**Geometria**

Geometria é a área da matemática que desempenha o papel de representar formas em um plano espacial. Enquanto a geometria euclidiana tradicional concentra-se na representação de objetos bidimensionais, a geometria 3D ou Espacial refere-se a análise de figuras e objetos que ocupam o espaço tridimensional, ou seja, aqueles que têm comprimento, largura e altura, utilizando coordenadas tridimensionais (x, y, z) para descrever pontos no espaço. (Luiz Paulo)

O entendimento da geometria é essencial para a criação de malhas poligonais, que são uma das formas de modelagem gráfica. As malhas poligonais podem ser descritas como uma coleção de pontos, arestas e faces conectados em polígonos como triângulos e quadriláteros, que servem para simplicar o processo de renderização. Estes polígonos constituem a superfície da malha poligonal que é utilizada para modelar e representar objetos tridimensionais (Siim Tiigimägi, 2021). Qualquer tipo de objeto, como pontos, linhas, cubos, esferas, edíficios, terrenos e animais pode ser construído e renderizado computacionalmente através das malhas poligonais.

Além da malha poligonal, um objeto 3D também guarda informações sobre o seu material, constituído de atributos que são aplicados ao modelo para modificar como ele vai ser visto no seu estado final, deixando-o com uma aparência mais característica e dinâmica. Estes materiais são classificados em diferentes tipos, tais como cor, brilho, reflexo, transparência e refração. Cada um destes tipos de materiais adicionam ou modificam atributos básicos dos modelos como e por exemplo a transparência que permite que seja possível enxergar através dele, sendo útil para simular objetos como vidro, e também o reflexo que adiciona a capacidade de refletir a luz, útil para simular espelhos e objetos de metal. (Cleyder Felipe, 2022)

Durante a modelagem, os objetos 3d normalmente são criados com uma cor cinza padrão, e para que o modelo possua uma aparência mais detalhada é necessário a aplicação de uma ou mais textura ao modelo. Textura são normalmente imagens 2d aplicadas ao redor de modelos 3D para criar uma aparência mais realista e detalhada. Além de definir a cor primária de um objeto, as texturas podem definir suas outras propriedades de seu material, como seu brilho ou transparência. (Nazanin Shahbazi, 2023)

A iluminação é a parte da modelagem 3D que lida com a configuração de simulação de fontes de luz em cenas tridimensionais permitindo a visão mais clara de seus componentes. Essa é uma parte importante, pois tem grande impacto na renderização final da cena, e uma iluminação mal feita pode fazer com que os modelos não apresentem muito bem todos os seus detalhes. (Arash Naghdi, 2020)

Existem diversas formas de configurar formas de luz em uma cena de acordo com o objetivo que deseja alcançar com ela. Algumas destas formas de iluminação são a direcional, que gera uma fonte de luz que segue uma direção fixa e ilumina igualmente todos os objetos em seu caminho; a de ponto, que diferente da direcional gera uma fonte de luz em um ou mais pontos específicos que se expande em todas as direções e perdendo intensidade à medida que se distancia; a holofote, que gera uma fonte de luz em um ponto que se expande de forma cônica em uma direção com intensidade focado no centro; e a ambiente, que gera luz em uma área determinada que expande em todas as direções muito utilizada como luz padrão em cenas. (Tina Lee, 2023)

O funcionamento de luz no mundo real é extremamente complicado e depende de muitos fatores. Portanto, para ser capaz de simular um ambiente iluminado, um computador deve utilizar um modelo simplificado que possa ser renderizado em tempo real, ao mesmo tepmo que oferece resultados semelhantes aos encontrados no mundo real. Um destes modelos é o *Phong Lightning Model*, que baseia-se em certas propriedades dos materiais que determinam como a luz será refletida no ambiente: *Ambient*, cor base do objeto na ausência de luz; *Diffuse,* cor do objeto ao refletir uma luz e *Specular,* pequeno ponto brilhante de uma luz que aparece no objeto. (VRIES, 2020).

Outra forma de simular a luz é pelo método PBR (p*hysically based rendering),* ou renderização baseada em física, uma técnica de simular iluminação similar ao da vida real. O PBR foca em calcular a interação da luz nas superfícies e textura dos modelos de forma fisicamente realista, por exemplo como a luz é refletida, absorvida e refratada pelo material. É uma técnica amplamente utilizada por ser mais prática e eficiente do que seus antecessores, além de ser compatível com diferentes sistemas de iluminação e prover um resultado final mais realista. (Jeff Russell)

Pelo PBR, os materiais se tornam muito mais configuráveis, fornecendo atributos como *Metalness (*reflexão do ambiente); *roughness (*aspereza); *refraction (*refração); *environment (*reflexão do plano de fundoo*); normal (*elevação e profundidade, sem afetar a geometria); *displacement* (elevação maior, afetando a geometria); *opacity (*transparência); *sheen (*para tecidos ou roupas); *transmission (*transparência com reflexão); *clearcoat (*reflexão limpa em material aspero*); paralax (*acentuação de relevo*); decal (*detalhes independentes adicionais*); ambient occlusion (*sombras suaves nos detalhes da superfície*); thickness (*espessura utilizada na refração*); iridescence (*mudança na faixa de cores pelo ângulo do observador*) e anisotropic (*afeta a reflexão a depender do ângulo do observador*).* (A23D, c2023; Babylon.js, c2023; Three.js, 2023; PluralSight, 2022).

