

Rendering Competition

by Sihan Chen



For details of the character (e.g., complex pattern on both clothes and crown), I first implemented the image texture and normal map to support more fine-grained meshes. Then for more realistic lighting, I calculated the TBN (tangent, bitangent, normal) maps for mesh, and then reconstructed the shading frame (also for spheres). Besides, I implemented Disney BSDF for all the listed 10 parameters to better control the complex materials. Some of the parameters even support image input instead of constant presetting values. To finetune each body parts' parameters, I used a very fine-grained game engine exported mesh, so that I can set parameters for different parts (e.g., brooch on the chest, diamond on the back of hand, jewelry on the crown, ruby on the stick), and detach them from the main parts where they are attaching (e.g., fabric clothes, metallic crown). Getting the character model is hard, as you can't find it anywhere from the assets market. So I must unpacking the game package to get the assets. Actually the rendered image should be more realistic due to the full preparation (game mesh, TBN map, full Disney bsdf), but the normal map is broken at some part during unpacking the game package somehow, leading artifacts in some places if observing closely. After getting the character mesh, it's also hard to change to given pose from the 'A' pose due to her long sleeves and skirts. Only using common skeleton binding will not

work, I have to repaint some weights. To make the clothes more realistic, I did some sculpting on the long sleeves as well. For better lighting and highlighting the character's details, I implemented colored spotlight to provide blue color.

Because I didn't implement the heterogeneous participating media and my homogeneous media can't support spatial varying parameters as well due to time restrictions of working alone, instead of only using homogeneous media to represent water, I attached a dielectric water surface on that. To make the water deeper, I also input a albedo into the dielectric bsdf. Also for the moon, without heterogeneous media, it's hard to simulate the fog on the moon. Another reason is I have to make the moon very big in the scene, and it's far away from the camera. So using homogeneous self emission media doesn't contribute a lot (the self emission is very weak). I chose to set lots of colored point lights around the moon and use a detailed image texture with tuned disney bsdf parameters that could simulate shiny white fog. The final homogeneous participating media using (with hg phase function) is the magic stick. Inside the hollow structure of it, there is a self emission media to simulate magic.

To get fancier visual effect about the magic, I also used a tricky alpha map. Essentially the magic visual effect is pattern on a mesh, then I reuse the transparent bsdf combining with alpha map to implement this magic visual effect.

To get an interesting background and continuous lighting, I implemented the environment map. Through some rotation, I set the galaxy to connect the stick magic visually. Adding some jellyfish on the water heading towards a direction also connects the stick magic visually, and cut the whole image together with the galaxy as a separating line.

Finally, because the scene is very complex (many lights around moon and complex characters), I can't lift up the spp too much. But I found the intel open image denoise works quite well, permitting me a more aggressive scene setting.