PHYSICAL SIMULATION AND PHOTO-REALISTIC RENDERING ANALYSIS OF A THREE-DIMENSIONAL MARBLE PENDULUM

FINAL PROJECT ASSIGNMENT

Aleyna Etemoğlu
22220030049

Mersin University
Department of Computer Engineering

Dr. Furkan Gözükara, Assistant Professor

Mersin

January - 2026

1.**ABSTRACT**

Within the scope of this project, 3D modeling, material assignment, and animation of a marble-textured pendulum system were performed using Blender 5.0.1 software. The performance and quality differences between Cycles and Eevee rendering engines were analyzed throughout the study. The resulting 240-frame PNG sequence was integrated using video editing techniques to generate the final simulation output.

Keywords: Blender, 3D Modeling, Cycles Render, Animation, PNG Sequence.

İÇİNDEKİLER

	Sayfa
İNSİDE OVER	1
SUMMARY	2
TABLE OF CONTENT	2
2. İNTRODUCTİON	3
3. MATERIALS AND METHODS	3
4. RESULTS AND DISCUSSION	27
5.REFERENCES	28

2. INTRODUCTION

In computer graphics, real-time physics simulations and photorealistic visualization are among the cornerstones of modern engineering applications. In particular, the accurate digital representation of complex physical movements is of critical importance in both academic research and industrial design processes. Within the scope of this project, the goal is to analyze the physics-based simulation of a pendulum's motion—one of the fundamental problems of classical mechanics—within a 3D environment using advanced visualization techniques.

The primary focus of the project is to demonstrate the interaction of light with objects having different material properties, such as marble and metal, in the most realistic way possible using ray-tracing-based rendering engines. The pendulum apparatus was modeled to consist of pipelines, support mechanisms, and marble-textured moving parts. To ensure geometric accuracy during the modeling process, advanced techniques such as "Mirror" and "Subdivision Surface" were utilized to achieve high-polygon surface quality.

In this study, a detailed analysis was conducted on light refraction, shadow sharpness, and material reflections using the Cycles rendering engine within the Blender software. In the methodology section of the project, the impact of hardware acceleration (NVIDIA CUDA) methods and resolution scaling techniques—used to optimize rendering time—on the results is discussed. The final output has been structured as an animation sequence consisting of 240 frames and converted into a smooth video simulation at 60 frames per second.

3. MATERIALS AND METHODS

3.1. Geometric Modeling and Path Design

The pipe assembly forming the outline of the pendulum simulation was designed using Blender's curve and surface modeling tools.

- **Bezier Curve and Tangent Angles:** The design of the path is based on the Bezier curve, a mathematical function. Each control point (vertex) of this curve possesses tangent handles that dictate the direction and degree of the curve's curvature.

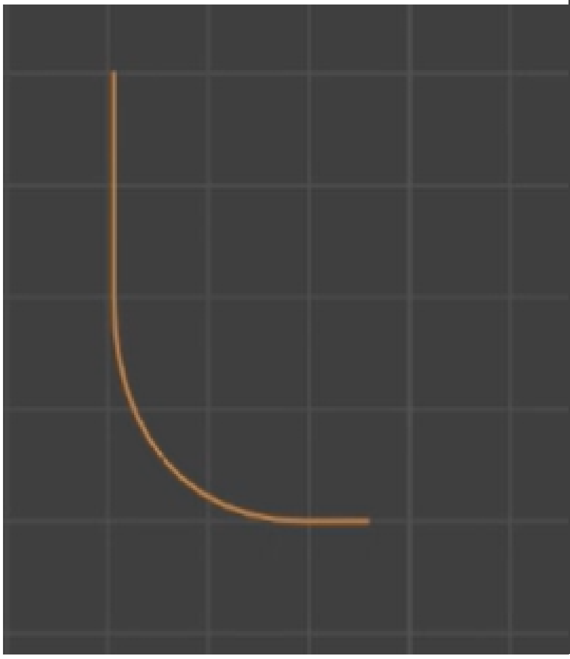
- **Angular Variation and S-Shape:** To achieve the "S" shape of the pipe, the angles between control points were adjusted to create circular transitions of 90 degrees and 180 degrees. While the first curve at the top begins with a positive slope, the slope is zeroed at the midpoint and transformed into a negative angle in the opposite direction at the bottom.

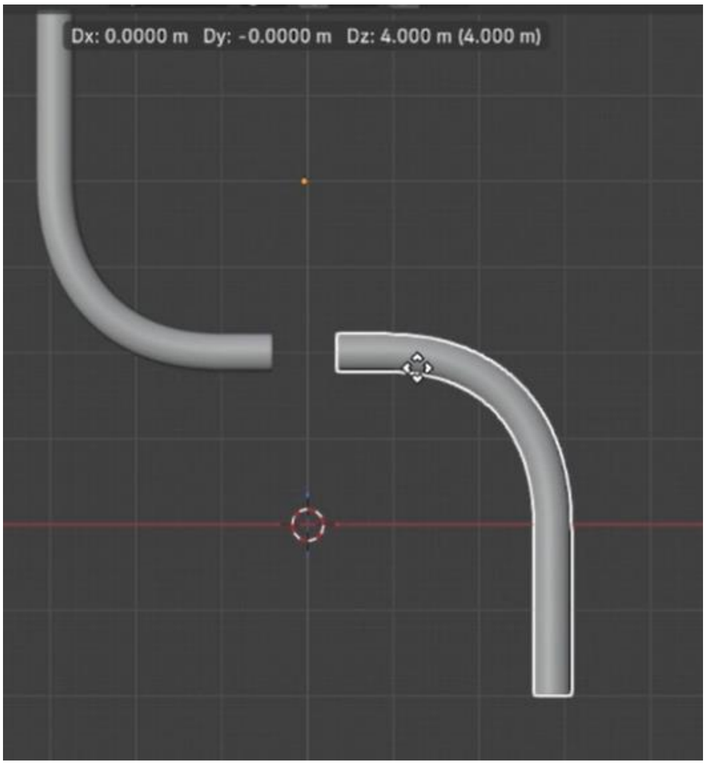
- **Geometric Symmetry:** The vertical lines at both ends of the path are parallel to each other and positioned at a 0 vertical angle. Thus, the central gap through which the pendulum passes is geometrically balanced relative to the exact center point.

- **Mathematics of Beveling (Circular Cross-section):** The thickness of the pipe was achieved through the "Bevel" parameter, which generates a 360 degrees circular cross-section (circle profile) at every point along the path. This process ensures that normal vectors are calculated outward while transforming a 1D line into a 3D volume.

3.2. Isolation of the Pipe Segment and Geometric Duplication

In this stage of the modeling process, the lower half of the initially generated full path was deleted to obtain a single segment, which was then used as a reference to reconstruct the structure.





- **Segment Isolation (Delete/Dissolve):** The control points (vertices) at the lower section of the pipeline were selected in **"Edit Mode"** and removed using the **"Delete" (X)** command. This procedure was performed to create the central gap for the pendulum and to define the fundamental building block (L-type curve) of the model.

