Radiant Rhythm
-
Illumination on a Chaotic Pendulum

B10901042 Wen Perng
NTUEE
2023.05.31

Project Goal

The goal is to simulate the **chaotic pendulum** outside the college of engineering building.

Simulation includes:

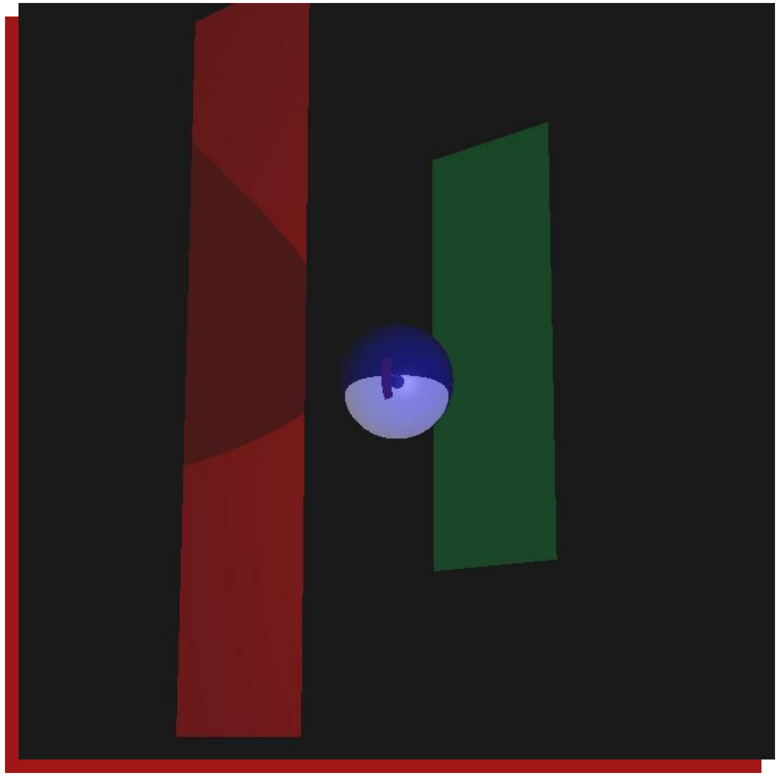
1. the chaotic motion of the pendulum
2. the lighting on the metallic beams
3. the lighting on the diffusive pendulum frame
4. the background.

Results

Development

The first step is to create the required geometric objects in the light tracing engine.

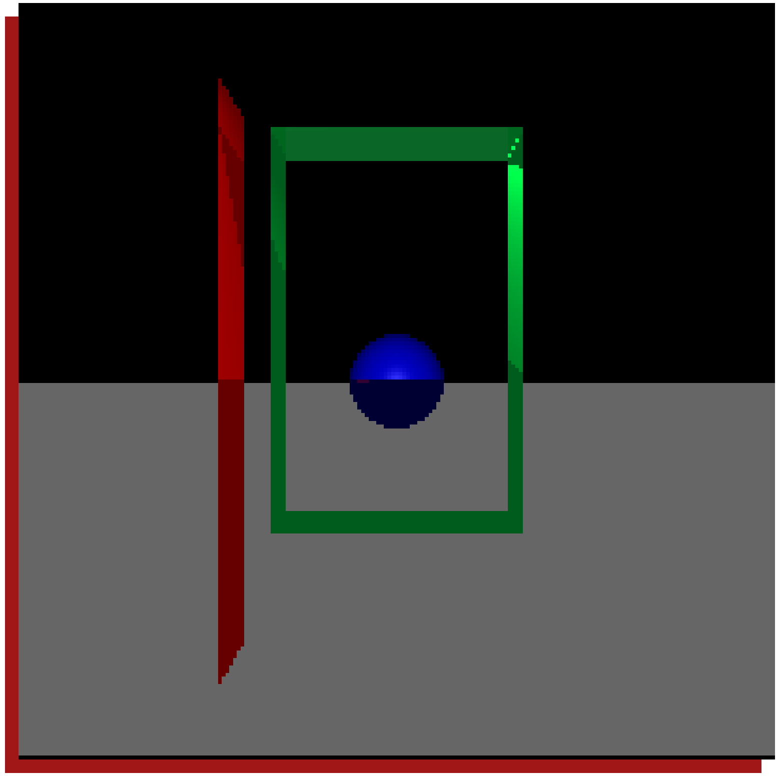
1. Planes



Development

The first step is to create the required geometric objects in the light tracing engine.

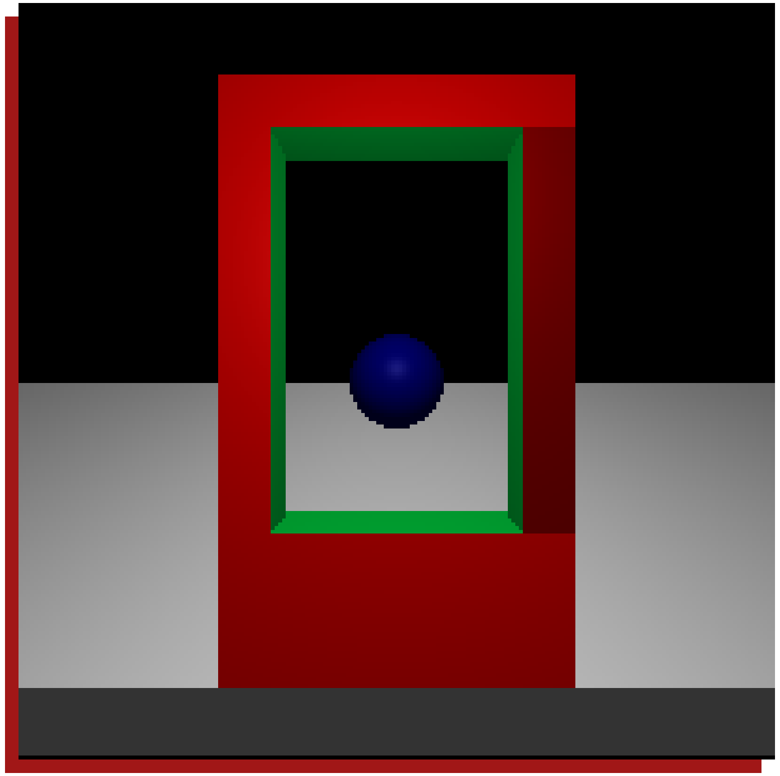
1. Planes



Development

The first step is to create the required geometric objects in the light tracing engine.

