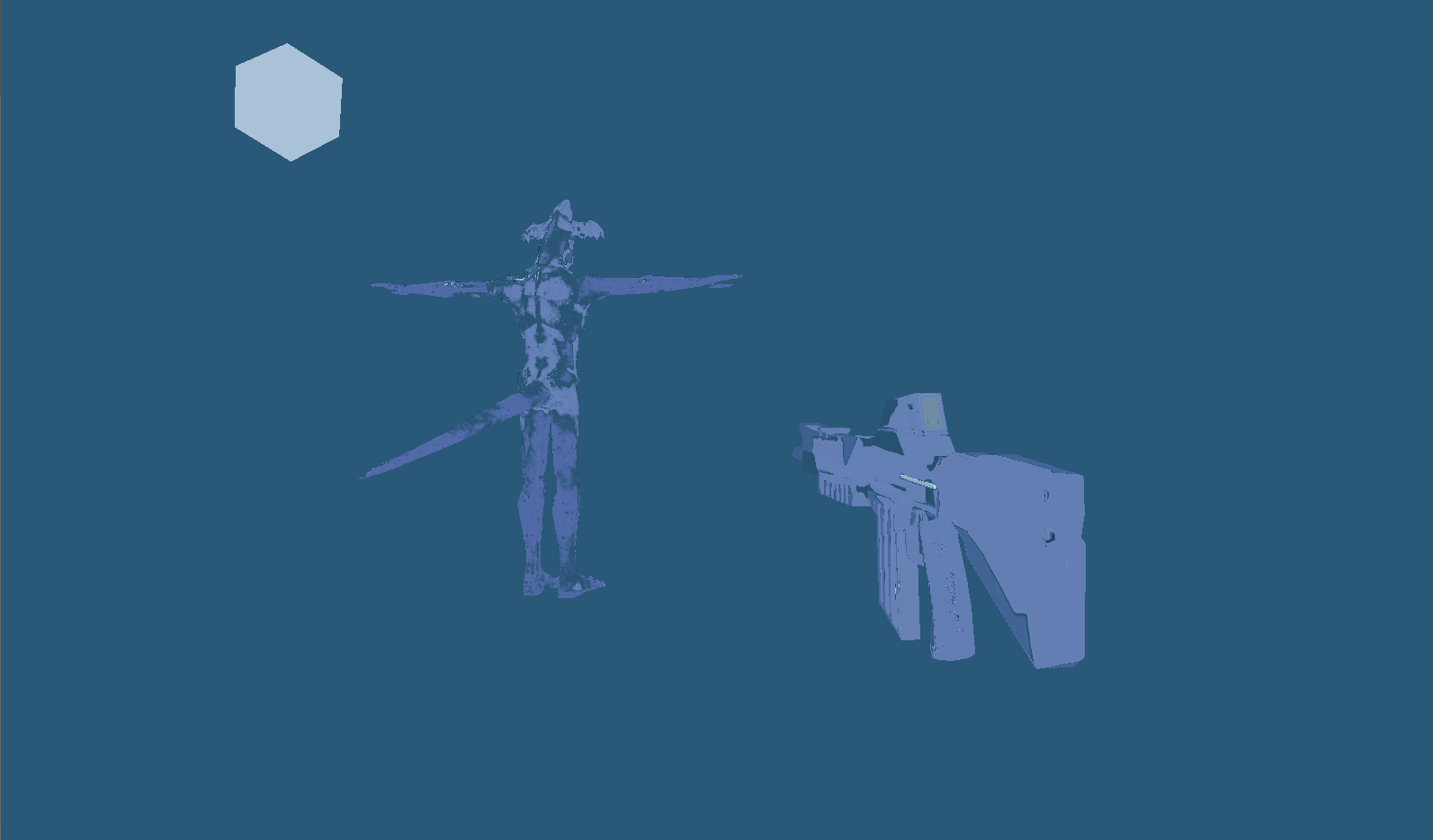
No lighting Warm Colour Grading Ambient Lighting with Warm Colour Grading

