3Delight Technical Specifications

Key technical features of 3Delight 7.0

Table of Contents

1 Rendering

RenderMan compliance

> 3DELIGHT fully adheres to the RenderMan standard. Additionally, 3DELIGHT offers almost all the extensions introduced by other RenderMan-compliant renderers and adds a wealth of unique features.

REYES Rendering Algorithm

> The core rendering algorithm in 3DELIGHT is REYES based. This algorithm has proven its worth during more than two decades of outstanding quality pictures. Images produced using this technique have a *feel* that is difficult to produce using ray traced based solutions[1].

Multi-threading and Multi-processing

> Renders can be started in both multi-threading and multi-processing. Additionally, processes can be started on remote machines without additional task scheduling software.

Anti-aliasing

> *Screen Space*
>
> > High quality, efficient, screen space filtering can be performed using a variety of filters. The filtering window can be extended to any width (multi-pixel filtering) with negligible effect on performance.
>
> *Texture Space*
>
> > Efficient anisotropic texture filtering delivers impeccable quality anti-aliasing for all texture and shadow map lookups.

Realistic Camera Shutter

> *Motion Blur*
>
> > 3DELIGHT simulates motion blur using an algorithm that mimics the shutter of a real camera. **Multi-segment** motion blur and **deformation** blur are supported on *all geometric primitives*.
>
> *Depth of Field*
>
> > Efficient and high-quality depth of field is achieved by simulating the bokeh of a real camera.

Sub-pixel Precise Displacements

> Contrary to other rendering packages, displacements in 3DELIGHT are almost a *no cost* operation.

On-demand Ray Tracing

> Ray tracing is triggered when reflections or refractions are needed. 3DELIGHT implements a state of the art ray tracer that correctly renders smooth surfaces and provides **correct ray derivatives** to shaders. 3DELIGHT doesn't rely on surface tessellation in its ray tracer: surfaces are used *as is*, saving memory on subdivision surfaces and parametric surfaces.

---

[1] Note that 3DELIGHT can, if asked, use its ray tracer as a base renderer.

Global Illumination

> Photon mapping, final gathering and HDRI lighting are all supported. High dynamic range input and output is supported through floating point TIFF, cineon or OpenEXR files.

Arbitrary Output Variables

> The renderer is able to output, in a single rendering pass, as many different variables as desired. Common output variables are color, depth, normals, ...etc.

Lighting and Shadows

> Area lights, soft shadows, ray traced shadows, shadow maps and "deep" shadow maps are supported

Subsurface Scattering

> 3DELIGHT has automatic subsurface scattering features, freeing you from complicated setups and multi-pass renderings.

Level of Detail

> Geometry can be specified at many detail levels (with appropriate detail blending curves) for efficient rendering of highly complex scenes.

Toon Shading and Outlining

> 3DELIGHT has advanced features for inking and outlining. The renderer is able to outline *any* output variable[2] and produces outlines of very high quality.

Complete File Format Support

> 3DELIGHT has support for most commong image file formats:

> | | |
> |---|---|
> | *Input* | TIFF, OpenEXR, JPEG, PSD (*PhotoShop* files), IFF, PIC, SGI, PIC (*SoftImage* files), GIF and Radiance. |
> | *Output* | TIFF, PSD, OpenEXR, Cineon, IFF, PNG, Radiance and '.tdl' texture files. |

---

[2] Other renderers mainly operate on depth and normal variations.

2 Programmable Shading

RenderMan Shaders

RenderMan shaders are fully supported (surface, displacement, light, atmosphere, interior and imager). All data types and all the standard shadeops are supported. Light categories, message passing (both *forward* and *backward*) as well as output variables are supported.

Optimizing Compiler

3DELIGHT implements an *optimizing* compiler that has almost all the code optimization functionalities found in a traditional C++ compiler.

SIMD Shader Execution

3DELIGHT uses a *Single Instruction Multiple Data* shading paradigm that enables very efficient shading. The basic idea behind SIMD shading is to execute each SL instruction on a *set* of shading samples instead of just one sample. This has the beneficial effect of *amortizing* shader evaluation cost as well as better CPU cache usage.