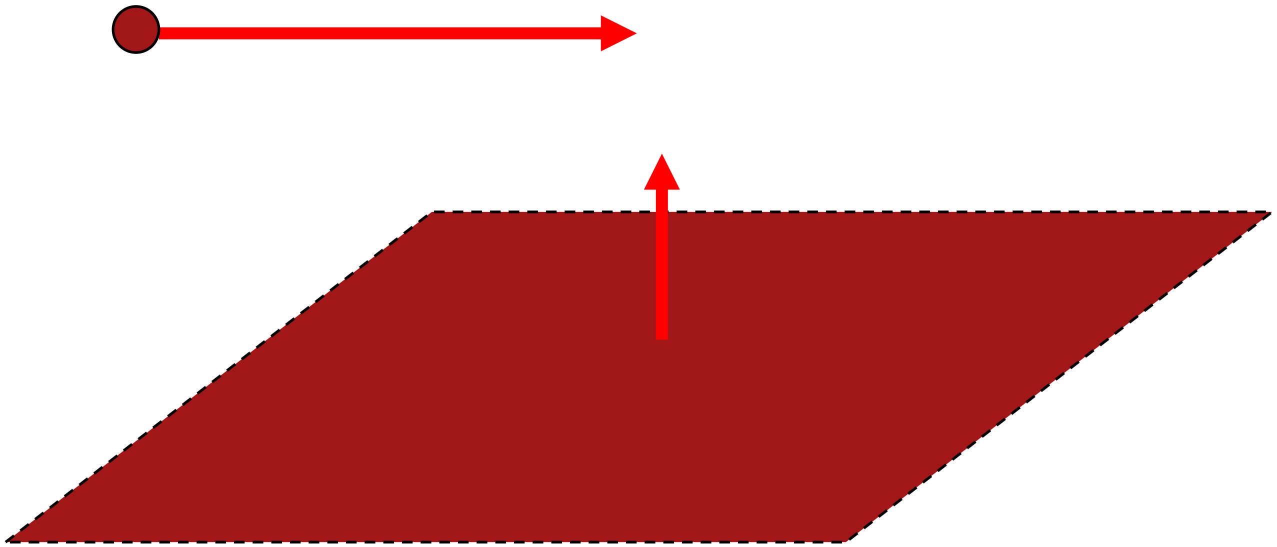
1. Planes



Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes

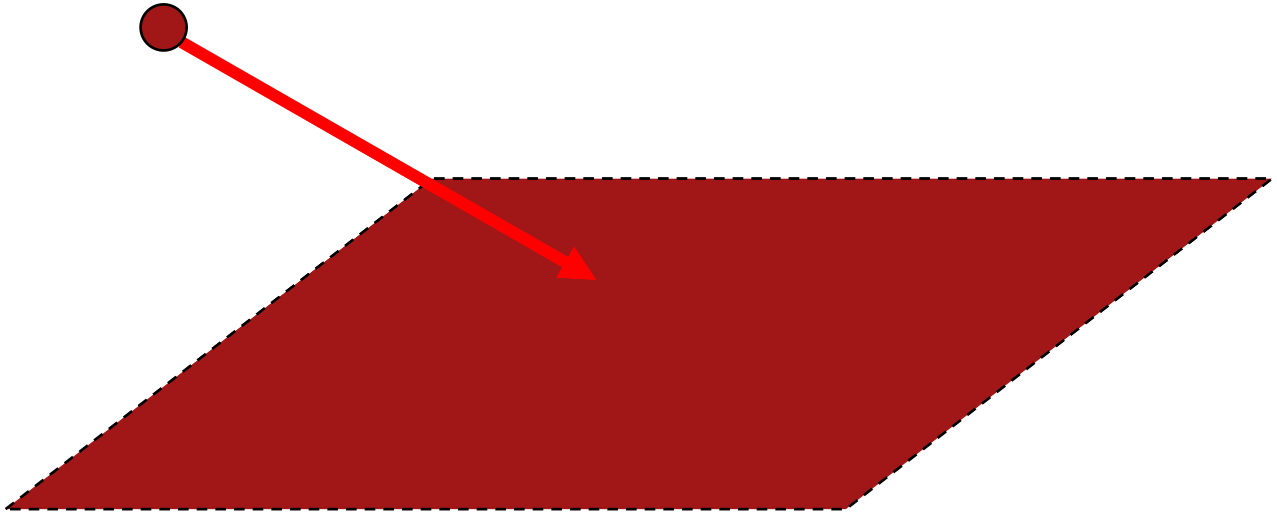


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes

$$\text{Point} = \text{ray.origin} + t \cdot \text{ray.direction}$$



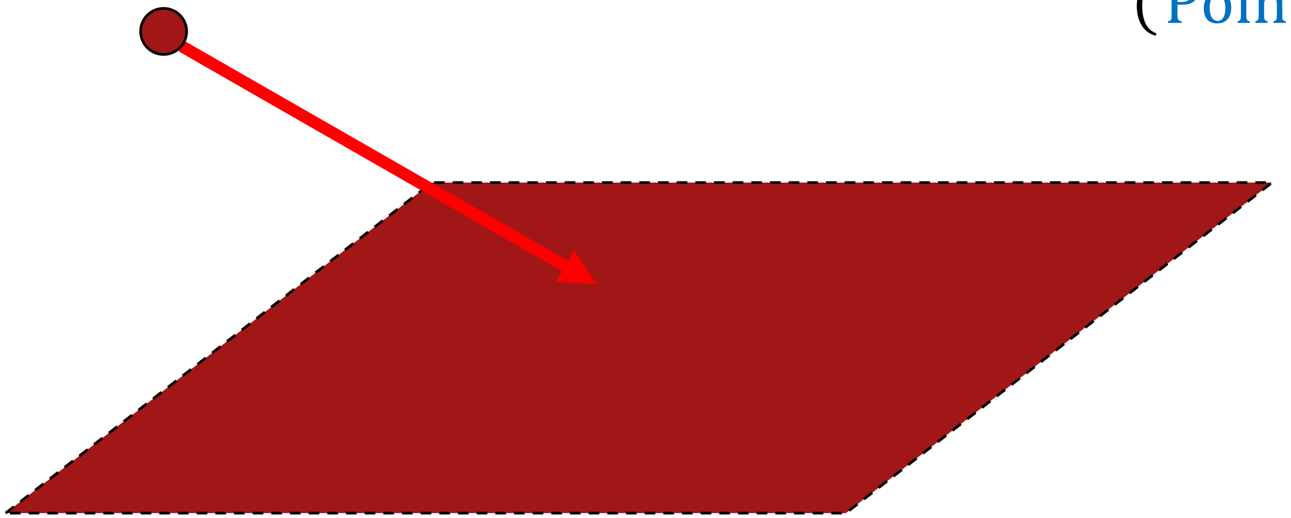
Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes

$$\text{Point} = \text{ray.origin} + t \cdot \text{ray.direction}$$

$$(\text{Point} - \text{Plane.ref}) \cdot \text{Plane.normal} = 0$$

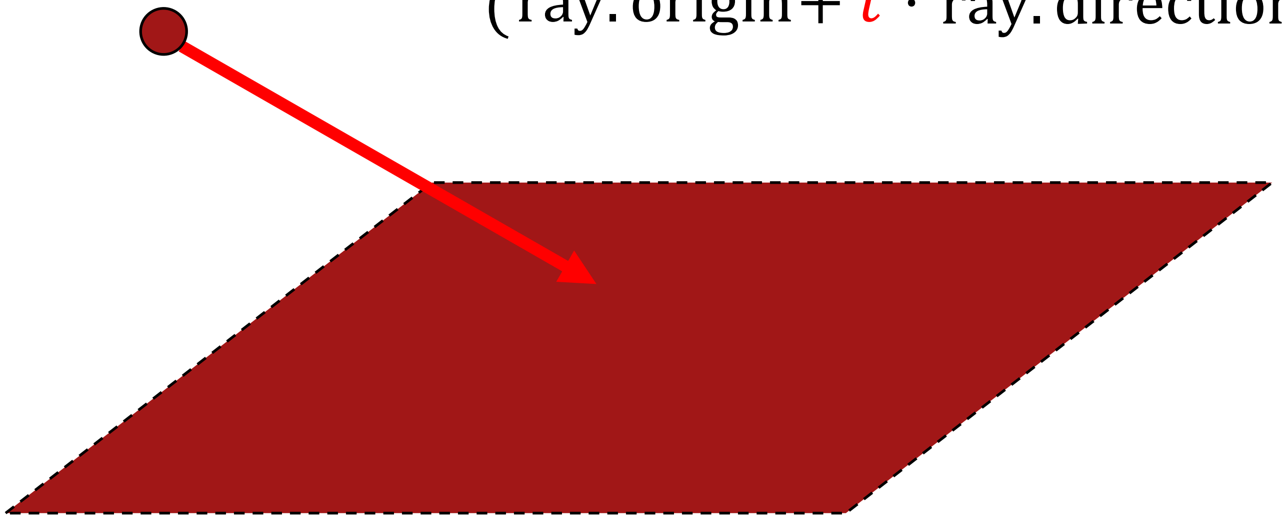


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes

$$(\mathrm{ray.origin} + t \cdot \mathrm{ray.direction} - \mathrm{Plane.ref}) \cdot \mathrm{Plane.normal} = 0$$

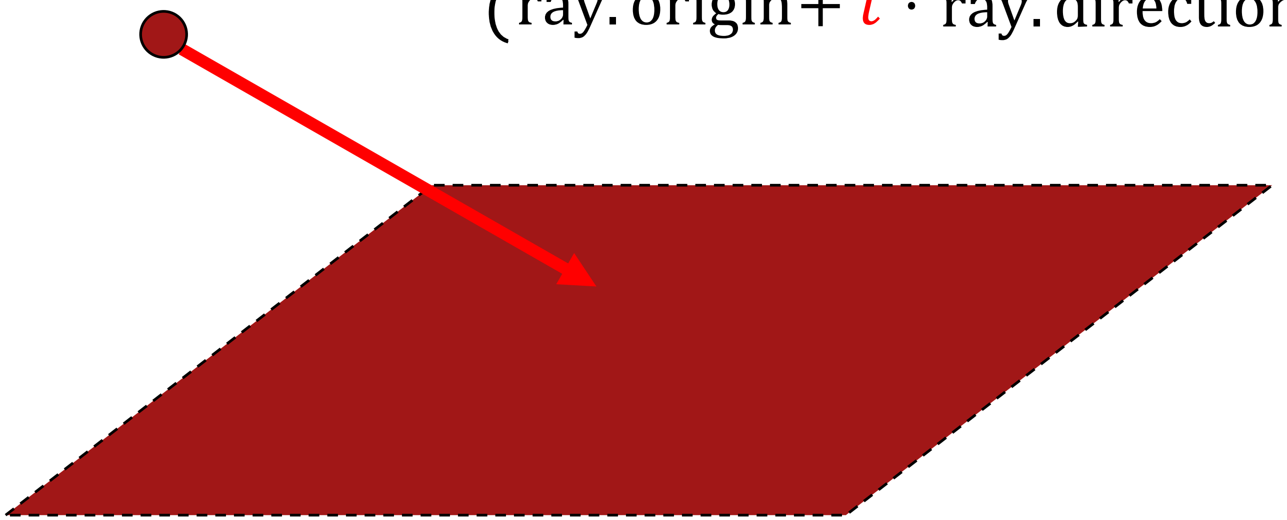


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes

$$(\mathrm{ray.origin} + {\color{red}t} \cdot \mathrm{ray.direction} - \mathrm{Plane.ref}) \cdot \mathrm{Plane.normal} = 0$$