- **Volumetric Formation (Object Data Properties):** A cylindrical volume was assigned to the isolated 1D curve through the **"Geometry -> Bevel -> Depth"** parameter. A full 360 degrees circular cross-section was utilized to ensure a smooth surface area for the pipe.

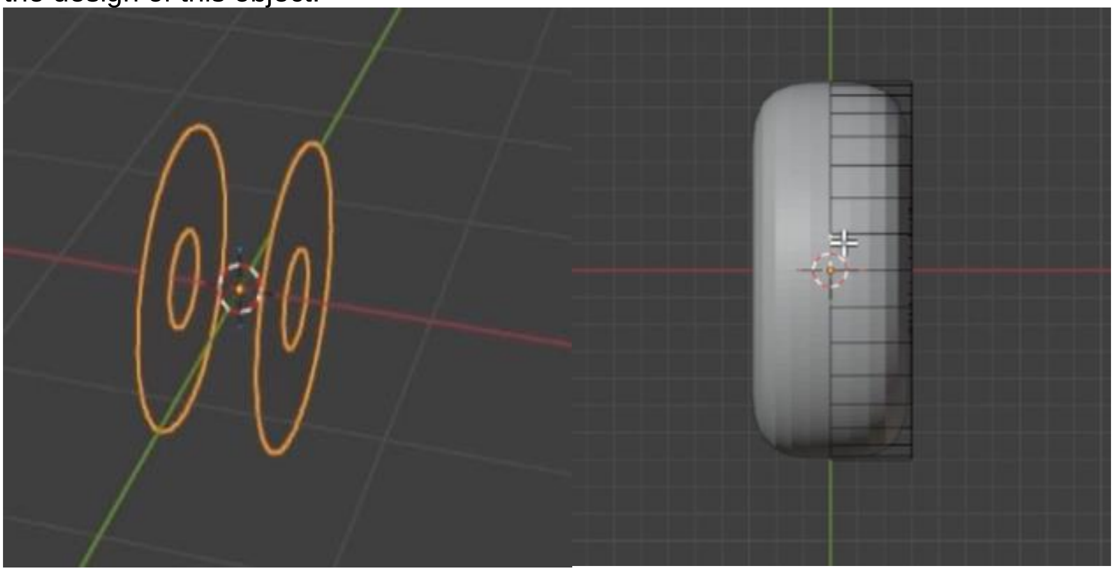
- **Angular Alignment and Reference Point:** The vertical axis of the pipe was positioned at exactly 90 degrees (perpendicular), while the curve on the horizontal axis was terminated at the 0 plane. These precise angles guarantee that the segments align perfectly during duplication.

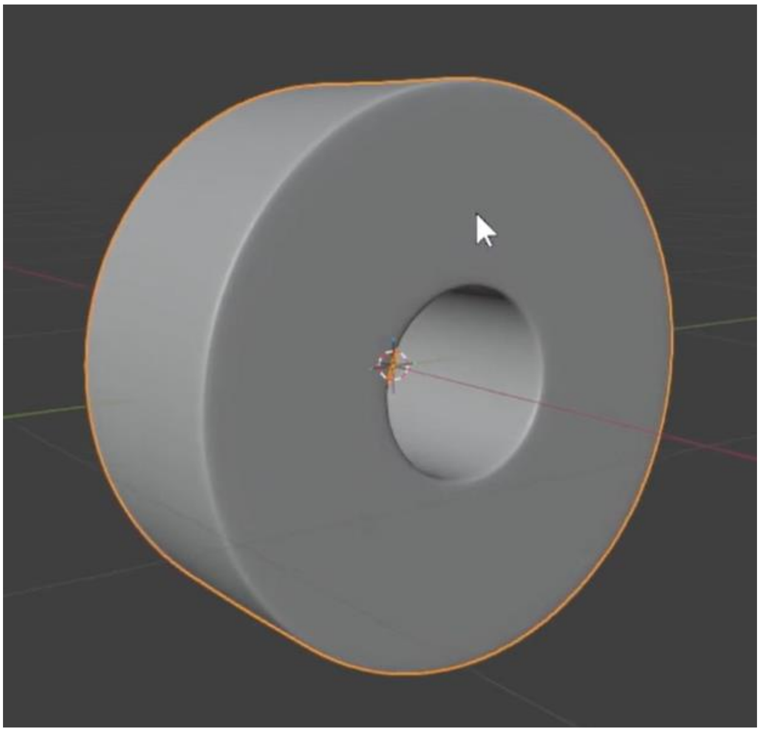
- **Object Duplication:** The existing upper segment was selected and replicated using the **"Shift + D"** command, then shifted downward along the vertical axis to be positioned as its opposite counterpart.

- **Rotational Symmetry:** The duplicated lower segment was rotated 180 degrees relative to the main center point to form the symmetrical equivalent of the original "S" shape. Consequently, a mathematically equal gap was maintained between the upper and lower segments, and the 4.000 m (Dz) oscillation space for the pendulum was adjusted with millimetric precision.

3.3. Modeling Process of the Inner Connection Ring:

The small inner ring, which connects the large marble ring to the main pipeline in the pendulum system, is one of the most critical components ensuring both mechanical connection and motion transmission. Geometric precision and symmetry were prioritized in the design of this object.





- **Creation of the Circle Object:** The modeling process initiated by adding a **"Circle"** object to the scene via the **"Shift + A"** shortcut. The initial vertex count was set to **64** to achieve a smooth circular form.

- **Angular Positioning and Rotation:** The object was rotated **90 degrees on the Y-axis** to be perfectly perpendicular to the plane of the pendulum ring. This angular adjustment ensures the circle sits precisely centered within the slot of the larger marble ring.

- **Mirror Modifier Application:** The **"Mirror Modifier"** was utilized to ensure both sides of the ring are mathematically identical. Only the right half of the object was modeled, while the left half was automatically reflected across the **X-axis**.

