3D Renderer of Implicit Surfaces

Tristano Munini

April 6, 2021

Index

Signed Distance Functions : Distance Surfaces

From *"Sphere tracing: a geometric method for antialiased ray tracing of implicit surfaces"* an implicit surface is defined by a function that, given a point in space, indicates whether the point is inside, on or outside the surface.

Definition (Distance Surface)

A distance surface is implicitly defined by a function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ that characterizes $A \subset \mathbb{R}^3$, set of points that are on or inside the implicit surface:

$$A = \{x : f(x) \leq 0\}$$

The surface can be also defined with $f^{-1}(0)$, which gives exactly the points on the surface.

Signed Distance Functions : Point-To-Set Distance

We can define the surface from the outside using a *point-to-set* distance:

$$d(x, A) = min_{y \in A} ||x - y||$$

Thus $d(x, A)$, given a point $x \in \mathbb{R}^3$, returns the shortest distance to the surface defined by A.
We can interchange $d(x, A)$ with $d(x, f^{-1}(0))$ even if they are slightly different, because here we'll handle points that aren't in surfaces interiors.

Signed Distance Functions : Bound and SDF

Definition (Signed Distance Bound)

We say that $f : \mathbb{R}^3 \to \mathbb{R}$ is a signed distance bound of its implicit surface $f^{-1}(0)$ if and only if

$$|f(x)| \leq d(x, f^{-1}(0)) \tag{1}$$

Definition (Signed Distance Function)

We say that f is a signed distance function (SDF) when holds

$$|f(x)| = d(x, f^{-1}(0)) \tag{2}$$

The first definition says that a signed distance bound is always at least as cautious as the true distance function $d(x, f^{-1}(0))$ or the SDF.

Signed Distance Functions : DUFs and DIFs

Two other names for the concepts of the previous slide are *distance underestimate (implicit) functions* (DUFs) and *distance implicit functions* (DIFs).
Remembering that $DUF(x) \leq DIF(x)$ for every DUFs and DIFs respecting the definitions above.
We will interchange Signed Distance Bound with DUF and SDF with DIF, because they express the same concepts.

Signed Distance Functions : Lipschitz

Definition (Lipschitz Function)

We say that a function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ is Lipschitz over a domain D if and only if there is a positive, finite, constant λ such that

$$|f(x) - f(y)| \leq \lambda ||x - y|| \tag{3}$$

Such λ is called the Lipschitz constant.

There is no upper limit to λ, but there is a lower bound. Let $Lip(f)$ be the function returning such minimum value.

Signed Distance Functions : Lipschitz

Manipulating 3 we can observe that

$$\lambda \geq \frac{|f(x) - f(y)|}{||x - y||}$$

$$\lambda \geq \frac{f(x) - f(y)}{||x - y||}$$

$$\lambda \geq \lim_{||x-y|| \to 0} \frac{f(x) - f(y)}{x - y} = f'(x)$$

the last step, given the continuity of f, permits us to estimate a safe lower bound for the Lipschitz constant.

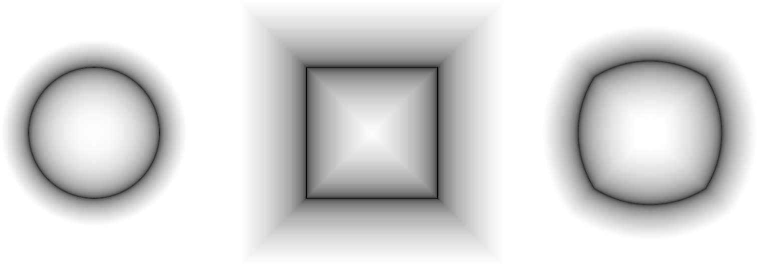
Theorem

Let f be Lipschitz with Lipschitz constant λ. Then the function f/λ is a SDF of its implicit surface.

Proof.

In the assignment. □

Signed Distance Functions : Examples



The images can be seen as 2D section of 3D objects done with a yz-aligned plane passing through the origin.