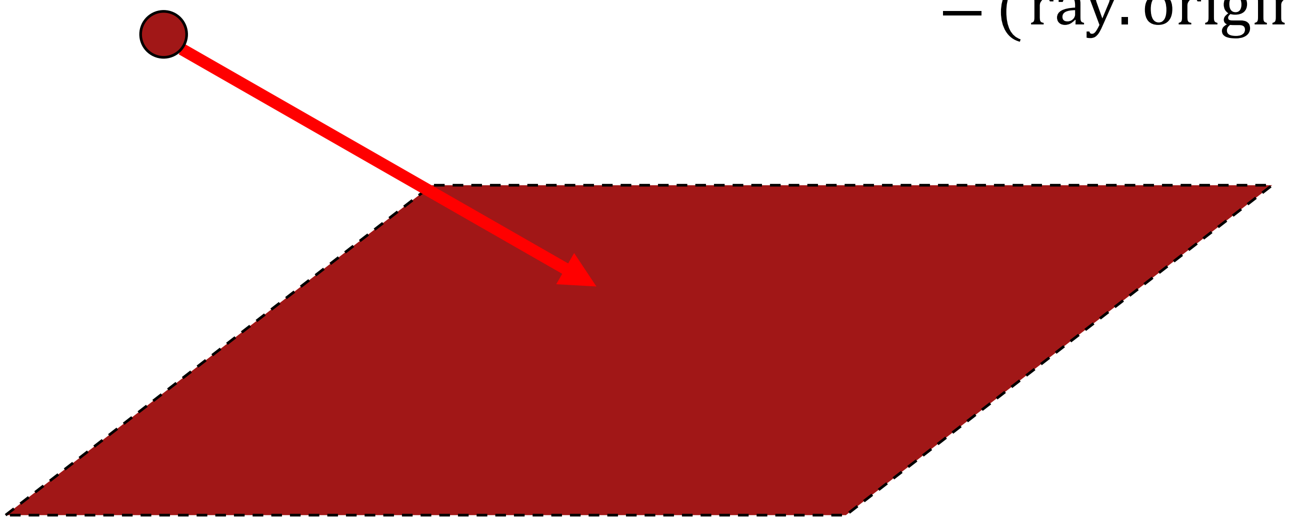


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes

$$t \cdot (\text{ray.direction} \cdot \text{Plane.normal}) = -(\text{ray.origin} - \text{Plane.ref}) \cdot \text{Plane.normal}$$



Development

The first step is to create the required geometric objects in the light tracing engine.

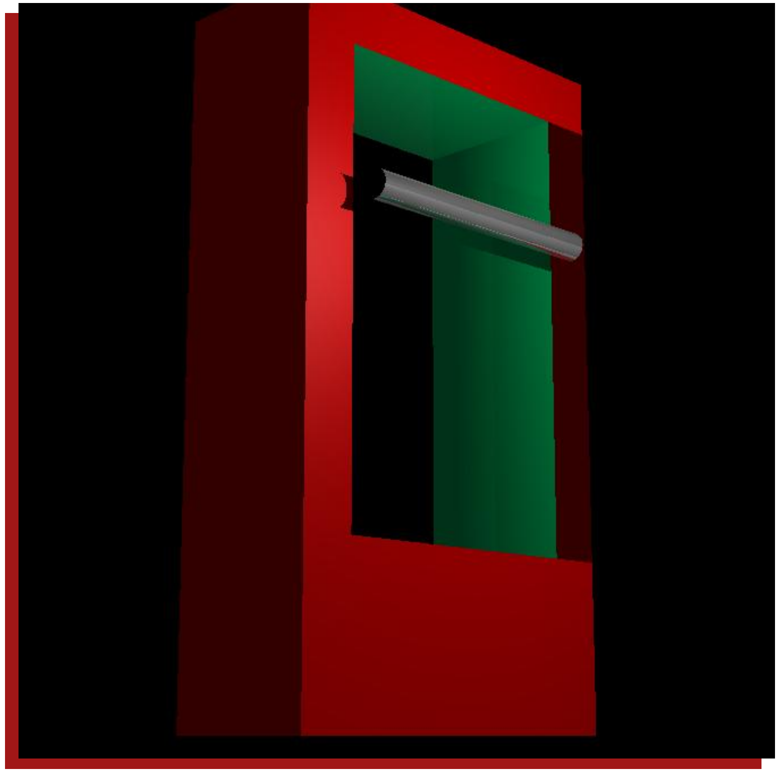
1. Planes

$$t = -\frac{(\text{ray.origin} - \text{Plane.ref}) \cdot \text{Plane.normal}}{(\text{ray.direction} \cdot \text{Plane.normal})}$$

Development

The first step is to create the required geometric objects in the light tracing engine.

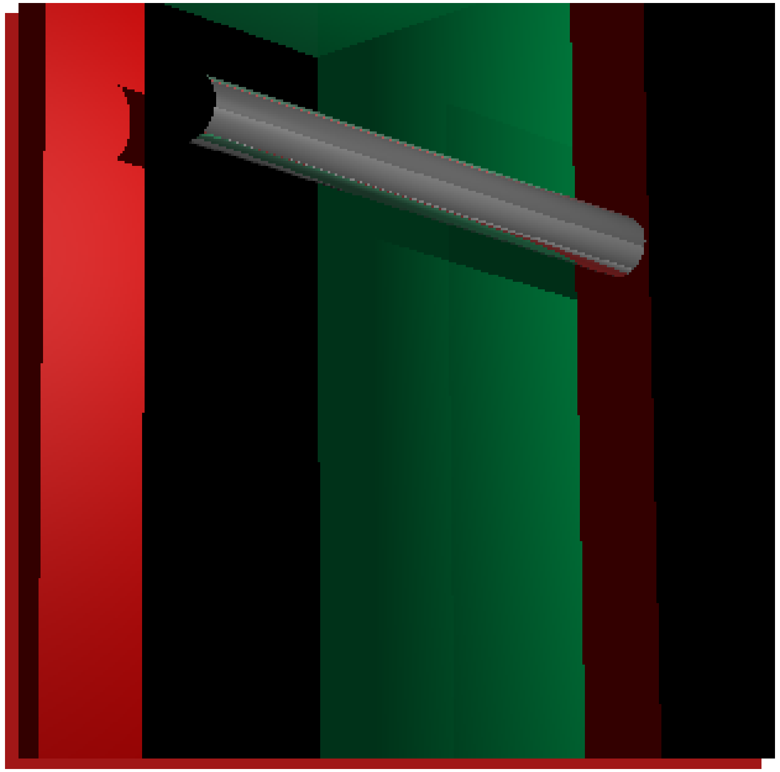
1. Planes
2. Cylinders



Development

The first step is to create the required geometric objects in the light tracing engine.

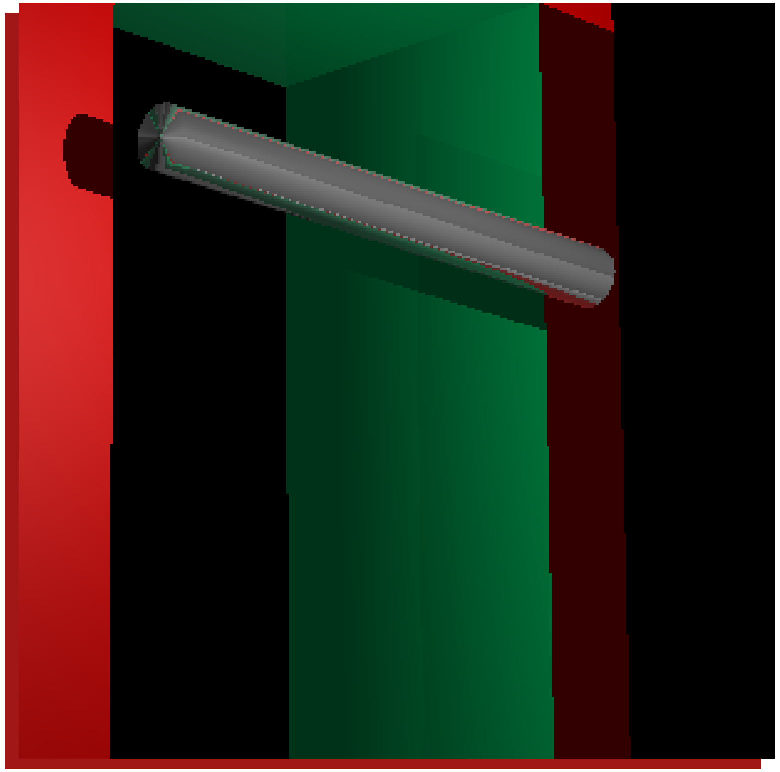
1. Planes
2. Cylinders



Development

The first step is to create the required geometric objects in the light tracing engine.