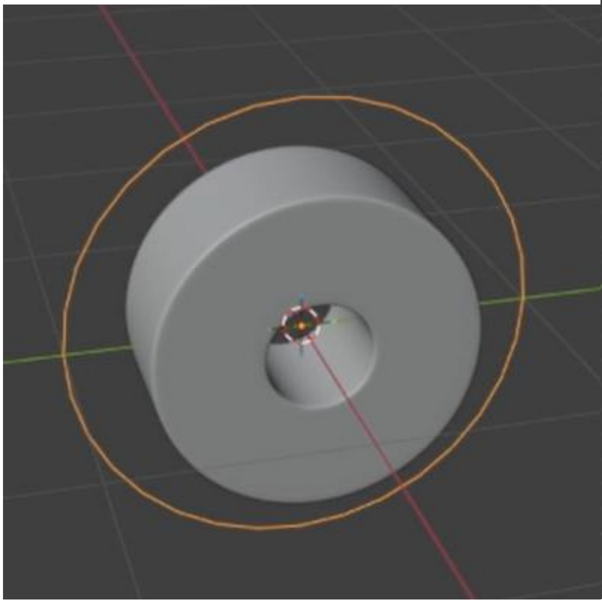
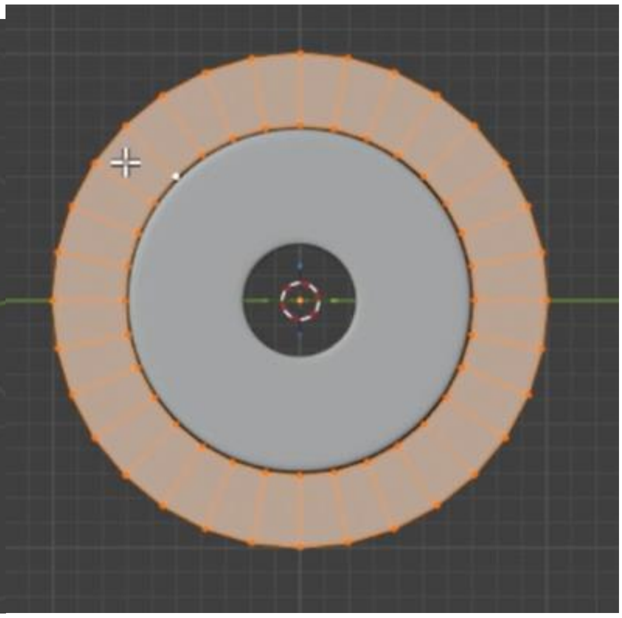
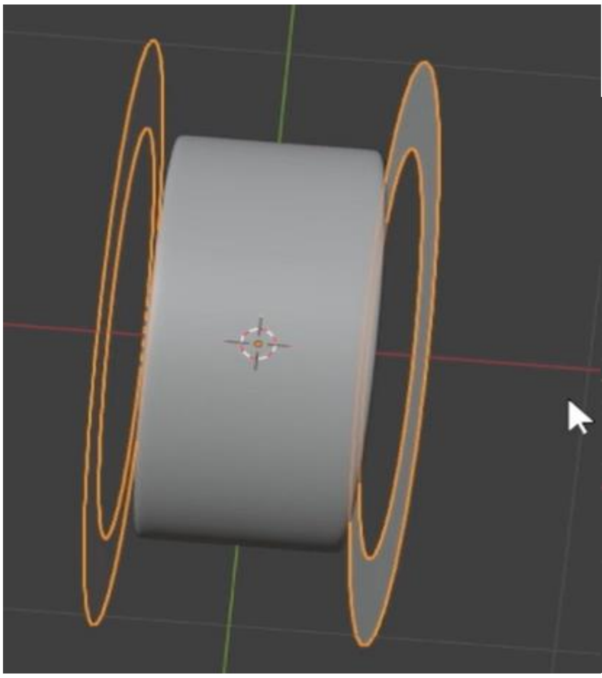
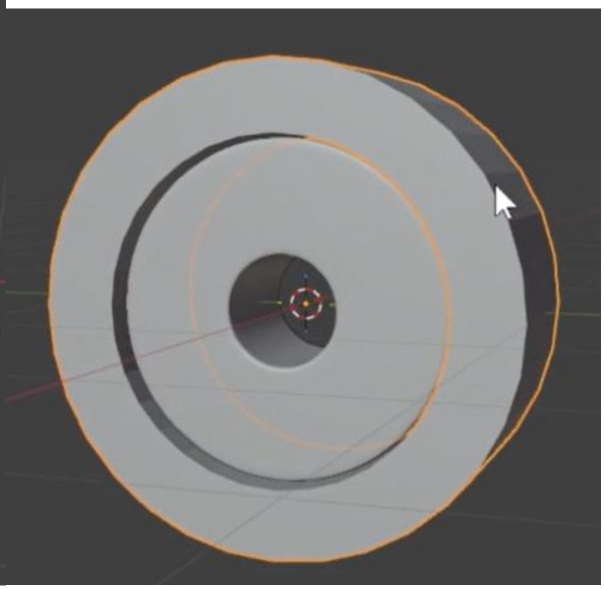
  - **Clipping:** The **"Clipping"** feature was enabled in the mirror settings, allowing the vertices at the center line to snap together, ensuring the object behaves as a single unit.

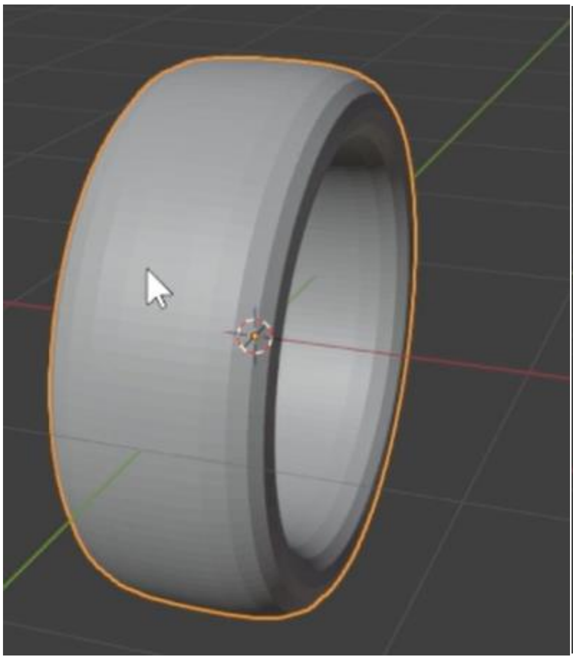
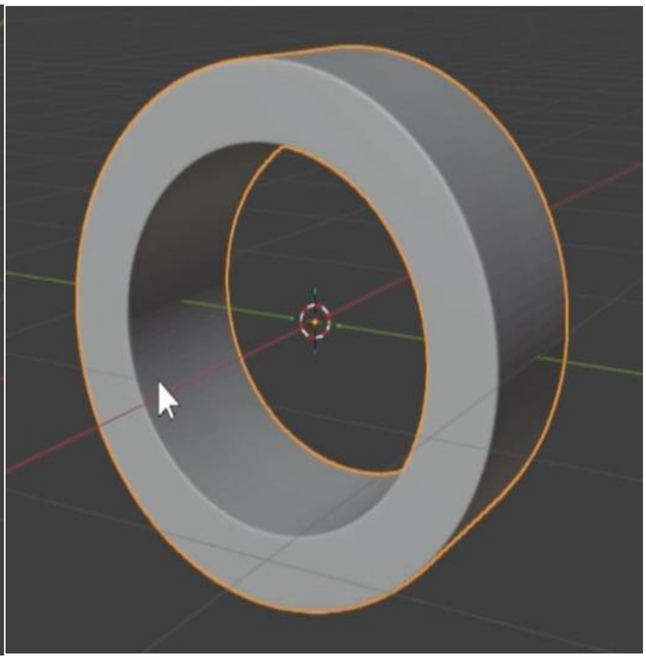
- **Adding Volume (Solidify & Extrude):** The **"Solidify"** modifier was applied to provide depth to the 2D circular structure. The ring's wall thickness was calculated with a clearance of **2 to 5** to prevent friction within the larger ring during oscillation.