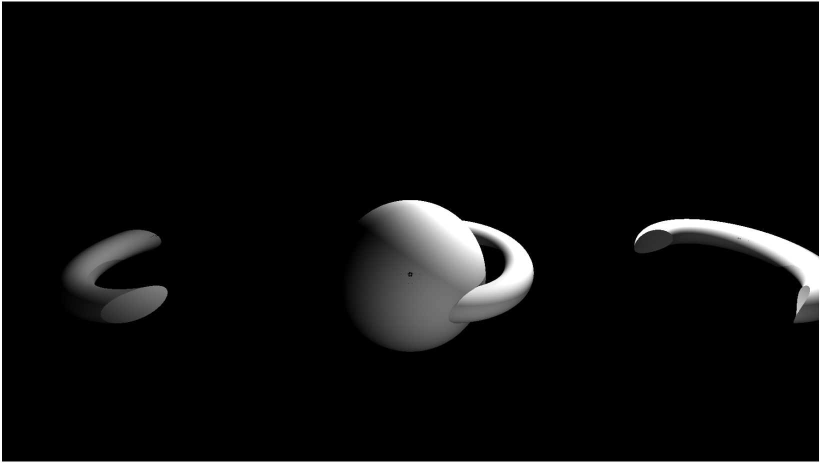
$$\begin{aligned} \text{circle} &: x^2 + y^2 - 1 \\ \text{square} &: max(|x|, |y|) - 1 \\ \text{squircle} &: k * (max(|x|, |y|) - 1) + (1 - k) * (x^2 + y^2 - 1) \\ &\quad \text{where } k \in [0, 1] \end{aligned}$$

Signed Distance Functions : Constructive Solid Geometry

SDFs make easy to create complex shapes from few simple primitives. This technique it known as Constructive Solid Geometry (CSG).

- union $min(f_1, f_2)$
- intersection $max(f_1, f_2)$
- subtraction $max(f_1, -f_2)$
- mixing $k * f_1 + (1 - k) * f_2$ with $k \in [0, 1]$

Signed Distance Functions : Constructive Solid Geometry



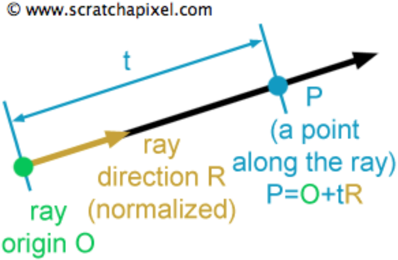
From left to right: intersection, union and subtraction of a sphere and a torus.

Ray Tracing : Rays and Cameras

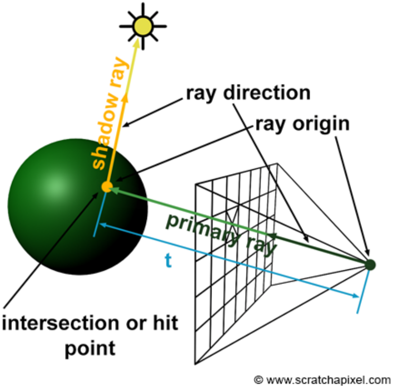
A ray is defined as a point and a direction in formulas:

$$r(t) = O + tR$$

where $O, R \in \mathbb{R}^3$ and R is a unit vector.
At each time t we can compute a point position on the ray.



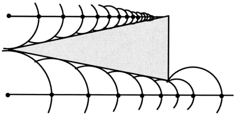
Ray Tracing : Rays and Cameras



A pinhole camera inside a scene with a sphere and a light source.

Ray Tracing : Sphere Tracing

Ray Marching proceeds along the rays with a fixed step, in Sphere Tracing an adaptive step is used.
The step size is given by a SDF (or more generally by a DUF).
$d(x, S)$, for some $x \in \mathbb{R}^3$ and surface S, means we can move x in every direction by $d(x, S)$ being sure at worst to just hit the surface.
In other words we are drawing a safe sphere around the point, in which there is no surface. That's an *unbounding sphere*.



Ray Tracing : Sphere Tracing

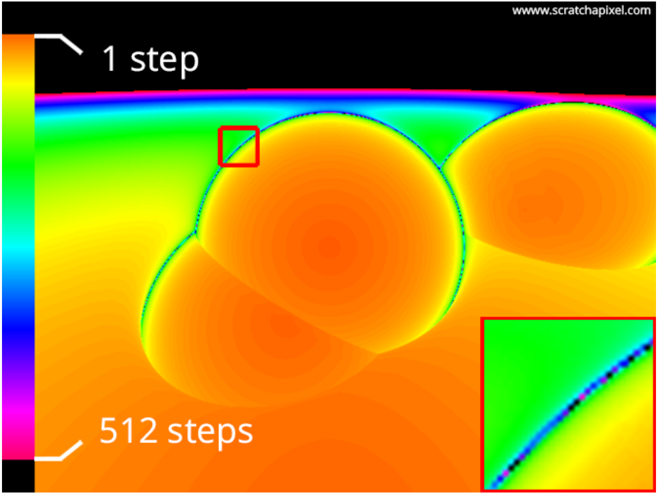
Theorem

Given a function $f : \mathbb{R} \to \mathbb{R}$ *with the Lipschitz bound* $\lambda \geq Lip(f)$ *and an initial point* t_0*, sphere tracing converges linearly to the smallest root grater than* t_0*.*

Proof.