Além da iluminação, uma cena precisa de sombras para dar maior realismo. As sombras são resultados da ausência de luz devido à oclusão de algum objeto, e auxiliam o observador a visualizar a relação espacial entre os objetos, dando um melhor senso de profundidade a uma cena Geralmente o processo de cálculo de sombras é feito em tempo real através de diversas técnicas, como o uso de *shadow maps*, que renderiza a cena várias vezes a partir do ponto de vista das fontes de luzes presentes (VRIES, 2020). Também é possível pré-computar os *lightmaps*, texturas que definem as partes do objetos iluminadas pelas fontes de luzes da cena, para os objetos e fontes de luzes imóveis e assim não precisar recomputar seus valores a cada quadro, obtendo grande ganho de desempenho. (PLAYCANVAS, 2023).

Outro efeito que pode adicionar realismo a cenas é o uso de iluminação global, no qual é utilizado a informação de vários objetos espalhados pela cena para calcular a reflexão, transparência e sombras. Um destes algoritmos é o *Ray Tracing,*, que utiliza os raios de luz através da cena para calcular o seu transporte pelos diversos objetos espalhados pela cena. Este método produz resultados muitos mais realistas que a rasterização direta, que renderiza a cena por um ponto de vista apenas. O *Ray Tracing* consegue produzir apenas um número limitado de efeitos, como reflexões agudas e sombras. Uma técnica ainda mais avançada é o *Path Tracing,* que cálcula caminhos pelos quais a luz viaja e se reflete pelos objetos até o observador, sendo capaz de produzir um resultado fisicamente acurado e realista. (CAULFIELD, 2022; MÖLLER et al, 2018)

Ainda, no final da renderização, é possível adicionar efeitos de pós-processamentos e assim melhorar significamente a aparência da imagem gerada na tela. Alguns dos efeitos mais comuns são: *fog* (neblina ou névoa no plano de fundo); *anti*-*aliasing* (remover os serrilhados das bordas); *color* *curves* (controle de saturação, *hue*, sépia, contraste, luminosidade); *depth* *of* *field* (desfoque da lente, criando um efeito cinematográfico); *chromatic* *aberration* (dispersão de cores); *bloom* (brilho das áreas mais luminosas); *motion* *blur* (borramento em objetos em movimento); *tonemapping* (alteração de cores para HDR); *vignette* (sombreamento nas bordas da tela); SSAO (*Screen Space Ambient Occlusion*, uma forma de calcular o *ambient occlusion* somente com dados da tela, salvando desempenho); SSR (*Screen Space Reflection,* reflexões mais sutis que podem ser usados em poças de água) e lens distortion (distorção da imagem pelo formato da lente). (BABYLON.JS, 2023; UNITY, 2022).

Para trazer os objetos a vida, um sistema de animação é necessário. Um método comum de simular os movimentos de um objeto é pelo sistema de animação esqueletal, que adiciona ao modelo 3D do objeto uma estrutura esquelética constituída de múltiplas articulações, sendo um deles o nó raíz, no qual é organizado a hierarquia constituinte da estrutura esquelética. Cada articulação guarda uma lista de polígonos conectados diretamente a ele, movendo-os ao ter sua posição ou rotação alterada pela animação, além de propagar o movimento para as articulações no nível menor de sua hierarquia. Uma animação é construída através *keyframes*, quadros chaves definidos pelo modelador, que são interpolados pelo programa a fim de criar uma animação fluída (VRIES, 2020)

Animação esqueletal é uma técnica de animação que utiliza uma estrutura de hierarquia de partes conectadas para adicionar mais controle a objetos e simular movimentos de forma mais realista. Ela funciona colocando em um objeto uma série de partes interconectadas invisíveis na renderização final, que simula um esqueleto, podendo ser conectado a partes específicas do objeto ou a ele em um todo. Essas partes podem ser utilizadas para mover o objeto. Partes relacionadas também serão movidas de forma mais suave, criando uma movimentação mais fluida e agradável. (Adobe 2023)