- **Surface Refinement (Subdivision & Shade Smooth):** To eliminate the faceted appearance, the **"Subdivision Surface"** level was increased to 2, and the **"Shade Smooth"** command was applied to achieve photorealistic smoothness.

3.4. Modeling Process of the Inner Carrier Ring:

This object is the primary circular mesh structure that passes through the pendulum ring and supports the load of the entire mechanism.

- **Initialization of Base Geometry:** The modeling began by adding a **Circle** mesh via the **Shift + A** menu. To enhance surface smoothness and collision quality during physics simulation, the vertex count was set to **32**.

- **Creating the Ring Form (Inset and Extrude):**

    - In **Edit Mode**, while all edges were selected, the **E (Extrude)** and **S (Scale)** commands were used to generate an inner edge loop.
    - This established a 360° symmetrical circular face area by defining the difference between the inner and outer diameters.

- **Volumetric Depth (Z-Axis):** The 2D ring form was given vertical depth along the **Z-axis** using the **E** command. The thickness was determined with a mathematical tolerance to prevent friction with the main pipeline.

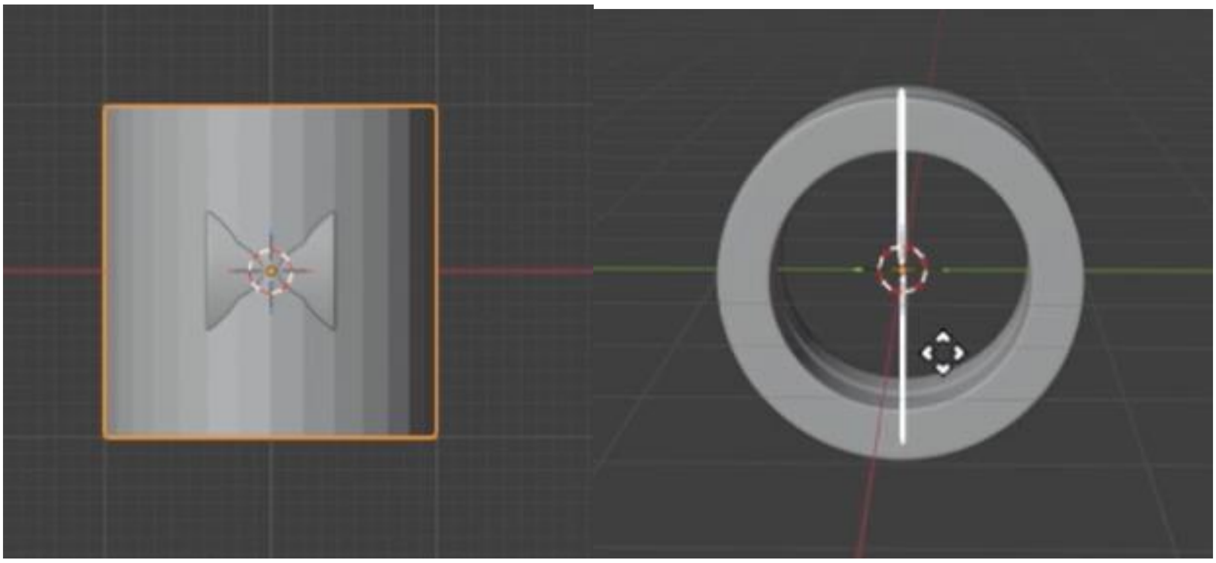
- **Mirror and Symmetry Settings:** To ensure the top and bottom surfaces remained perfectly parallel, a **Mirror Modifier** was utilized along the Z-axis, keeping the **Center of Mass** precisely at the geometric center.

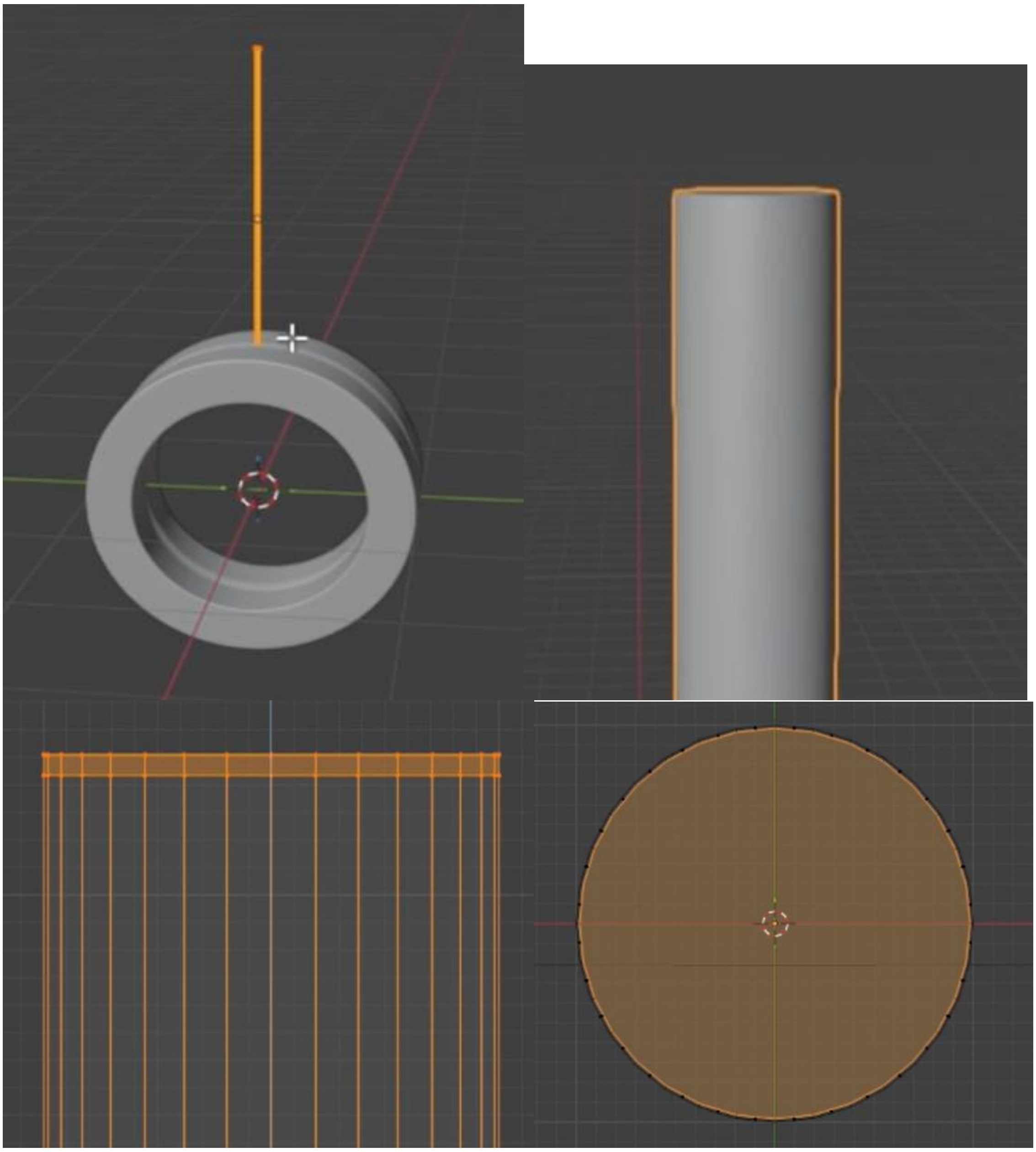
- **Angular Alignment:** While the object was initially on the horizontal plane (0°), it was rotated exactly **90 degrees on the Y-axis (R -> Y -> 90)** to align with the mechanism. This ensures the ring sits perpendicular to the pipe.