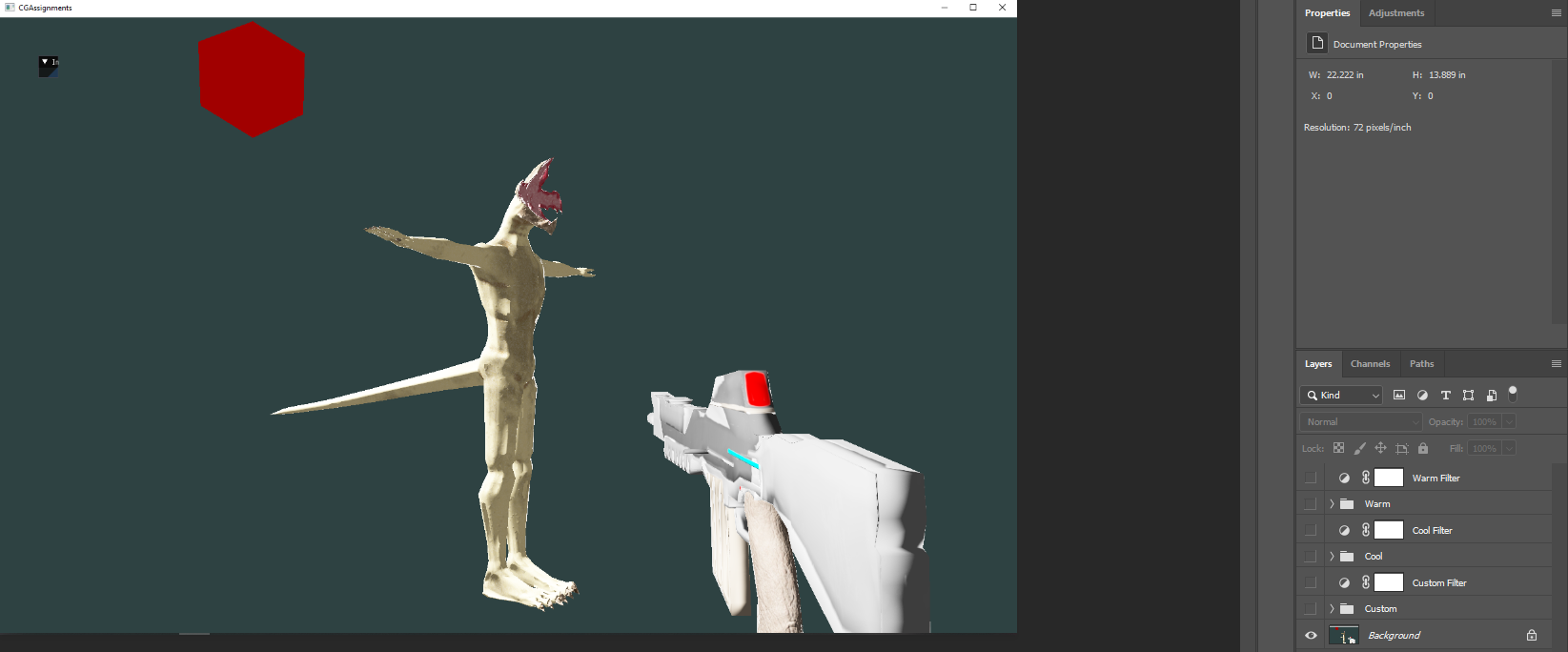
Specular Lighting with Warm Colour Grading Specular and Rim Lighting

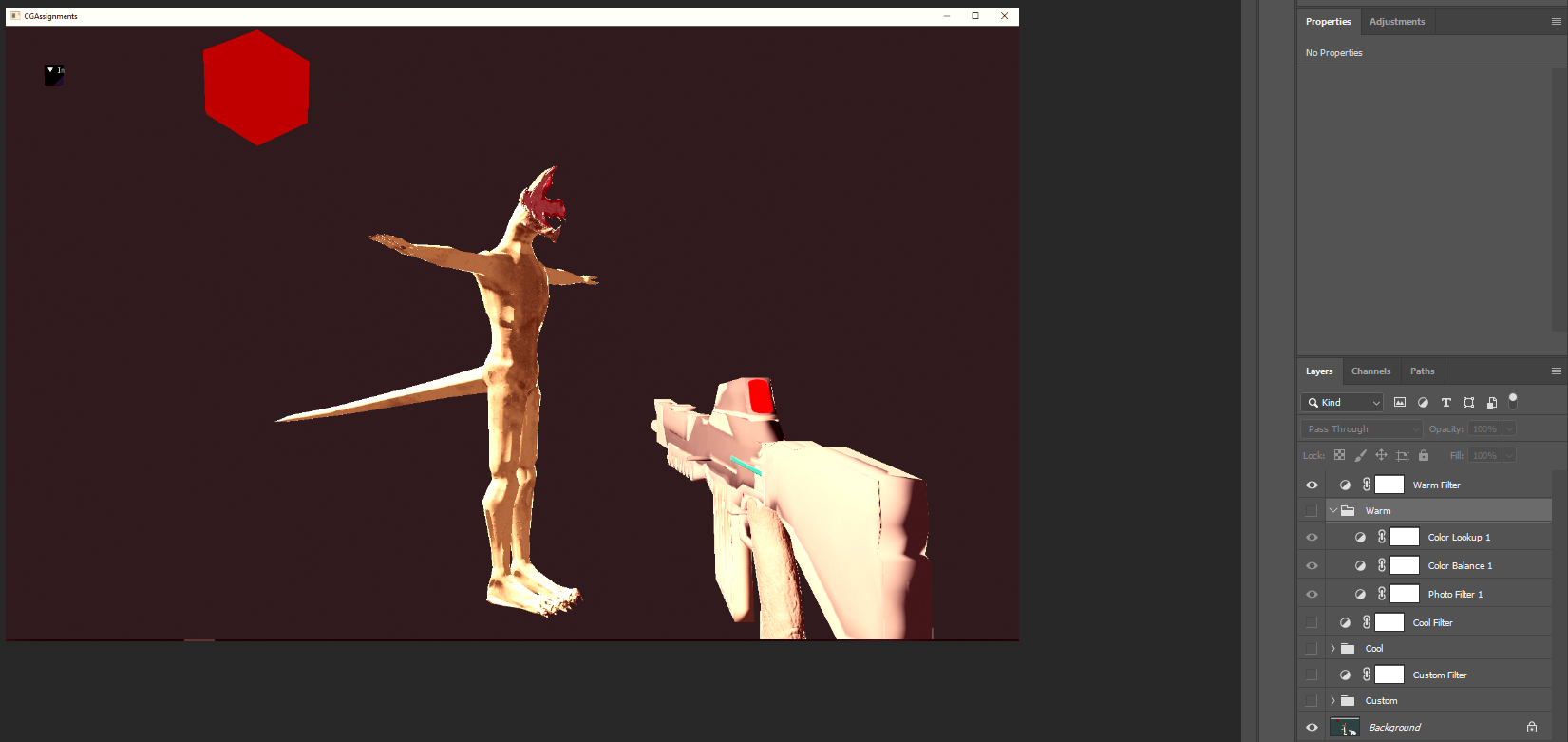
Specular, Rim, and Ambient Lighting Cool Colour Grading



