

Abstract

Ray Tracing is one of the most important rendering techniques used in computer graphics. Ray traced images are more accurate and photo-realistic as compared to direct rendering. Ray Tracing was earlier considered impractical for rendering scenes at interactive rates because of its high computational cost. However, with the advancements in modern Graphics Processing Units (GPU) and CPUs, ray tracing at interactive rates has now become possible.

Parametric patches have been widely used in many fields to describe a model accurately. They provide a compact and effective way of representing an object and also possess the ability to remain curved on zooming. Ray Tracing of parametric surfaces was considered to be a static process because of the high complexity of intersection algorithms. With advancements in ray tracing techniques and high compute power devices, recent works on ray tracing parametric surfaces have reported near interactive results.

We present a scheme for interactive ray tracing of Bezier bicubic patches using Newton iteration in this dissertation. We use a mixed hierarchy representation as the acceleration structure. This has a bounding volume hierarchy above the patches and a fixed depth subpatch tree below it. This helps reduce the number of ray-patch intersections that needs to be evaluated and provides good initialization for the iterative step, keeping the memory requirements low. We use Newton iteration on the generated list of ray patch intersections in parallel. Our method can exploit the cores of the CPU and the GPU with OpenMP on the CPU and CUDA on the GPU by sharing work between them according to their relative speeds. A data parallel framework is used throughout starting with a list of rays, which is transformed to a list of ray-patch intersections by traversal and then to intersections and a list of secondary rays by root finding. We are able to significantly outperform multi-core CPU implementation and previous GPU implementation using the mixed hierarchy model.

Shadow and reflection rays can be handled exactly in the same manner as a result. The secondary ray list is again sent to the starting of the algorithm to perform mixed hierarchy traversal and intersection tests. We perform fixed depth multiple bounce ray tracing. We also show how our method extends easily to generate soft shadows using area light sources. These effects provide higher realism to the ray traced images.

We render a million pixel image of the Teapot model at 125 fps on a system with an Intel i7 920 and a Nvidia GTX580 for primary rays only and at about 65 fps with one pass of shadow and reflection rays. We are able to ray trace bigguy in a box scene with multi-bounce at near interactive rates. We

get a speed up of about 5-30x for our hybrid Newton's method implementation over our optimized CPU implementation and about 20-50x over previous GPU implementation of Kajiya's method to ray trace Bezier surfaces. Traversing the mixed hierarchy is the most time consuming step of the algorithm. We expect to see better performance with greater cache size. The hybrid model would be optimal for systems with equal compute power of CPU and GPU. The proposed model is suitable for parallel architecture, hybrid systems and multi-GPU systems.

Global illumination effects have recently started gaining popularity with the progress in parallel architecture. We extend our algorithm to global illumination effects to demonstrate its capabilities. Global illumination effects have not been reported for parametric surfaces. We perform path tracing by tracing a large number of rays per pixel for a fixed depth. Number of samples greatly increase the quality of the image generated. We also perform more advanced effects like ambient occlusion, depth of field, motion blur and glossy surface. We are able to path trace a 512×512 image with 1000 samples per pixel in about 165 seconds. We report timings for other advanced effects. We find that ray coherence is essential for optimal performance when ray tracing Bezier surfaces on the GPU. Size of the dataset also plays a small part in the overall rendering times. The work done in this dissertation should serve as the starting point to optimally render Bezier surfaces with advanced global illumination techniques.