- **Surface Optimization (Shade Smooth):** To prevent a low-poly appearance during rendering, **Shade Smooth** was applied in **Object Mode**, and **Auto Smooth (30)** was activated to preserve necessary sharp edges.

3.5. Modeling Process of the Pendulum Support Rod:

This object is the main body that establishes the connection between the carrier ring and the pendulum weight, oscillating along the vertical axis.





- **Creation of the Cylindrical Base:** The modeling began by adding a **Cylinder** object via the **Shift + A** menu. To maintain an elegant and thin profile for the pendulum, the radius of the cylinder was kept at a minimal value.

- **Vertex and Diameter Configuration:** The top and bottom circular faces of the cylinder were structured with **32 vertices** to ensure a smooth cylindrical form during rendering.

- **Scaling and Extension (Z-Axis):** In **Edit Mode**, the top face of the cylinder was selected and extended upward using the **G + Z** (Grab) command. The length of the rod was meticulously adjusted to bridge the $4.000\,\text{m}$ gap between the main pipeline and the pendulum ring.

- **Alignment and Angular Precision:** The rod was fixed at a perfect **90-degree** vertical position using the **R -> Y -> 0** command. This 0° deviation is critical, as any tilt would cause the pendulum to swing unevenly during physics simulation.

- **Detailing with Loop Cuts (R):** Horizontal divisions were added to specific areas of the rod using **Ctrl + R** (Loop Cut). This allows the object to react more realistically in potential stress or flexibility simulations.

- **Pivot Point Adjustment:** The object's origin (pivot point) was moved to the top end. This ensures that the pendulum swings centered from its top attachment point, behaving like a real hanging rod.

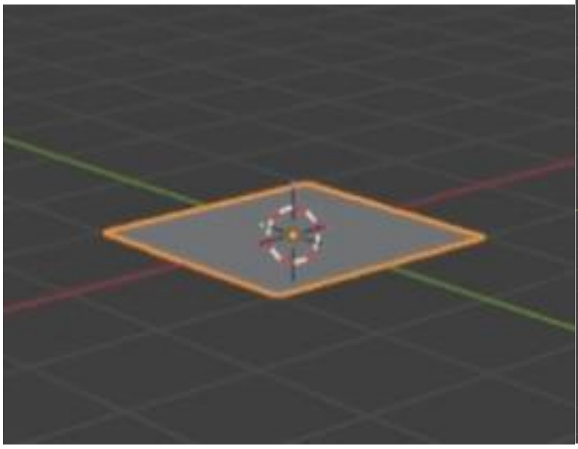
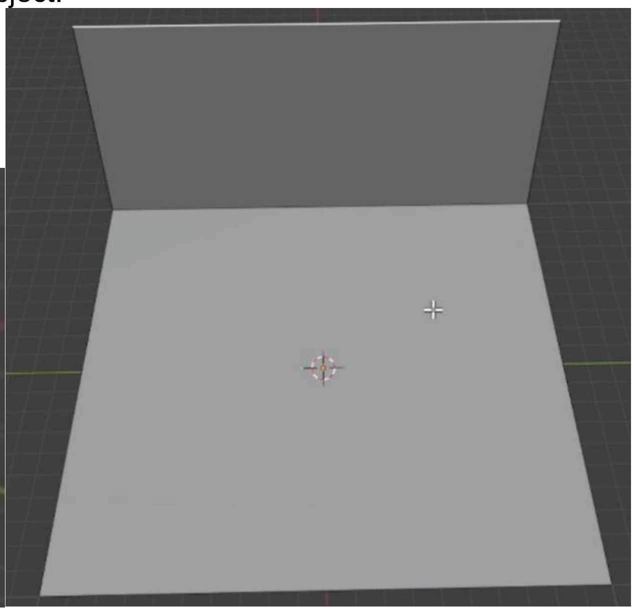
- **Precise Grouping with Box Select (B):** The **Box Select** method was utilized to collectively select the upper faces of the cylinder that needed extension or specific Loop Cut segments. By pressing the **"B"** key, a rectangular selection area was created, allowing for the millimetric selection of only the targeted region (e.g., only the top vertices of the rod) within complex vertex structures.

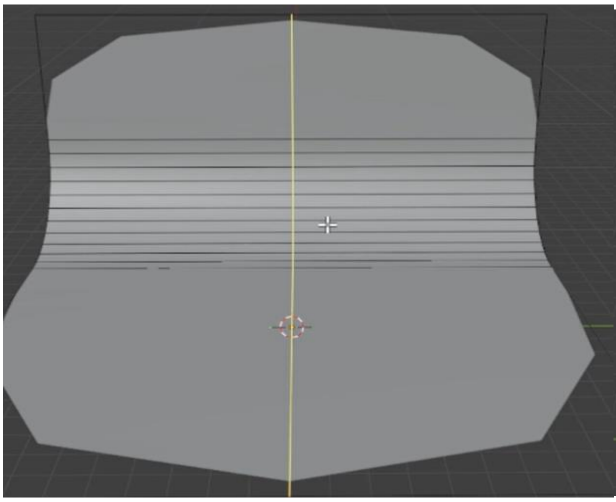
- **Selection Invert (Ctrl + I) Strategy:** The **Invert** command was employed to hide or delete all remaining parts while working on a specific section of the model. For instance, while only the main body of the rod was selected, **Ctrl + I** was used to select all other auxiliary geometries in the scene, enabling rapid processing of areas outside the focused object.

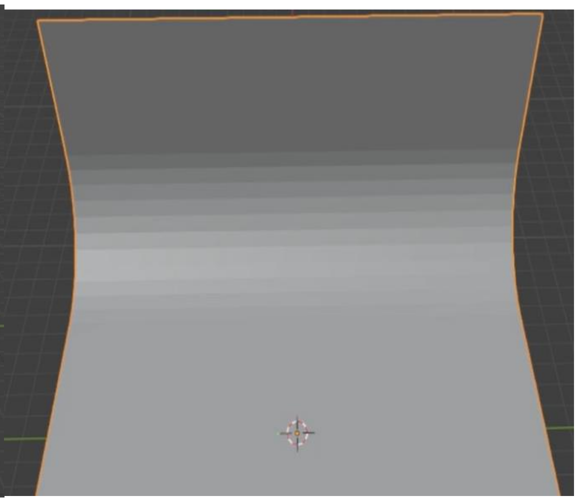
- **Correlation Between Angular Accuracy and Selection:** During box selection, the **Orthographic** view mode was engaged to prevent selection errors caused by depth perception. Consequently, vertices on both sides were manipulated equally, ensuring the rod's vertical axis maintained its perfect **90-degree** perpendicularity.