In the assignment. □

Ray Tracing : Sphere Tracing



Ray Tracing : Lighting and Shadows

To correctly compute the shading we need the surface normal at the point.
With SDFs the normal can be computed as the gradient ∇f of the implicit function $f(x, y, z)$ at $P = \langle x_0, y_0, z_0 \rangle$.
Observing the rate-of-change in the proximity of P can be done numerically with

$$
\begin{aligned}
\nabla f_{x_0} &= f(x_0 + \delta, y_0, z_0) - f(x_0 - \delta, y_0, z_0) \\
\nabla f_{y_0} &= f(x_0, y_0 + \delta, z_0) - f(x_0, y_0 - \delta, z_0) \\
\nabla f_{z_0} &= f(x_0, y_0, z_0 + \delta) - f(x_0, y_0, z_0 - \delta)
\end{aligned}
$$

$$\nabla f = \langle \nabla f_{x_0}, \nabla f_{y_0}, \nabla f_{z_0} \rangle$$

Ray Tracing : Lighting and Shadows

The surface normal is used inside the rendering equation.
The following gives the amount of light reflected by the surface

$$\frac{(n \cdot l)c_{light}}{4\pi}$$

where n is the surface normal, l a unit vector pointing the light and c_{light} its color.
Remember that the dot product $n \cdot l$ gives the cosine of the angle of incidence of the light rays.
The denominator comes from the solution of an integral over a hemisphere centered in P.

Ray Tracing : Lighting and Shadows

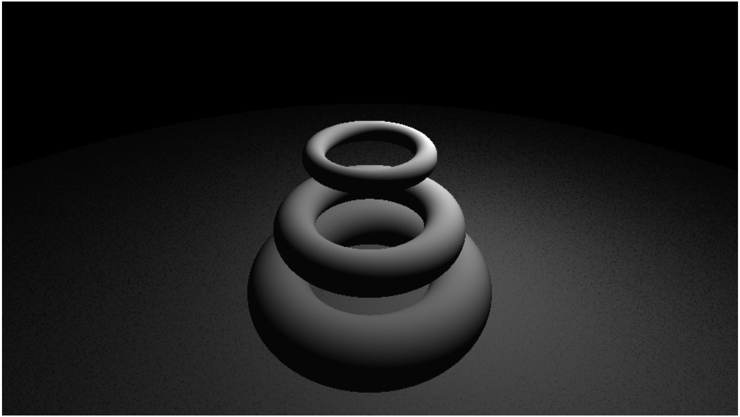


Image with shadows disabled.

Ray Tracing : Lighting and Shadows

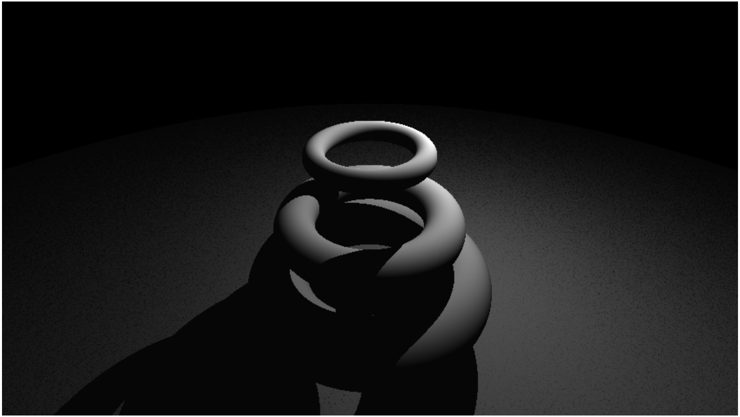
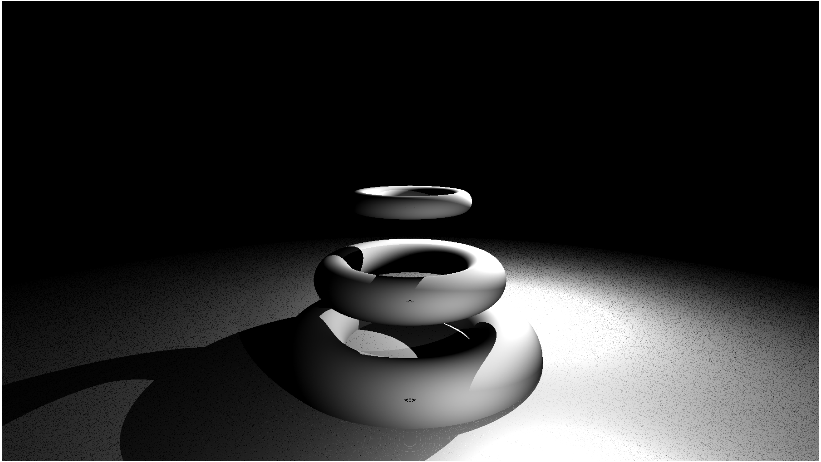


Image with shadows enabled.