**Materiais**

Materiais são atributos que são aplicados ao modelo para modificar como ele vai ser visto no seu estado final, deixando-o com uma aparência mais característica e dinâmica. Estes materiais são classificados em diferentes tipos, sendo estes cor, brilho, reflexo, transparência e refração. Cada um destes tipos de materiais adicionam ou modificam atributos básicos dos modelos como e por exemplo a transparência que permite que seja possível enxergar através dele, sendo útil para simular objetos como vidro, e também o reflexo que adiciona a capacidade de refletir a luz, útil para simular espelhos e objetos de metal. (Cleyder Felipe, 2022)

**Textura**

Durante a modelagem, os objetos 3d normalmente são criados com uma cor cinza padrão, e para que o modelo possua uma aparência mais detalhada é necessário a aplicação de uma ou mais textura ao modelo. Textura são normalmente imagens 2d aplicadas ao redor de modelos 3D para criar uma aparência mais realista e detalhada, não só colocando uma imagem por cima do modelo, mas também modificando aspectos do material do modelo como transparência e brilho. (Nazanin Shahbazi, 2023)

Há também as texture maps que são conjuntos de configurações podem ser divididas em diferentes tipos que modificam aspectos diferentes da textura e material dos modelos que podem ser usados juntos para chegar em resultados diferentes. Alguns destes tipos são diffuse, que armazena e modifica as cores básicas da textura, opacity, que altera a intensidade da transparência da textura, specular, que controla o quanto a textura é reflexiva, bump, que é usado para criar a sensação de profundidade, e glossiness, que controla como a luz será espalhada pela textura. (Balpreet Singh, 2022)

**Iluminação**

**Physically based rendering**

Physically based rendering (PBR) é uma técnica de simular iluminação que se baseia em modelos de iluminação da vida real. O PBR foca em calcular a interação da luz nas superfícies e textura dos modelos de forma fisicamente realista, por exemplo como a luz é refletida, absorvida e refratada pelo material. É uma técnica amplamente utilizada por ser mais prática e eficiente do que seus antecessores, além de ser compatível com diferentes sistemas de iluminação e prover um resultado final mais realista. (Jeff Russell)

**NPM**

NPM ou Node Package Manager é uma biblioteca e repositório de código aberto para pacotes de JavaScript softwares. Ele é uma ferramenta muito útil do Nodes.js que com funções simples permite aos desenvolvedores instalar, desinstalar, atualizar e gerenciar dependências em uma aplicação de forma prática e fácil. (Nicole Abramowski, 2022)

**ES modules**

ES modules é uma das grandes adições ao javascript moderno, tornando mais simples e rápido a importação e exportação de funcionalidades entres arquivos de javascript. O ES modules é uma forma de padronizar a forma de importação e exportação de funcionalidades utilizando de uma sintaxe nativa, sem a necessidade de utilizar bibliotecas adicionais. Baseando-se nas palavras-chaves export e import, esta sintaxe permite que os desenvolvedores definem de forma mais clara quais funcionalidades específicas estão sendo importadas e exportadas entre arquivos. (Gil Tayar, 2021)

**Skeletal animation**

Skeletal animation é uma técnica de animação que utiliza uma estrutura de hierarquia de partes conectadas para adicionar mais controle a objetos e simular movimentos de forma mais realista. Ela funciona colocando em um objeto uma série de partes interconectadas invisíveis na renderização final, que simula um esqueleto, podendo ser conectado a partes específicas do objeto ou a ele em um todo. Essas partes podem ser utilizadas para mover o objeto, partes relacionadas também serão movidas de forma mais suave, criando uma movimentação mais fluida e agradável. (Adobe 2023)

**Skybox**

Skybox é uma técnica de criação de cenas 3D com o intuito de melhorar sua ambientação e diminuir o custo de renderização. Ela consiste em criar uma caixa texturizada ao redor da cena para criar um plano de fundo do ambiente à cena sem ter que modelar e renderizar grandes áreas, melhorando o desempenho e velocidade de renderização da cena. (Yana Krasnolutska, 2023)

**Câmera**

Câmera é uma funcionalidade essencial para o desenvolvimento de cenas 3D, consistindo em um objeto invisível que permite a renderização e visualização da cena. As câmeras podem ser divididas em dois principais tipos, sendo eles perspective e orthographic. A câmera perspective reflete a forma que vemos na vida real, nela os objetos mais distantes parecem menores e objetos próximos parecem maiores, tendo sua área de visão controlada por uma variável chamada FOV (Field of View), que define quantidade da cena que é visível pela câmera. A câmera orthographic independentemente da distância os objetos permanecem em seu tamanho real, permitindo um melhor alinhamento de objetos e visualização para comparação de tamanhos. (Blender Manual)

**POST-PROCESSING and full-screen effects.** Unity User Manual, 2022. Disponível em: <https://docs.unity3d.com/Manual/PostProcessingOverview.html>. Acesso em 16 out. 2023.