3.6. Creation of the Floor and Studio Backdrop:

The infinite backdrop, upon which the scene objects are placed and where light reflections occur, was developed from a basic Plane object.





- **Initialization of the Plane Object:** The modeling began by selecting **Mesh -> Plane** from the **Shift + A** menu. Initially, the object was placed on the horizontal plane (with 0) at a standard size.

- **Scaling (S):** The floor was expanded using the **"S"** key to cover the entire oscillation range of the pendulum mechanism. To ensure it did not bleed out of the camera's frame during rendering, the plane was scaled significantly on both axes.

- **Extruding the Backdrop:** To create a studio-type infinite background, the back edge of the plane was selected and extended vertically along the Z-axis using the **"E + Z"** command. This vertical surface was set at a perfect **90 degree perpendicular angle** to the floor.

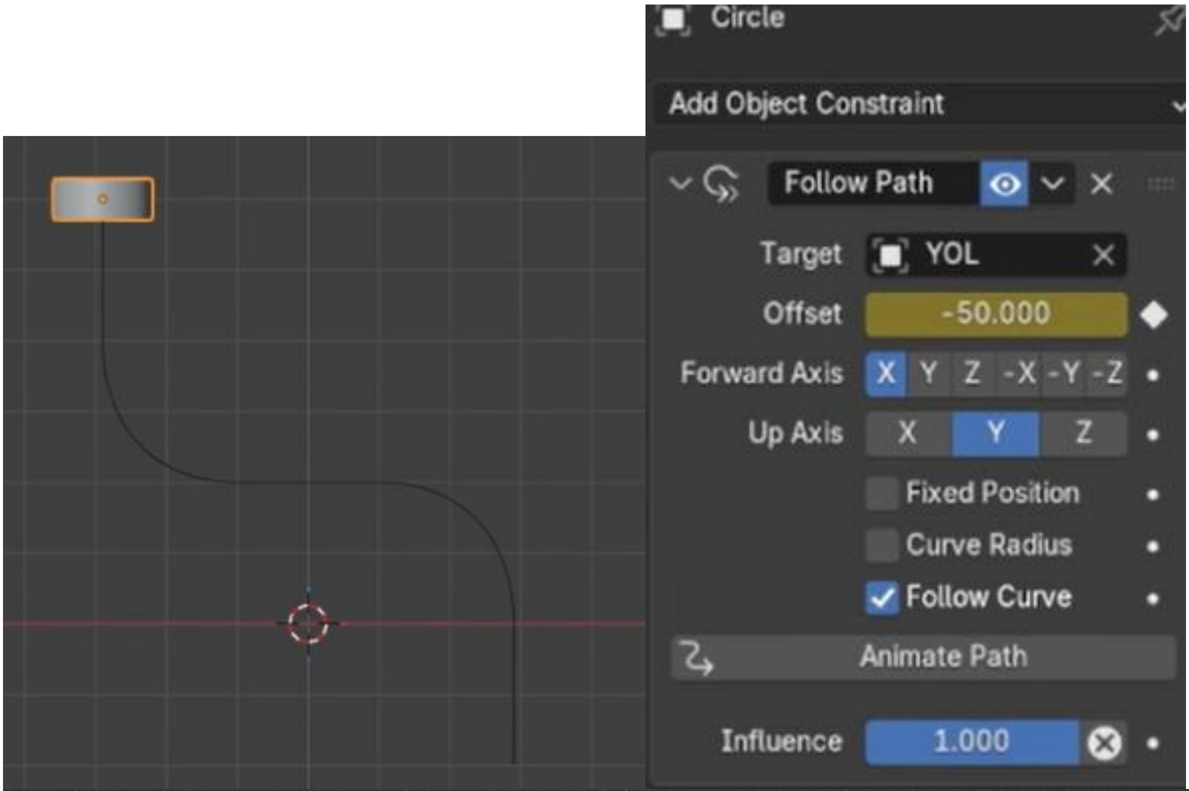
- **Curved Corner Design (Bevel - Ctrl + B):** The **"Bevel"** command was utilized to soften the sharp corner where the floor meets the vertical wall. By adding multiple segments with the mouse wheel, the corner was transformed into a smooth, rounded curve.

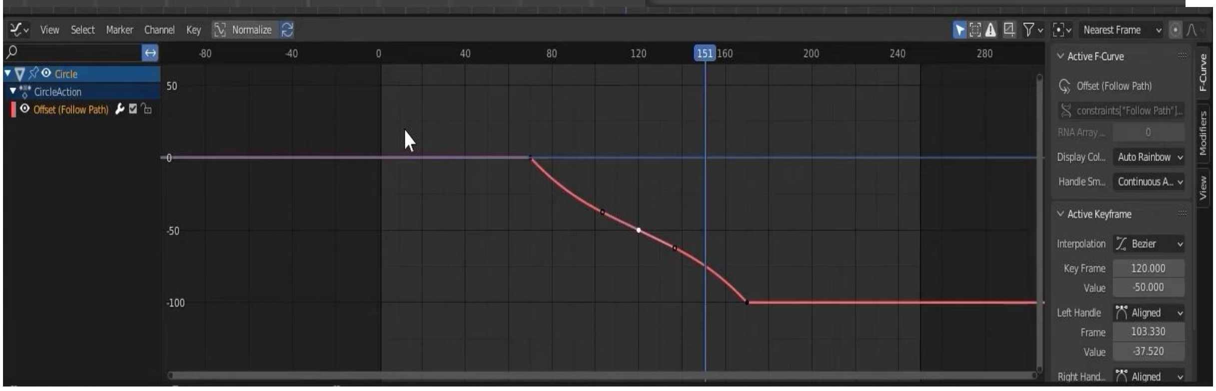
    - **Purpose:** Similar to photo studios, this prevents light from forming harsh shadows at corners, creating a seamless transition.

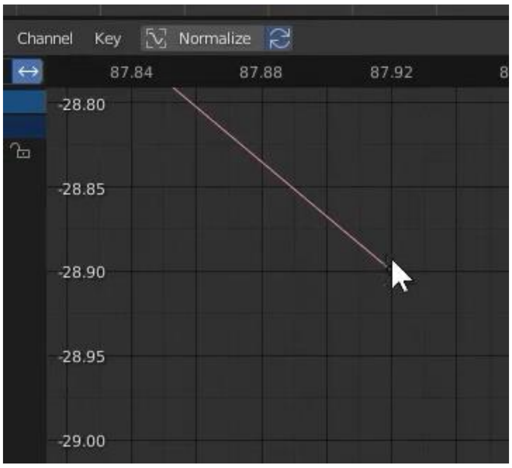
- **Application of Shade Smooth:** To hide the polygon edges on this curved structure, the **"Shade Smooth"** command was applied via a right-click. Consequently, the background appears as a single, smooth paper backdrop during the rendering process.

4.7. Dynamic Motion Mechanism and Path Following Settings:

The movement of the pendulum system along the main pipeline was structured using advanced object constraints and curve-based animation techniques to ensure physical realism, angular precision, and fluid timing (interpolation).