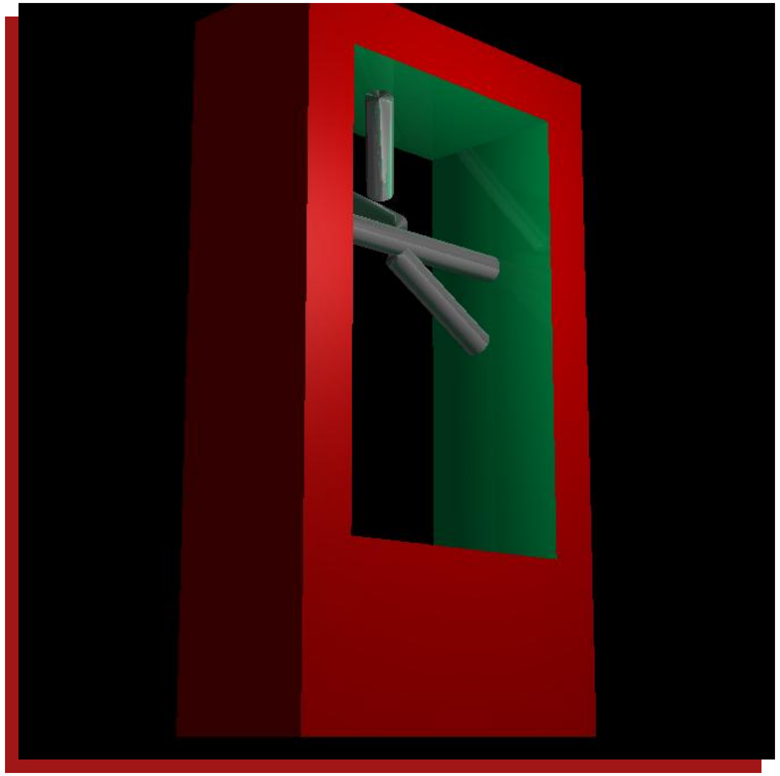
1. Planes
2. Cylinders



Development

The first step is to create the required geometric objects in the light tracing engine.

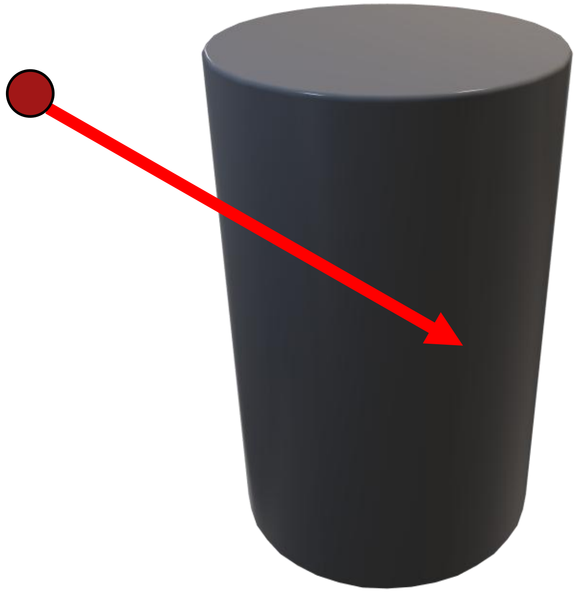
1. Planes
2. Cylinders



Development

The first step is to create the required geometric objects in the light tracing engine.

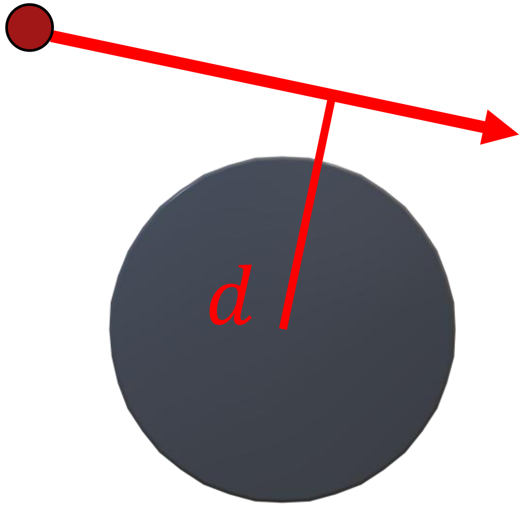
1. Planes
2. Cylinders



Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes
2. Cylinders

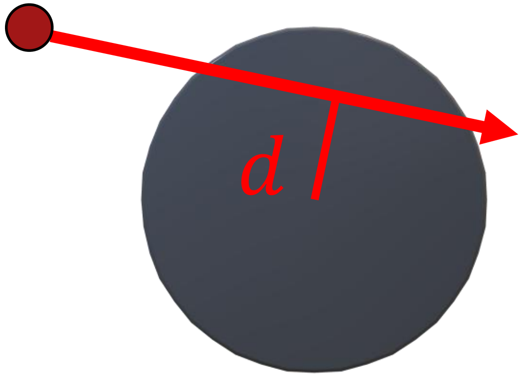


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes
2. Cylinders

$$\text{Point} = \text{ray.origin} + t \cdot \text{ray.direction}$$



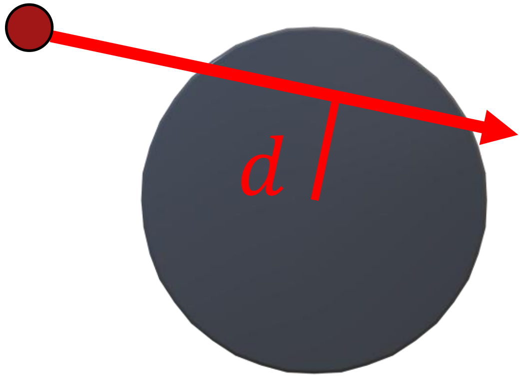
Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes
2. Cylinders

$$\text{Point} = \text{ray.origin} + t \cdot \text{ray.direction}$$

$$||\text{Point} - \text{Cyl.center}||^2 = \text{Cyl.radius}^2$$

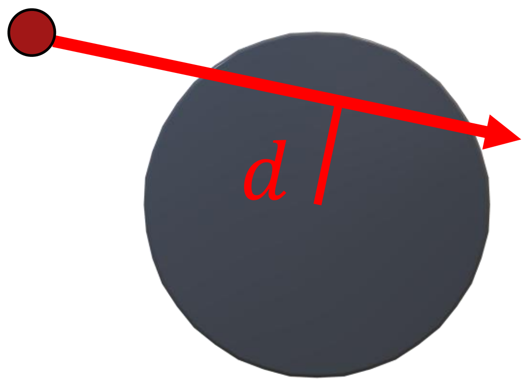


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes
2. Cylinders

$$||\mathrm{ray.origin} + t \cdot \mathrm{ray.direction} - \mathrm{Cyl.center}\,||^2 = \mathrm{Cyl.radius}^2$$

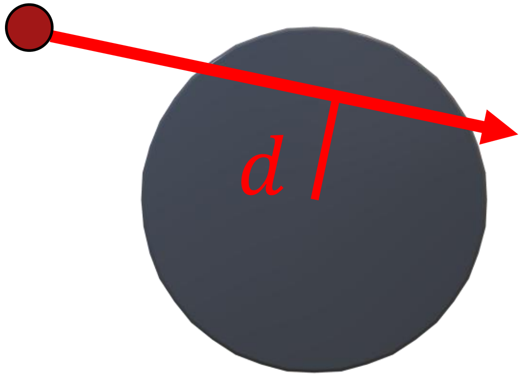


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes
2. Cylinders

$$||\mathrm{ray.origin} + \boldsymbol{t} \cdot \mathrm{ray.direction} - \mathrm{Cyl.center}\,||^2 = \mathrm{Cyl.radius}^2$$

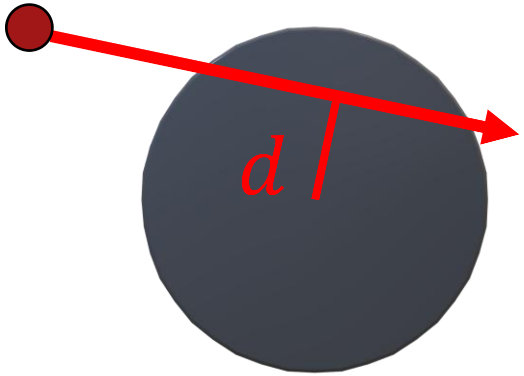


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes
2. Cylinders

$$t^2(\text{ray.direction} \cdot \text{ray.direction}) + t\,(\text{ray.origin} - \text{Cyl.center}) \cdot \text{ray.direction} + (\text{ray.origin} - \text{Cyl.center}) \cdot (\text{ray.origin} - \text{Cyl.center}) = \text{Cyl.radius}^2$$

