11/17/2017

Cássio

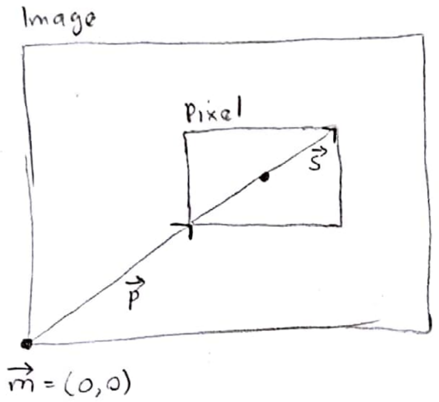
(1)

→ Getting Started on Ray Tracing

→ Intersections and ray computation

Let's start with our image.

We want a vector pointing at the center of the pixel



$\vec{m}$: bottom left of image

$\vec{p}$: bottom left of pixel

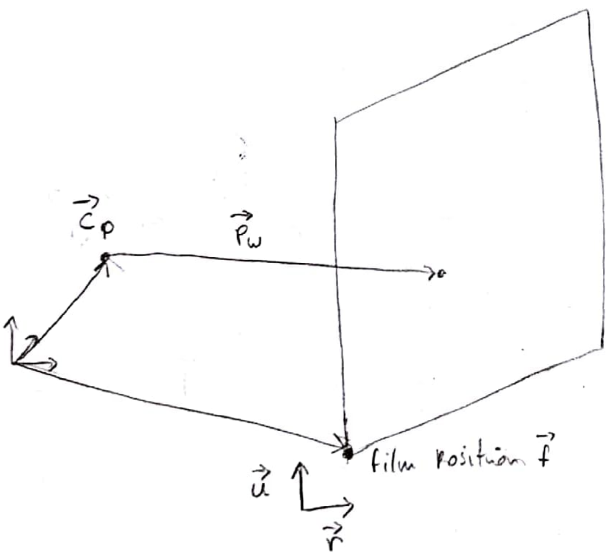
$\vec{s}$: pixel size

$\vec{c}$: center ?

$$\vec{c} = \vec{m} + \vec{p} + \frac{1}{2}\vec{s}$$

camera.h — Cell_Center

2D is boring. Let's take a look at 3D.



$\vec{c_p}$: camera position

$\vec{f}$: film position

$\vec{u}$: up in the film

$\vec{r}$: right in the film.

$\vec{P_w}$: pixel position in 3D.

$$\vec{P_w} = \vec{f} + c_x \vec{u} + c_y \vec{r}$$

camera.cpp — World_Position

How to Render Pixel?

→ calculate ray

→ cast ray → find intersections → determine color

→ set pixel in the image according to color.

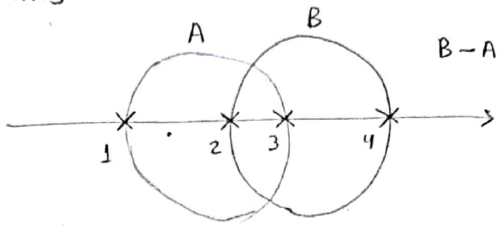
$$\text{ray} \begin{cases} \text{endpoint : camera position} \\ \text{direction : } (\vec{P_w} - \vec{C_p}) \leftarrow \text{normalized} \end{cases}$$

render_world.cpp — Render_Pixel

How to cast ray ?

→ Check for closest intersection

→ Record hits



B − A ⇒ Enter in 3, out in 4.

→ Calculate normal (Shader computation)

→ Shade surface

→ No intersection ? Shade background

Render_world.cpp − Cast_Ray

How to calculate intersection ?

→ hit at $t = \infty$

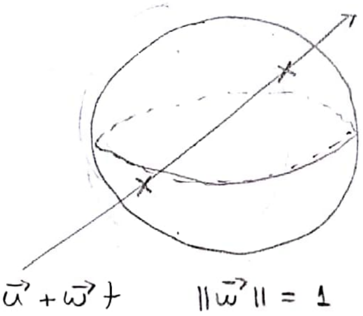
→ for each object, compute the intersections with ray

→ intersects with lower t ?

→ store closest obj / store t

Render_world.cpp − Closest_Int.

→ First test we have is a sphere. (00.txt / 00.png)



center: $\vec{c}$
radius: r

ray : $\vec{u} + \vec{w}t$ $\quad \|\vec{w}\| = 1$

On sphere: $\|\vec{x} - \vec{c}\|^2 = r^2 \Rightarrow (\vec{x} - \vec{c})(\vec{x} - \vec{c}) = r^2$

replacing ray: $(\vec{u} + \vec{w}t - \vec{c})(\vec{u} + \vec{w}t - \vec{c}) = r^2$

replacing $\vec{u} - \vec{c} = \vec{v} \Rightarrow (\vec{w}t + \vec{v})(\vec{w}t + \vec{v}) = r^2$

expanding: $(\vec{w}\cdot\vec{w})t^2 + 2(\vec{w}\cdot\vec{v})t + (\vec{v}\cdot\vec{v}) = 0$

$$t = \frac{(\vec{w}\cdot\vec{v}) \pm \left[(\vec{w}\cdot\vec{v})^2 - (\vec{w}\vec{w})(\vec{v}\vec{v} - r^2)\right]^{1/2}}{(\vec{w}\cdot\vec{w})}$$

→ Discriminant D

$D < 0$: no intersection ()

$D = 0$: intersection (t, t)

$D > 0$: intersection (t_0, t_1)

$t_0 < t_1$

$t_0 < t_1 < 0 \Rightarrow ()$

$t_0 < 0 \leq t_1 \Rightarrow (0, t_1)$

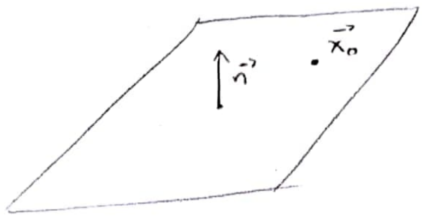
$0 \leq t_0 \leq t_1 \Rightarrow (t_0, t_1)$

What about a plane ? :

Plane implicit equation : $(\vec{x} - \vec{x}_0)\vec{n} = 0$

$\vec{n}$: normal

$\vec{x}_0$: any point on the plane



-> Exercise

-> $t = \dfrac{-\vec{n}(\vec{u} - \vec{x}_0)}{\boxed{\vec{n}\,\vec{w}}}$ -> necessary ?