The smoothing value from this point was copied and applied to frame 170.

- **Path Following Mechanism:** The **"Follow Path Constraint"** was utilized to ensure the carrier ring object moves precisely along the pipeline. This command was chosen to lock the object's coordinates to the 360 degree circular path of the pipe, preventing alignment errors.

- **Keyframe Generation (Timeline):** Keyframes were assigned using the **"I"** key for the start (Frame 1) and end (Frame 240) points of the motion. These frames established the total duration required for the pendulum to travel from one end of the pipe to the other.

- **Graph Editor and Velocity Curves (Bezier Interpolation):** To ensure the motion was non-linear and conveyed a sense of physical acceleration, the **"Graph Editor"** panel was employed.

    - **Fast-Slow-Fast Flow: "Bezier"** handles were applied to the start and end points of the curve. Consequently, the pendulum accelerates upon starting, decelerates slightly due to centripetal forces during the curved segment, and accelerates again upon reaching the straight path.

- **Angular and Temporal Calculations:**

    - **Frame Rate:** Since the output was set to **60 FPS**, the 240-frame sequence was mathematically calculated to correspond to a perfect **4-second** physical loop.
    - **Evaluation Time:** The path was treated as a percentage range from 0% to 100%, and the slope in the Graph Editor was steepened for the segments requiring maximum velocity.

- **Continuity Control:** To prevent velocity jumps at the junction of the two video segments (Frames 25 and 26), the curve tangents were aligned in parallel, ensuring a "pinpoint" smooth transition.

- **Target Path Designation:** In the "Follow Path" constraint, the Bezier curve named **"YOL"** was selected as the target. This restricts the pendulum from moving freely in space, forcing it to remain strictly on this rail.

- **Axis Alignment (Forward & Up Axis):** Axis settings were configured to ensure the pendulum progresses along the path without tilting or flipping.

    - **Forward Axis: X** was selected to ensure the front face of the pendulum follows the direction of the path.
    - **Up Axis: Y** was chosen to lock the vertical orientation of the pendulum.

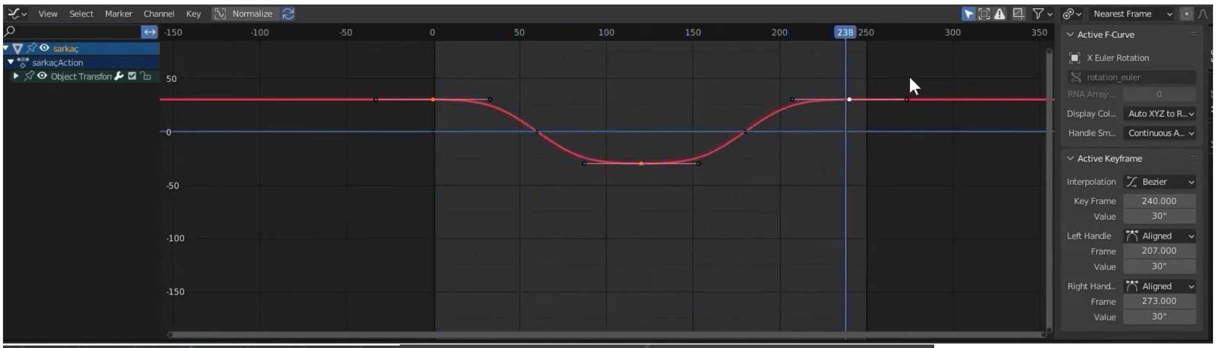
- **Offset Calculation:** The offset value of **-50.000** seen in the interface was used to determine the exact starting point of the pendulum on the path. This value was entered with millimetric precision to center the pendulum within the pipeline.

- **Follow Curve Feature:** By enabling this option (**checked**), the pendulum was made to automatically adjust its rotation relative to the curves of the pipe. Without this, the pendulum would clip through the pipe at turns.

- **Animate Path Command:** By executing this command, Blender was instructed to map the 0-to-100 path values onto an automated timeline. This ensures that the frames on the Timeline are perfectly synchronized with the physical motion.

3.8. Oscillation Animation and Periodic Motion Calculations:

The pendulum oscillation was managed via a helper object (Empty) rather than directly on the objects themselves, thereby establishing a physical hierarchy.



- **Utilizing Empty Objects and Parent-Child Hierarchy:** A **"Plain Axis Empty"** was added to the scene to facilitate animation control. All components of the pendulum (the rod and rings) were linked to this Empty object using the **"Parent" (Ctrl + P)** command.
    - **Reason:** This method was chosen to swing the entire mechanism from a single **Pivot Point** without distorting the internal geometric structure of the pendulum.
- **Rotational Keyframes:** To initiate the oscillation, the carrier rod was selected and specific angles were assigned to the **X-axis**.

- **Maximum Amplitude:** Values of **30°** (positive) and **-30°** (negative) were set for the peak points of the swing. This angle was chosen to balance visual aesthetics with physical plausibility.

• **Timing and Periodicity:** Within the 240-frame duration, keyframes were placed at equal intervals to allow the pendulum to complete a full oscillation cycle (swing right-return left-center).

• **Graph Editor and Bezier Interpolation:** To prevent linear (robotic) motion, **"Bezier"** curves were utilized in the **Graph Editor**.

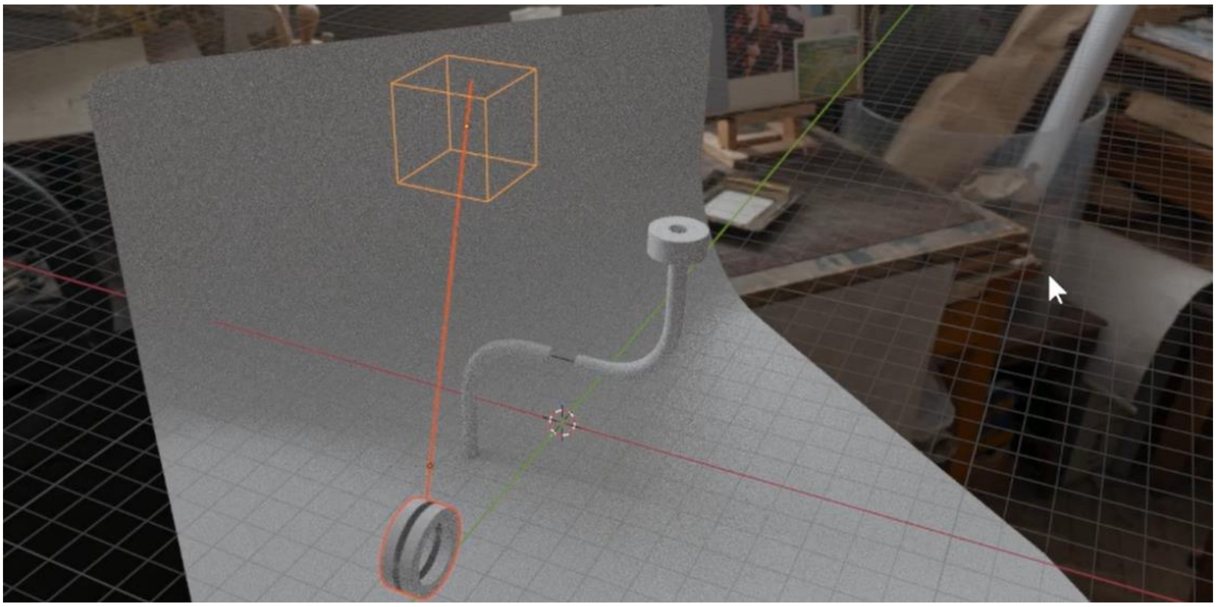
- **Deceleration and Acceleration:** At the peaks of the curve (handle points), the pendulum's velocity approaches zero (decelerates), reaching maximum speed as it passes through the midpoint. This reflects the actual physical behavior of a pendulum under gravity.