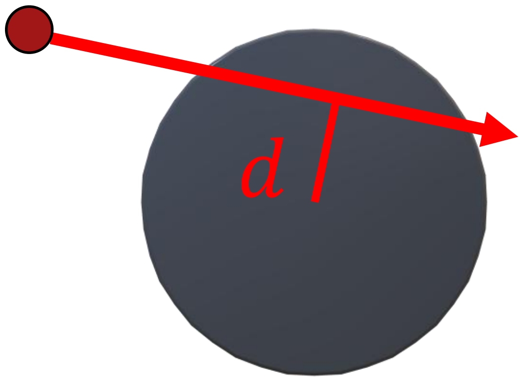


Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes
2. Cylinders

$$
\begin{aligned}
& t^2 a \\
& + t\ \mathrm{b} \\
& + c \\
& = 0
\end{aligned}
$$



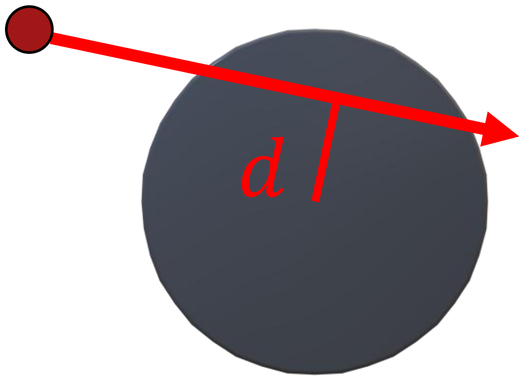
Development

The first step is to create the required geometric objects in the light tracing engine.

1. Planes
2. Cylinders

$$t^2 a + t\ \mathrm{b} + c = 0$$

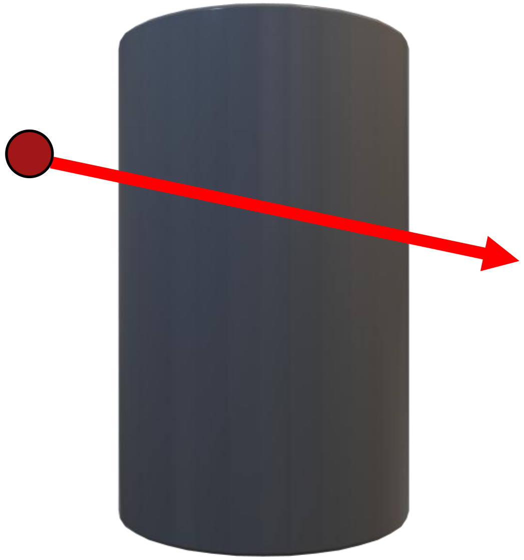
$$t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$



Development

The first step is to create the required geometric objects in the light tracing engine.

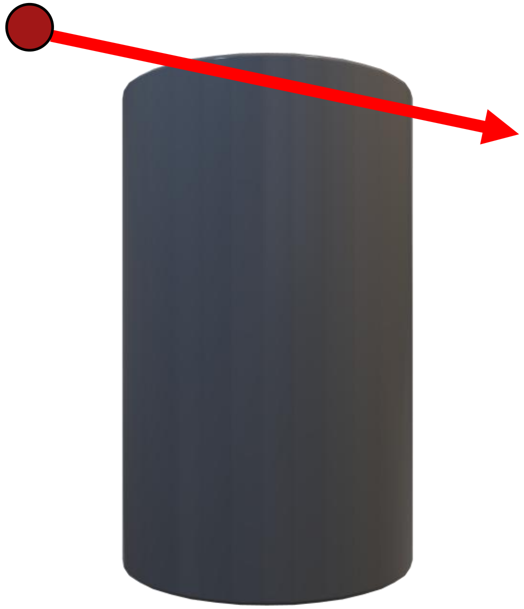
1. Planes
2. Cylinders



Development

The first step is to create the required geometric objects in the light tracing engine.

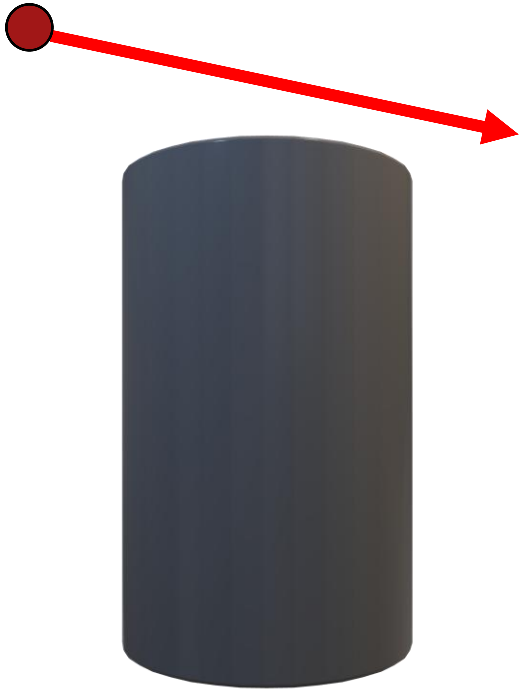
1. Planes
2. Cylinders



Development

The first step is to create the required geometric objects in the light tracing engine.

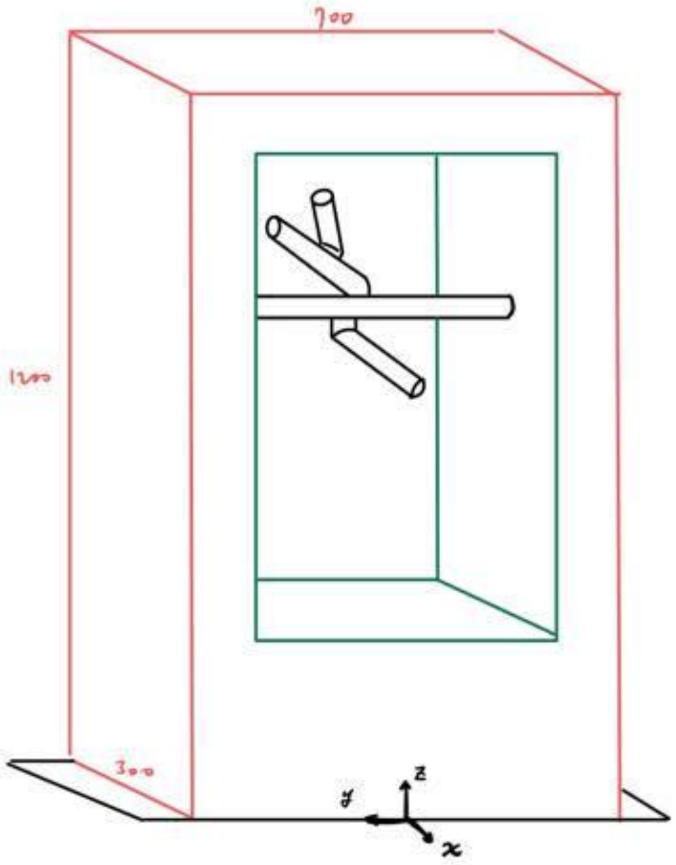
1. Planes
2. Cylinders



Development

Next, to create the dynamic motion of the pendulum.

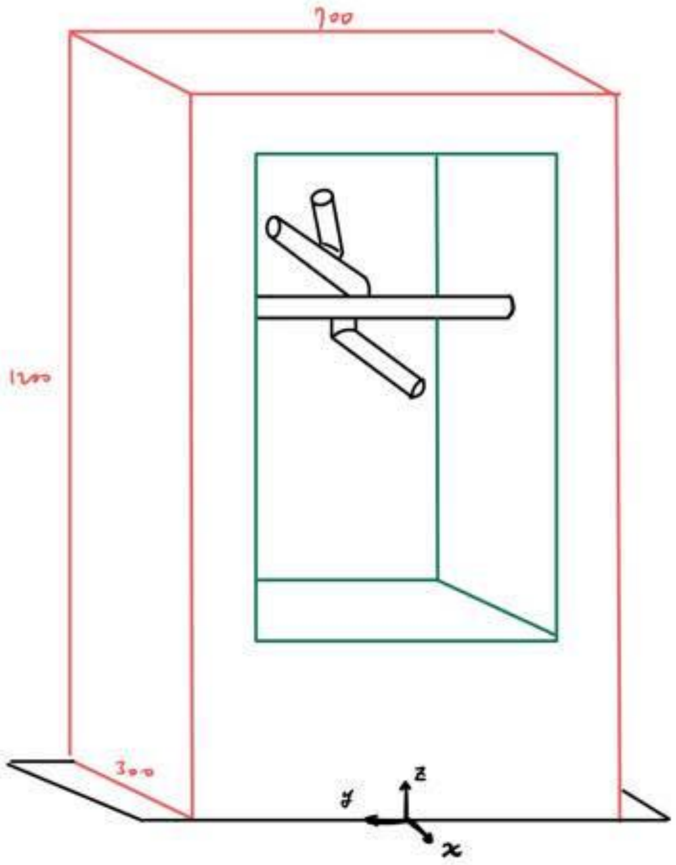
1. Generate a coordinate system.
2. Find the equations of motion.
3. Using finite difference method to iterate the motion.