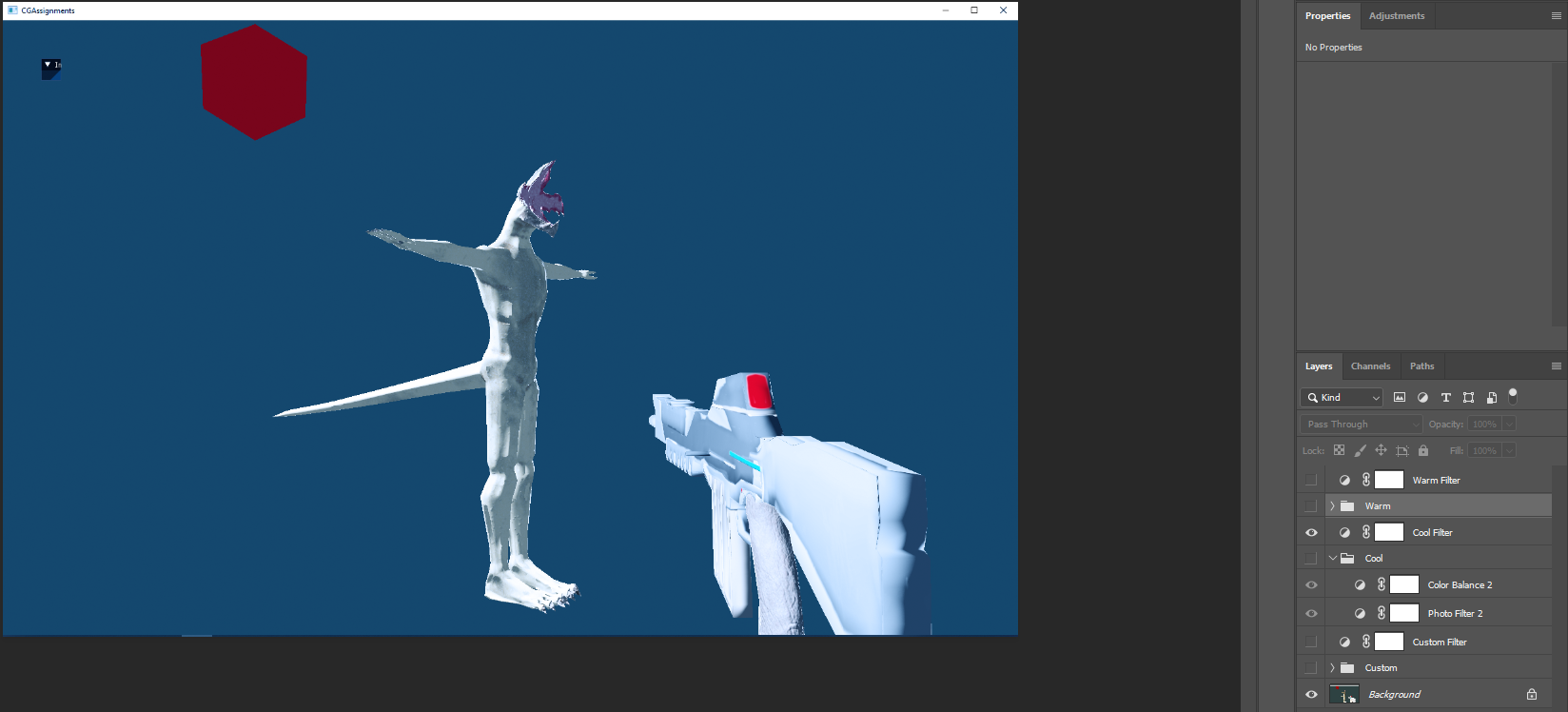
Custom Colour Grading



Base



Warm LUT



Cool LUT



Custom LUT