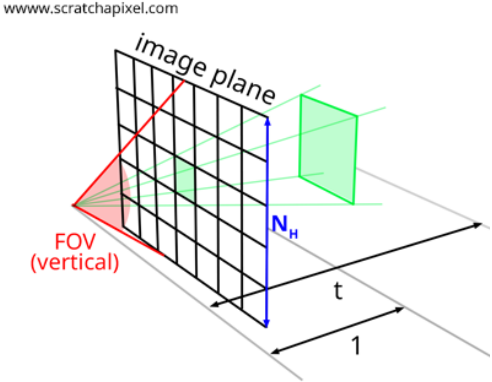
Ray Tracing : Lighting and Shadows



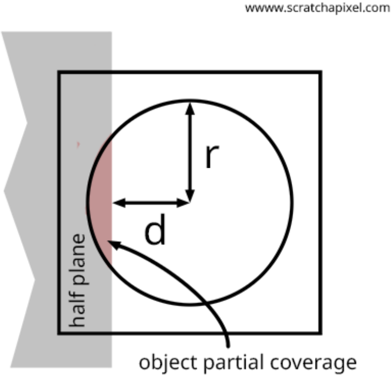
Note the incorrect “in shadow” pixels.

Ray Tracing : Anti-Aliasing

In ray tracing aliasing can't be fully resolved.
The common solution is to sample more colors per pixel and average the results, but doing so is costly.
Cones can be seen as a set of rays.



Ray Tracing : Anti-Aliasing



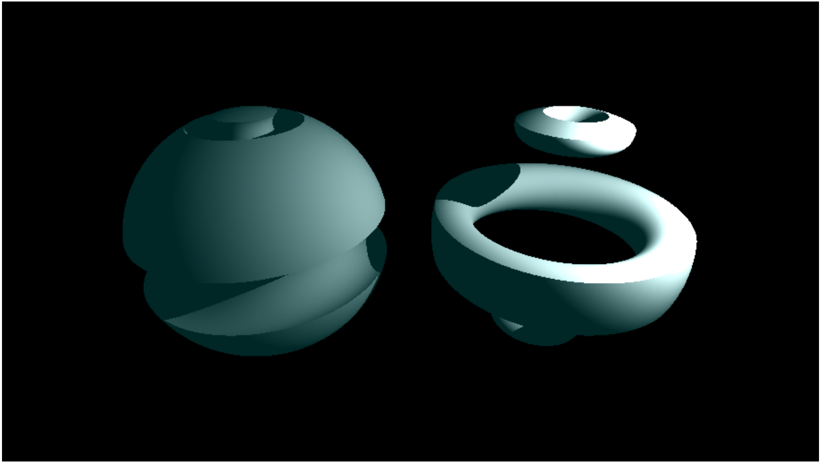
A section of cone, pixel extension and unbounding sphere.

Implementation

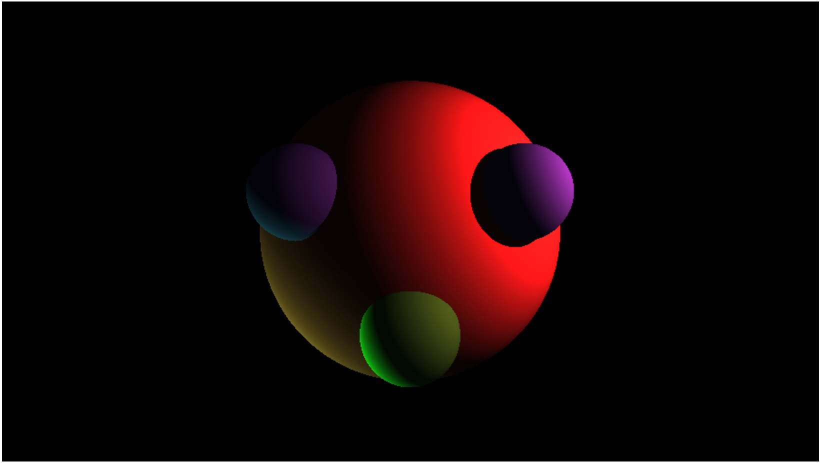
- C++ with two libraries: SDL2 and OpenGL Mathematics (GLM);
- all implemented “from scratch” starting from the theory or examples/tutorials;
- rendering done sequentially on CPU (space for future improvements);
- the code is cross platform and can be compiled easily with CMake;

Implementation : Examples

Implementation : Examples



Implementation : Examples



Thank You