Development

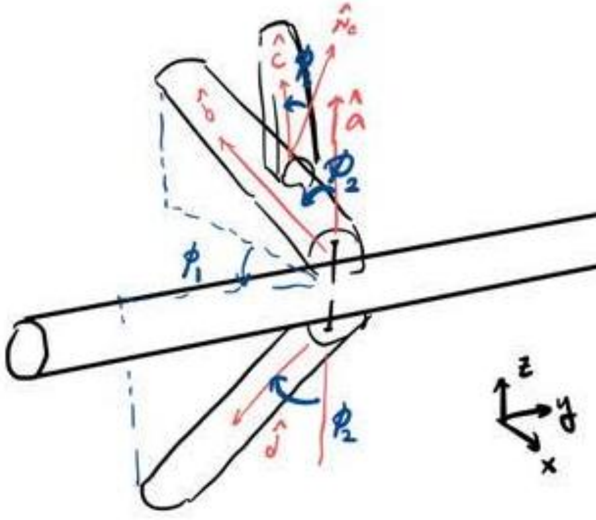
Next, to create the dynamic motion of the pendulum.

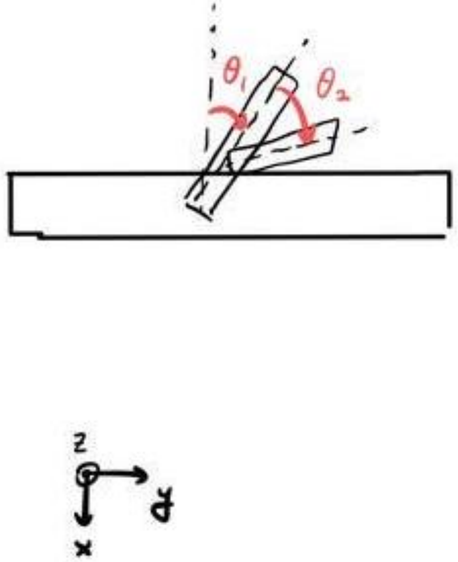
1. **Generate a coordinate system.**
2. Find the equations of motion.
3. Using finite difference method to iterate the motion.



Development

Next, to create the dynamic motion of the pendulum.





Development

Next, to create the dynamic motion of the pendulum.

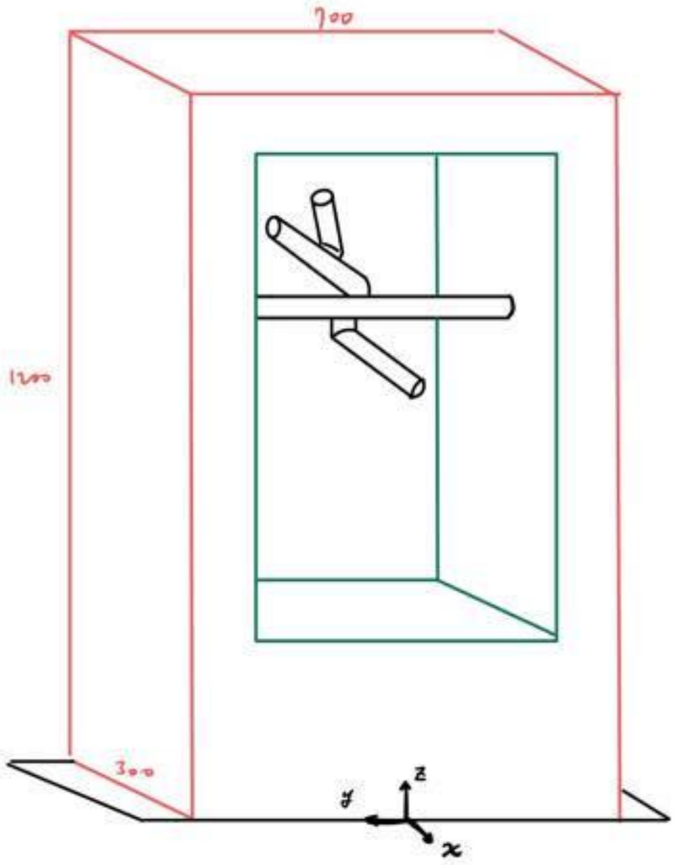
$$
\begin{cases}
\hat{a} = e^{-\hat{z}\hat{y}\frac{\alpha}{2}}(\hat{z})\,e^{\hat{z}\hat{y}\frac{\alpha}{2}} \\
\hat{b} = e^{-\hat{z}\hat{y}\frac{\alpha}{2}}\,e^{-\hat{y}\hat{x}\frac{\theta_1}{2}}\left(-\sin\phi_2\,\hat{y} + \cos\phi_2\,\hat{z}\right)e^{\hat{y}\hat{x}\frac{\theta_1}{2}}\,e^{\hat{z}\hat{y}\frac{\alpha}{2}} \\
\hat{c} = e^{-\hat{z}\hat{y}\frac{\alpha}{2}}\,e^{-\hat{y}\hat{x}\frac{\theta_1}{2}}\left(e^{-\hat{z}\hat{y}\frac{\frac{\pi}{2}-\phi_3}{2}}\,e^{-\hat{y}\hat{x}\frac{\theta_2}{2}}\left(-\sin\phi_3\,\hat{y} + \cos\phi_2\,\hat{z}\right)e^{\hat{y}\hat{x}\frac{\theta_2}{2}}\,e^{\hat{z}\hat{y}\frac{\frac{\pi}{2}-\phi_3}{2}}\right)e^{\hat{y}\hat{x}\frac{\theta_1}{2}}\,e^{\hat{z}\hat{y}\frac{\alpha}{2}} \\
\hat{d} = e^{-\hat{z}\hat{y}\frac{\alpha}{2}}\,e^{-\hat{y}\hat{x}\frac{\theta_1}{2}}\left(e^{-\hat{x}\hat{y}\frac{\phi_1}{2}}\left(-\sin\phi_2\,\hat{y} - \cos\phi_2\,\hat{z}\right)e^{\hat{x}\hat{y}\frac{\phi_1}{2}}\right)e^{\hat{y}\hat{x}\frac{\theta_1}{2}}\,e^{\hat{z}\hat{y}\frac{\alpha}{2}}
\end{cases}
$$

$$
\hat{n}_c = e^{-\hat{z}\hat{y}\frac{\alpha}{2}}\,e^{-\hat{y}\hat{x}\frac{\theta_1}{2}}\left(e^{-\hat{z}\hat{y}\frac{\frac{\pi}{2}-\phi_3}{2}}(\hat{z})\,e^{\hat{z}\hat{y}\frac{\frac{\pi}{2}-\phi_3}{2}}\right)e^{\hat{y}\hat{x}\frac{\theta_1}{2}}\,e^{\hat{z}\hat{y}\frac{\alpha}{2}}
$$

Development

Next, to create the dynamic motion of the pendulum.

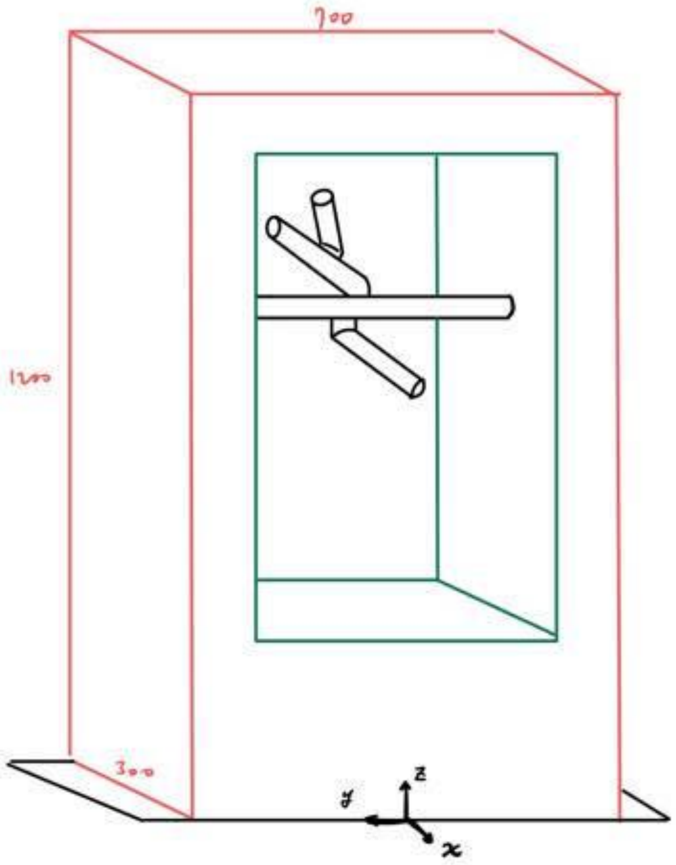
1. **Generate a coordinate system.**
2. Find the equations of motion.
3. Using finite difference method to iterate the motion.