MARTINDALE, Jon. **Ray tracing vs. path tracing — which is the best dynamic lighting technique?.** DigitalTrends, 2022. Disponível em: [https://www.digitaltrends.com/computing/ray-tracing-vs-path-tracing/#:~:text=Path%20tracing%20is%20also%20considered,rays%20of%20light%20would%20take](https://www.digitaltrends.com/computing/ray-tracing-vs-path-tracing/" \l ":~:text=Path tracing is also considered,rays of light would take).. Acesso em 16 out. 2023.

-- CAULFIELD, Brian. **O que é Path Tracing?.** Blog NVIDIA, 2022. Disponívem em: <https://blog.nvidia.com.br/2022/05/10/o-que-e-path-tracing/>. Acesso em 16 out. 2023.

**WHAT are Decals?.** A23D, c2023. Disponível em: <https://www.a23d.co/blog/what-are-decals/>. Acesso em 15 out. 2023.

**UNDERSTANDING Ambient Occlusion.** PluralSight, 2022. Disponível em: <https://www.pluralsight.com/blog/film-games/understanding-ambient-occlusion>. Acesso em 15 out. 2023.

Abramowski, Nicole. **“What is NPM? A Beginner's Guide”**. CareerFoundry, 2022; Disponível em: <https://careerfoundry.com/en/blog/web-development/what-is-npm/>. Acesso em 5 de outubro de 2023

Krasnolutska, Yana. **“Unlocking the Skies: Exploring the Immersive Potential of Unity 3D Skyboxes”.** MARKETSPLASH, c2023. Disponível em: <https://marketsplash.com/tutorials/unity-3d/unity-3d-skyboxes/>. Acesso em 7 de outubro de 2023.

Lee, Tina. **“Lights and Shadows: CG Lighting Types for 3D Animation”**. Academy Of Animated Art, c2023. Disponível em: <https://academyofanimatedart.com/lights-and-shadows-cg-lighting-types-for-3d-animation/> Acesso em 27 de setembro de 2023.

Naghdi, Arash. **“The ultimate guide to lighting fundamentals for 3D”**. Dream Farm Studios, c2020. Disponível em: [https://dreamfarmstudios.com/blog/the-ultimate-guide-to-lighting-fundamentals-for-3d/#:~:text=What%20is%203D%20lighting%3F,at%20different%20levels%20of%20complexity](https://dreamfarmstudios.com/blog/the-ultimate-guide-to-lighting-fundamentals-for-3d/" \l ":~:text=What is 3D lighting%3F,at different levels of complexity). Acesso em 27 de setembro de 2023.

Russell, Jeff. **“Basic Theory of Physically-Based Rendering”.** Marmoset LLC, 2023 Disponível em: <https://marmoset.co/posts/basic-theory-of-physically-based-rendering/>. Asesso em 02 de outubro de 2023.

Santos, Cleyder Felipe de Araujo. **“O que são materiais e texturas em um software 3D?”**. Alura, c2022. Disponível em: <https://www.alura.com.br/artigos/o-que-sao-materiais-texturas-software-3d>. Acesso em 26 de setembro de 2023.

Shahbazi, Nazanin. **“Exploring the World of 3D Textures: A Comprehensive Guide”**. Pixtune, c2023. Disponível em: <https://pixune.com/blog/3d-texturing/>. Acesso em 26 de setembro de 2023.

SILVA, Luiz Paulo Moreira. **"O que é geometria?"**; Brasil Escola. Disponível em: <https://brasilescola.uol.com.br/o-que-e/matematica/o-que-e-geometria.htm>. Acesso em 24 de setembro de 2023.

Singh, Balpreet. **“How to make photorealistic 3D graphics with different texture maps?”**. Webdew, c2022. Disponível em: <https://www.webdew.com/blog/how-to-make-photorealistic-3d-graphics>. Acesso em 27 de setembro de 2023.

Tayar, Gil. **“Using ES Modules (ESM) in Node.js: A Practical Guide (Part 1)”.** 2021. Disponível em: <https://gils-blog.tayar.org/posts/using-jsm-esm-in-nodejs-a-practical-guide-part-1/>. Acesso em 5 de outubro de 2023.

Tiigimägi, Siim. **“O que é uma malha Polygon e como editá-la?”.** 3D Studio, 2014 - 2023. Disponível em: [https://3dstudio.co/pt/polygon-mesh/#introduction](https://3dstudio.co/pt/polygon-mesh/" \l "introduction). Acesso em 26 de setembro de 2023.

**“Rigging and skeletal animation: what it is and how it works.”.** Adobe, c2023. Disponível em: <https://www.adobe.com/uk/creativecloud/animation/discover/rigging.html>. Acesso em 7 de outubro de 2023.

**“Cameras”.** Blender. Disponível em: <https://docs.blender.org/manual/en/3.6/render/cameras.html>. Acesso em 10 de outubro de 2023.