Machine Code Shaders

The shader compiler can produce, if asked, shaders that are compiled using the C++ compiler for even better performance. Such compiled shaders still preserve their SIMD nature.

DSO Shadeops

The collection of functions accessible to shaders can be extended using C++ written libraries.

Surface Derivatives

Surface derivatives for all shader variables are supported. This is an important quality feature (for correct anti-aliasing) that is not common in other rendering packages (especially in ray tracers).

Primitive Variables

Variables attached to geometric primitives are correctly interpolated and passed to shaders.

3 Geometry

Polygons   All polygonal geometries, with arbitrary numbers of vertices and faces, are supported (convex, concave, with holes).

Subdivision Surfaces

3DELIGHT renders Catmull-clark subdivision surfaces with sub-pixel accuracy, providing smooth, high-quality, renders. **Creases** and **darts** are supported.

Hierarchical Subdivision Surfaces

Hierarchical Catmull-clark subdivision surfaces (with hierarchical edits) are fully supported.

Parametric Surfaces

Bézier, B-spline, Catmull-rom, Hermite, bilinear patches and NURBS are supported.

Curves   3DELIGHT renders extremely large numbers of thin curves very efficiently, making it a good choice for fur and hair rendering. Similar to parametric surfaces, curves can be rendered in any basis and can also be specified as NURB curves.

Particles   Efficient particle rendering is an important feature in high-end rendering. To that end, 3DELIGHT provides *lightweight* particle rendering to efficiently render millions of particles. Additionally, particles can be rendered as disks, spheres, blobbies or bilinear patches (for sprite rendering).

Blobbies (Implicit Surfaces)

Blobbies are efficiently rendered using a hierarchical space subdivision algorithm. As with all other smooth surfaces, no tessellation artifacts are ever visible.

Trimming   NURBS trimming is supported. Trim curves are specified as NURBS and are rendered to sub-pixel accuracy.

Instance and Procedural Geometry

Object instancing as well as *on demand loading* are supported (using a multiplicity of methods).

Arbitrary Vertex Variables

Attaching variables to geometry, to later access them in shaders (see Chapter 2 [Programmable Shading], page 4), is supported. 3DELIGHT also offers very high quality subdivision schemes for variables attached to subdivision surfaces, removing distortion artifacts associated with linear interpolation schemes and the like (especially visible on texture coordinates).

4 Advanced Pipeline Features

Network Cache

3DELIGHT offers an extension for more efficient rendering in networked environ-
ments: a local file cache for textures and archives to minimize network traffic
and file server load. This proves particularly powerful when using a large quan-
tity of rendering servers.

The working principle of 3DELIGHT network cache is to copy files to a local file
system and reuse them when needed. If either a texture or a RIB archive is
"cacheable" and the file cache is full, one or more files are removed from the
cache to make space for the new one; an LRU (Least Recently Used) strategy
is used to choose which texture(s) or archive(s) to remove from the cache.

No user interaction is necessary to maintain the cache: 3DELIGHT *keeps all files
in sync at all times.*

Rx Library API

3DELIGHT's Rx library provides access to important shadeops used in the ren-
dering core. This includes texturing and noise functions.

Ri Plug-in Filters API

Ri plug-in filters are an important extension to the RenderMan interface, pro-
viding flexibility unknown to non RenderMan systems and pipelines. In short,
a *Ri plug-in* is a component lying between the application that generates Ri
calls and the rendering core. Being in such a strategic place, the plug-in can
alter or filter, in any desirable way, all Ri commands received by the renderer.

Baking   Both 3D and 2D baking are supported using easy to use shadeops.

5 Integration

Maya 3DELIGHT comes with *3Delight For Maya*, a full featured plug-in that uses 3DELIGHT as a renderer.

Massive[1] 3DELIGHT is supported by *Massive* "out of the box".

Houdini[2] 3DELIGHT is supported by *Houdini* "out of the box".

Mental Ray[3]
     3DELIGHT can read and render all the geometry contained in '`.mi`' files.

---

[1] *Massive* is a trademark of Massive Software.
[2] *Houdini* is a trademark of Side Effects Software.
[3] *Mental Ray* is a trademark of Mental Images.

Concept Index