Development

Next, to create the dynamic motion of the pendulum.

1. Generate a coordinate system.
2. **Find the equations of motion.**
3. Using finite difference method to iterate the motion.



Development

Next, to create the dynamic motion of the pendulum.

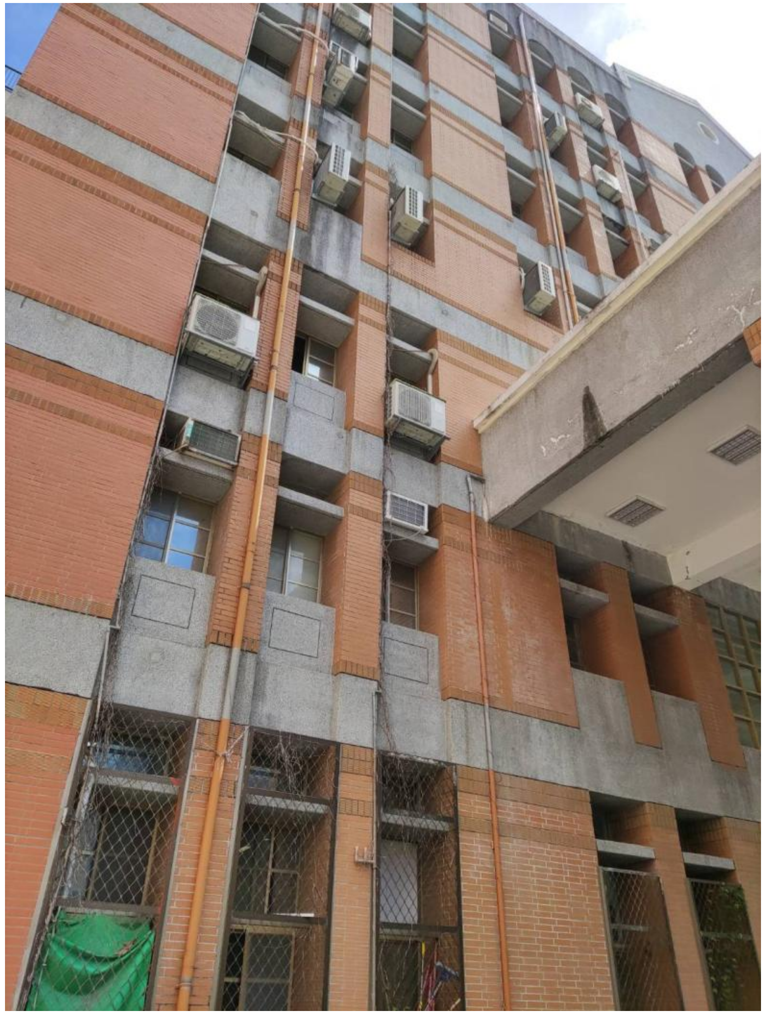
$$\hookrightarrow \vec{\tau}_{\hat{n}_c} = \left[\frac{\ell_2}{2}\hat{c} - \left(\frac{\ell_2}{2}\hat{c}\cdot\hat{n}_c\right)\hat{n}_c\right]\times M_c\vec{g} = I_c\ddot{\vec{\theta}}_2$$

$$\hookrightarrow \vec{\tau}_{\hat{a}} = \left[\left(\frac{\ell_2}{2}\hat{c} + \ell_3\hat{b} + r_1\overset{\hat{n}_c\,+}{\hat{a}}\right) - \hat{a}\left(\frac{\ell_2}{2}\hat{c} + \ell_3\hat{b} + r_1\overset{\hat{n}_c\,+}{\hat{a}}\right)\cdot\hat{a}\right]\times M_c\vec{g}$$

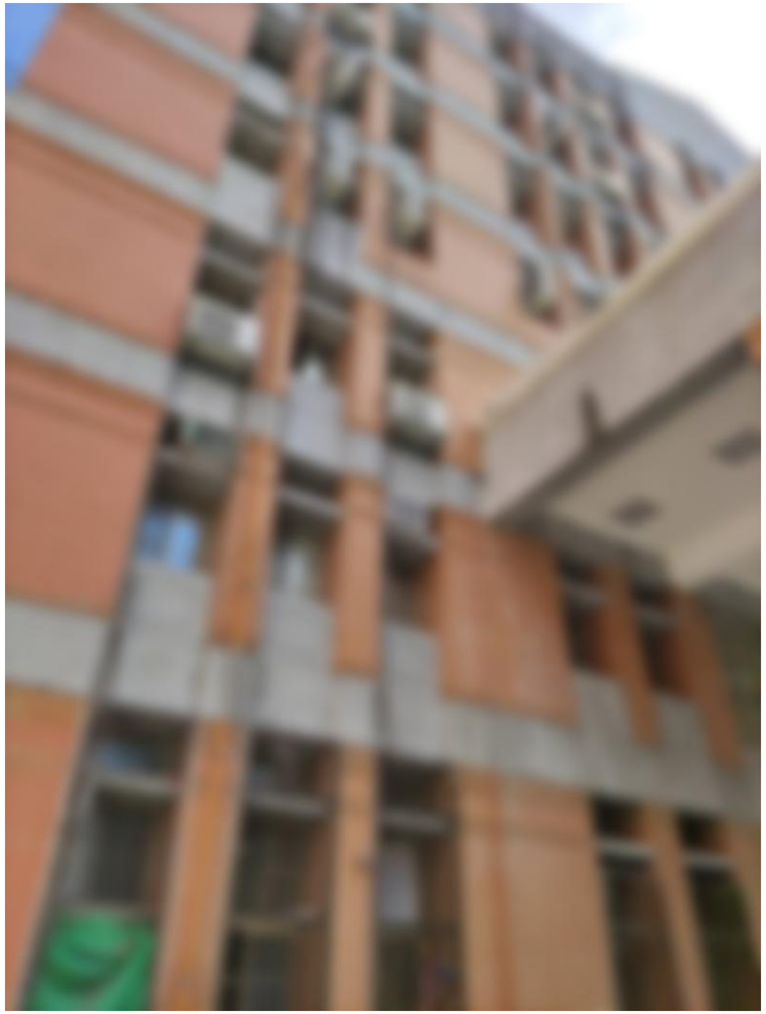
$$+\left[\left(\frac{\ell_1}{2}\hat{b} + r_1\hat{a}\right) + \left(\frac{\ell_1}{2}\hat{d} - r_1\hat{a}\right)\right]\times M_b\vec{g} = I_c\ddot{\vec{\theta}}_1$$

Development

Lastly, generate the blurry background of the college.



imgaussfilt()



Conclusion

Time took: one week

Rendering: 6 hrs (19375 secs)

Conclusion

Time took: one week

Rendering: 6 hrs (19375 secs)

Problems encountered:

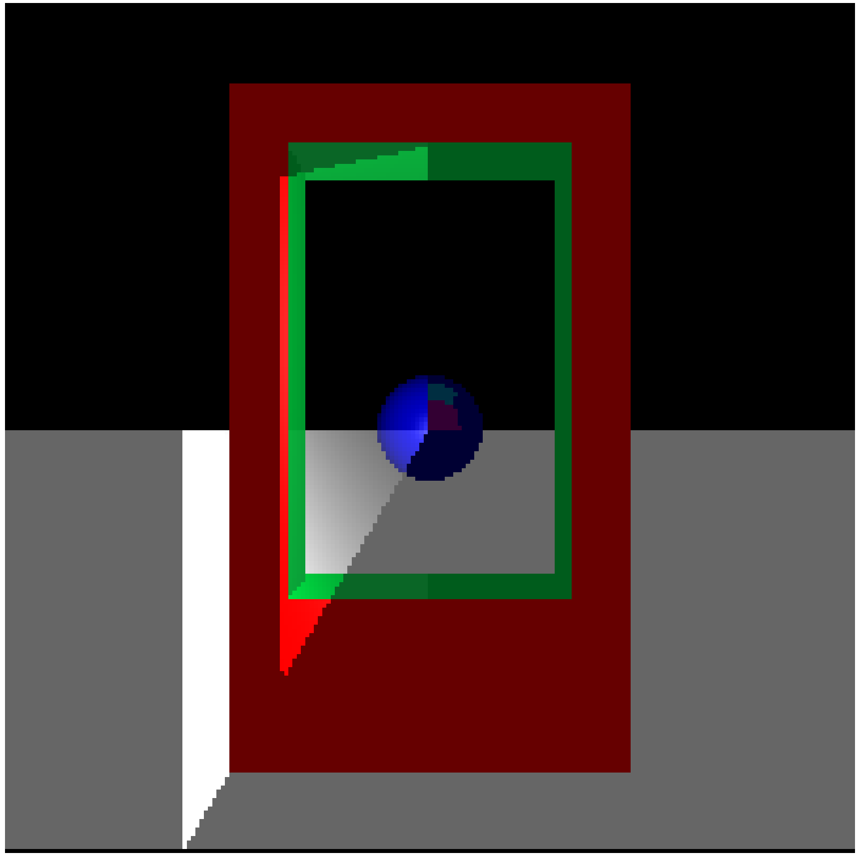
Conclusion

Time took: one week

Rendering: 6 hrs (19375 secs)

