Raytracing
Version 1: Nov 6, 2017
Version 2: Nov 13, 2017

Basics, algorithms, specific examples of implementation, history, advantages/disadvantages, show self-made references images/animations using raytracing and comparing with & w/o,

History:

- Descartes' "Dioptrics" and "Optics" in *The Cambridge Descartes Lexicon* by Jeffrey McDonough
    - Descartes introduces the idea of plotting light rays and calculating their interactions with various surfaces in the early 17th century in one of his earliest works, *Dioptrics.* The paper also discusses reflection and refraction. Essentially, the ideas that Descartes presents in this paper serve as the foundation of modern ray casting.
- *Some techniques for shading machine renderings of solids* by Arthur Appel (1968)
    - This paper presents the first ray casting algorithm used for rendering objects in a scene.
- An Improved Illumination Model for Shaded Display by Turner Whitted (1980)
    - This paper introduces a recursive ray tracing algorithm. When a ray intersects an object, it may generate up to three more rays which are used to simulate reflection, refraction, and shadows. This algorithm is more expensive than its predecessors, but it results in very realistic images.
- *Distributed Ray Tracing* by Robert L. Cook, Thomas Porter, Loren Carpenter (1984)
    - This paper discusses the technique of Distributed Ray Tracing, sometimes referred to Stochastic Ray Tracing. The idea of Distributed Ray Tracing is that instead of using a single calculation, multiple calculations are averaged together to produce the final value. This process is called over-sampling, and the effect is that there are less artifacts and smoother shading. This effect is used in a variety of different applications and commonly referred to as anti-aliasing.
- *Backward Ray Tracing* by James Arvo (1986)
    - This paper discusses backward ray tracing, a technique introduced in order to better simulate diffuse reflection of indirect light. This technique differs from traditional ray casting in that not only are rays emitted from the eye but also from the light sources present in the scene. This technique is much more efficient than forward ray tracing because, when the rays that are emitted from the eye are subsequently reflected or refracted, they rarely ever hit the light sources, revealing nothing about how the light hits that point of reflection or refraction. Therefore, a small proportion of the rays reach the destination, resulting in wasted computations in an already expensive process.
- *High Quality Rendering using Ray Tracing and Photon Mapping* by Henrik Wann Jensen, Per Christensen (2007)
    - [...]

Raytracing in Film

-Claude Kalache of Pixar made a controversial decision when he wanted to change how light worked at Pixar because it meant many of the artists had to relearn how to create the animation scenes.

-The change had to do with raytracing. They wanted the effect it produced but it was too costly due to needing to map out millions of beams of light.

-They wanted to add the ray tracing because it created a more accurate environment in terms of reflection, refraction, colored shadows, etc.

-They used a new Global Illumination system, which automatically placed reflections and shadows based on where the lights of the scene were located. This transition meant that the artists had less to do but heavily increased the work that their renderers had to do. “Global illumination = ‘super hard core ray tracing’.”

-Pixar didn’t use raytracing until the movie “Cars” but even then it wasn’t a full implementation, due to the computational expense. They only used it for specific scenes and main characters.

Raytracing in Autodesk Maya

-Algorithm it uses is: for each pixel in the image, set the ray, trace the ray, calculate illumination.

-When generating rays, it traces a ray for each pixel in the image plane

-It simulates light rays from the light source to the eye

-In render settings under raytracing user can modify:

    -Reflections: max number of times a light ray can be reflected

    -Refractions: max number of times a light ray can be refracted

    -Shadows: “ “ “ “ “ reflected and, or refracted and still cause an object to cast a shadow

    -Bias: Corrects issue of dark areas or incorrect shadows on 3D motion blurred objects

-Raytracing built into Maya Software renderer so it is a simple implementation

Forward vs backward ray tracing

-Forward raytracing entails following rays from a camera towards objects, and then assessing their distances from various light sources and other objects.

-Backwards raytracing however, starts at the light source and heads towards the objects. The advantage of backwards being that the rays are more efficient; most rays coming off the light source are relevant to rendering, whereas the rays from forward tracing are rather inefficient.

-Forward tracing does however do a better job of determining object colors.

Technical Breakdown slide

-Create ray from eye to each pixel on the scene, and find the first object blocking the ray.

-Determine the shading of the object(pixel) based on material properites and lighting.

-Assumed that if it faces light it will be reached by the light. I.e. the light isnt blocked by another object or in shadow

-Most light rays from a light source don’t even enter the viewers eye, so they calculate rays from objects to the light source, instead of from the light source to the objects.