

Ray Tracing: Notes d'implementació

Aquestes notes complementen les notes de classe amb detalls dels càlculs per a la implementació del Ray tracing

Càlcul de direcció dels raigs

Paràmetres de la càmera:

- O: posició de l'observador
- C: centre de l'escena
- up: vector amunt
- ϑ : obertura horitzontal de la càmera.
- rati: obertura vertical / obert. Horitzontal ($=H/W$)
- H: nombre de píxels verticals (height)
- W: nombre de píxels horitzontals (width)

Passos a seguir:

1. Calculem els eixos de l'observador

$$Z = |O - C|$$

$$X = |up \times Z|$$

$$Y = Z \times X$$

2. Calculem l'amplada w' i alçada h' a partir de l'obertura ϑ i el rati (veure slides)
3. Calculem els desplaçaments Δx , Δy a partir dels eixos X i Y i la relació del nombre de píxels W, H i l'amplada i alçada w', h'

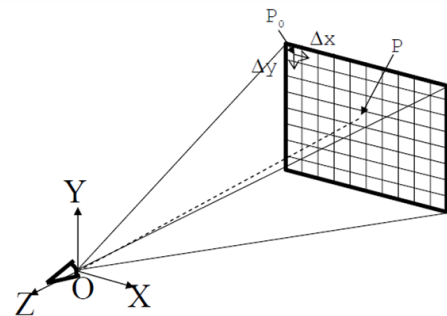
$$\Delta x = X * w' / W = X * 2 * \tan(\vartheta/2) / W$$

$$\Delta y = Y * h' / H = Y * \text{rati} * 2 * \tan(\vartheta/2) / H$$

4. Calculem el punt P i el punt inicial P_0

$$P = O - Z$$

$$P_0 = P - \Delta x * (W - 1) / 2 + \Delta y * (H - 1) / 2$$



Recomanació:

Feu un precàlcul de P_0 , Δx i Δy abans d'iterar per els píxels, no cal calcular-les a cada vegada.

Algorisme de Ray Casting

```
for(x = 0; x < Screen.width; x++){
    for (y = 0; y < Screen.height; y++){
        rdir = computeRay(Camera, x, y);
        color = intersecScene(Scene, Camera.position, rdir);
        plot(x, y, color);
    }
}
```

```
// Calcuem la direcciò per a cada pixel
function computeRay(Cam,x,y)
{
    return (P0 + Δx * x - Δy * y - O).normalize;
}
```

Exemple 400x400 pixels:

```
var Camera = {
    position: [0,0,0],
    up       : [0,1,0],
    center   : [0,0,-1],
    fov      : 60,
}
var Scene = {
    geometry: [{
        radius : 2.0,
        center : [0,0,-10],
        color  : [0.0,1.0,0.0,1.0],
    },
    ],
};

var lights = {
    data: [{
        position: [0,0,0],
        color   : [0.67,0.67,0.67,1]
    },
    ],
}
```