Problems encountered:

1. Weird lighting glitches



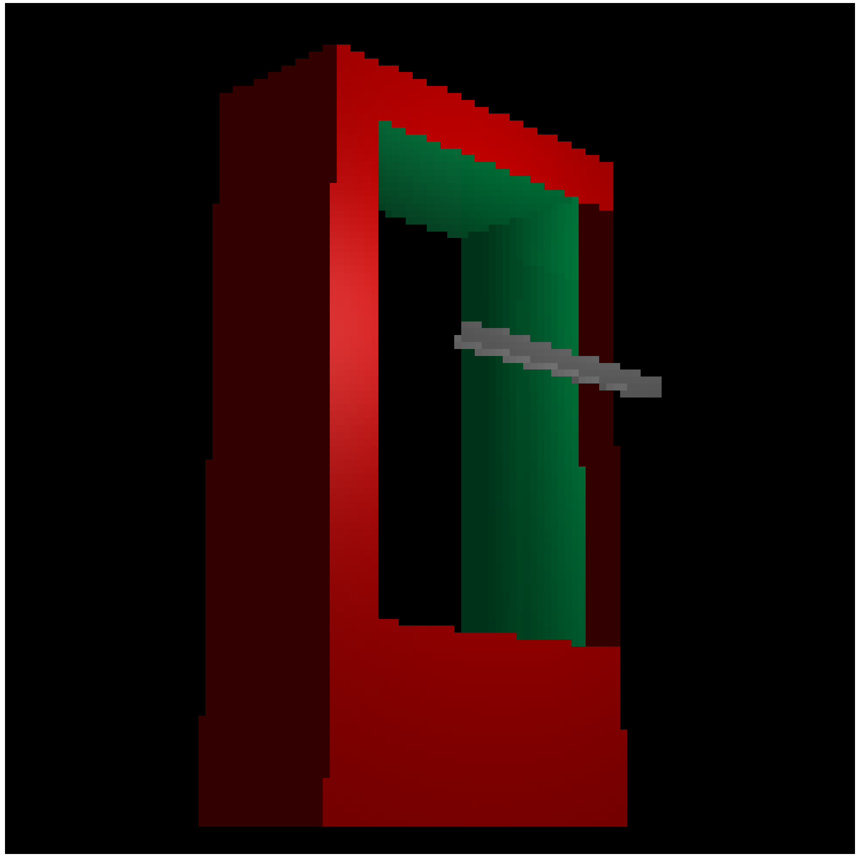
Conclusion

Time took: one week

Rendering: 6 hrs (19375 secs)

Problems encountered:

1. Weird lighting glitches



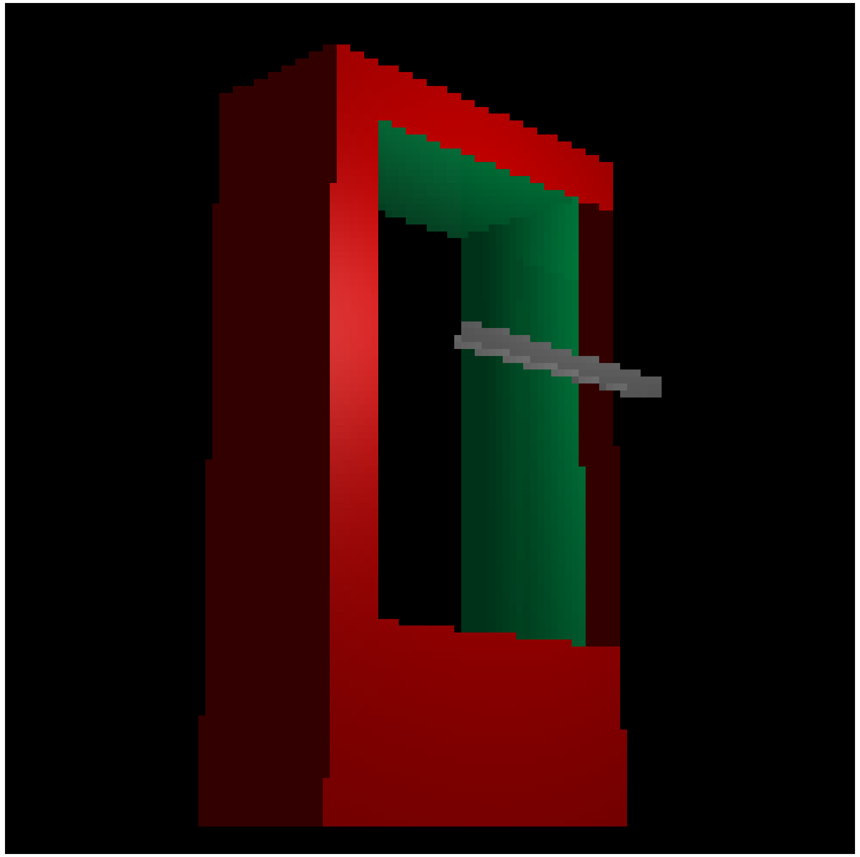
Conclusion

Time took: one week

Rendering: 6 hrs (19375 secs)

Problems encountered:

1. Weird lighting glitches
2. Long simulation time



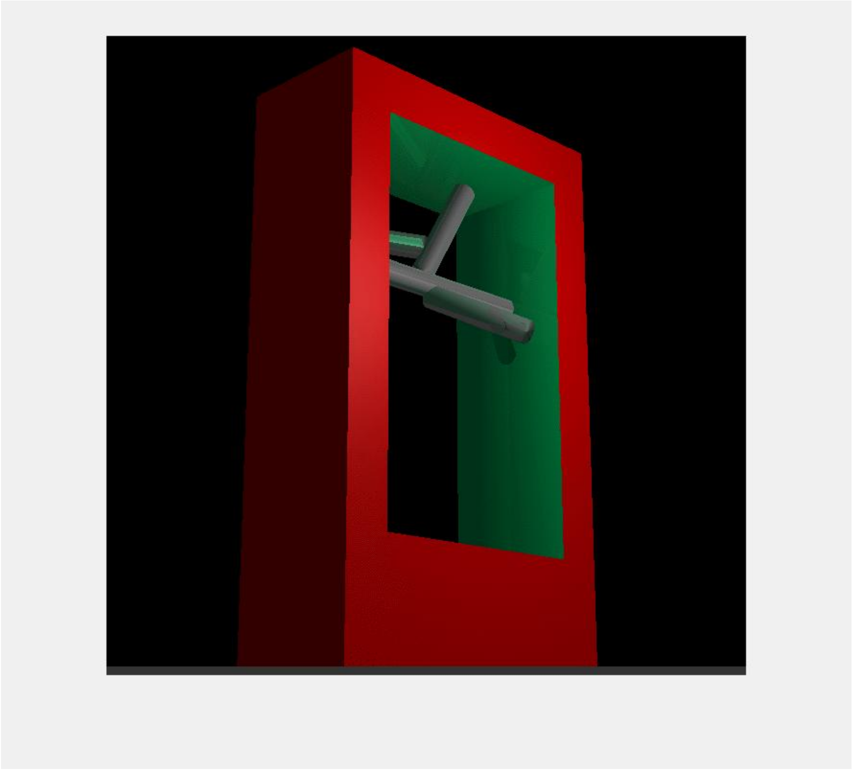
Conclusion

Time took: one week

Rendering: 6 hrs (19375 secs)

Problems encountered:

1. Weird lighting glitches
2. Long simulation time
3. Wrong handedness of coordinates



Conclusion

Time took: one week

Rendering: 6 hrs (19375 secs)

Problems encountered:

1. Weird lighting glitches
2. Long simulation time
3. Wrong handedness of coordinates

Presentation time: 7 min 30 sec