• **Utilizing Euler Rotation:** To avoid complex gimbal lock or orientation issues, the **"X Euler Rotation"** mode was preferred. This ensures the oscillation remains strictly on a single axis.

• **Aligned Handles:** The handles in the Graph Editor were set to **"Aligned"** mode, ensuring that when the swing direction reverses, the motion remains fluid without any "snapping" or jagged transitions.

3.9. Environmental Lighting and Background Optimization:

To ensure realistic light refractions and provide flexibility during the post-production stage, HDRI lighting and transparent rendering techniques were implemented.

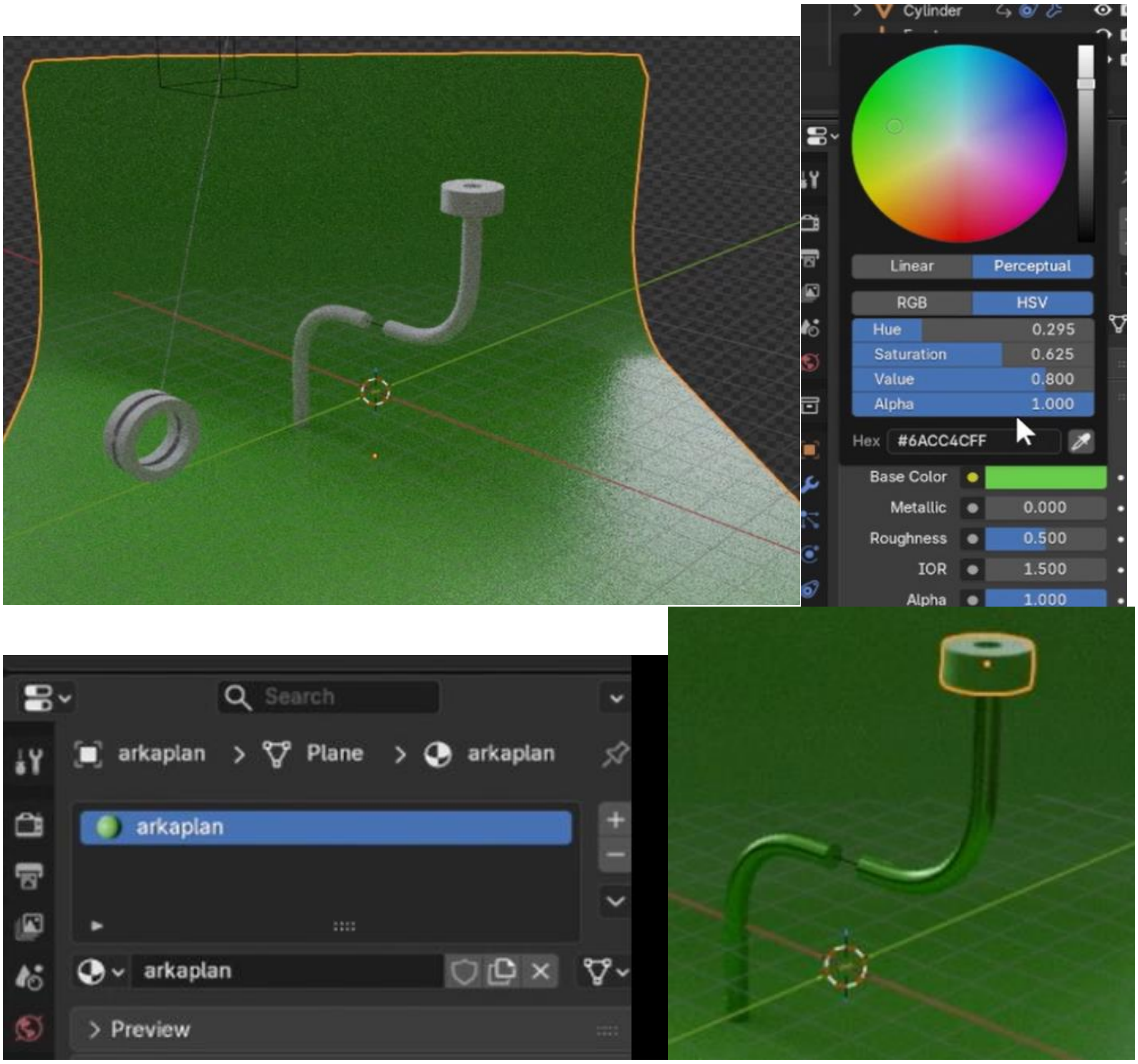


- **Poly Haven and HDRI Utilization:** To provide the scene with natural lighting and realistic reflections on marble surfaces, a high-dynamic-range image (HDRI) was downloaded from the **Poly Haven** library. This file was assigned as an **"Environment Texture"** under the **"World Properties"** tab.

- **Lighting Strategy:** Unlike using a standard lamp, HDRI provides 360-degree color and light data to every angle of the object. This ensured that the highlights on the marble rings were perfectly synchronized with the environment.
- **Transparent Background Configuration:** To prevent the HDRI image from appearing in the background and to isolate the objects from the backdrop during rendering, the **"Transparency"** feature was enabled.
    - **Command:** The **"Transparent"** checkbox was ticked under **Render Properties -> Film**.
- **Editing Advantage:** This process allowed the video background to be recorded as empty (transparent). Consequently, during the Video Editing stage, the desired studio backdrop or alternative colors could be easily placed behind the pendulum.

3.10. Material Design and Coloring Process:

To achieve a photorealistic appearance for the objects in the project, marble textures and metallic surfaces were designed using Blender's "Principled BSDF" based shader structure.



- **Background and Environmental Coloring:** To establish the overall atmosphere of the scene, a new material named **"arkaplan"** (background) was created. This material, assigned to the ground plane, was set to a vibrant green tone to complement the reflections on the marble rings.

- **Creation of Marble Texture:** The marble texture for the pendulum rings was generated using **"Noise Texture"** and **"Color Ramp"** nodes.

  - **Color Selection:** A light gray/white base color with dark gray and slight greenish veins was chosen to achieve a natural stone aesthetic.
  - **Roughness:** To simulate a polished surface, the **Roughness** value was decreased to between **0.1** and **0.2**. This low value provides a "wet" or "polished" sheen to the surface.

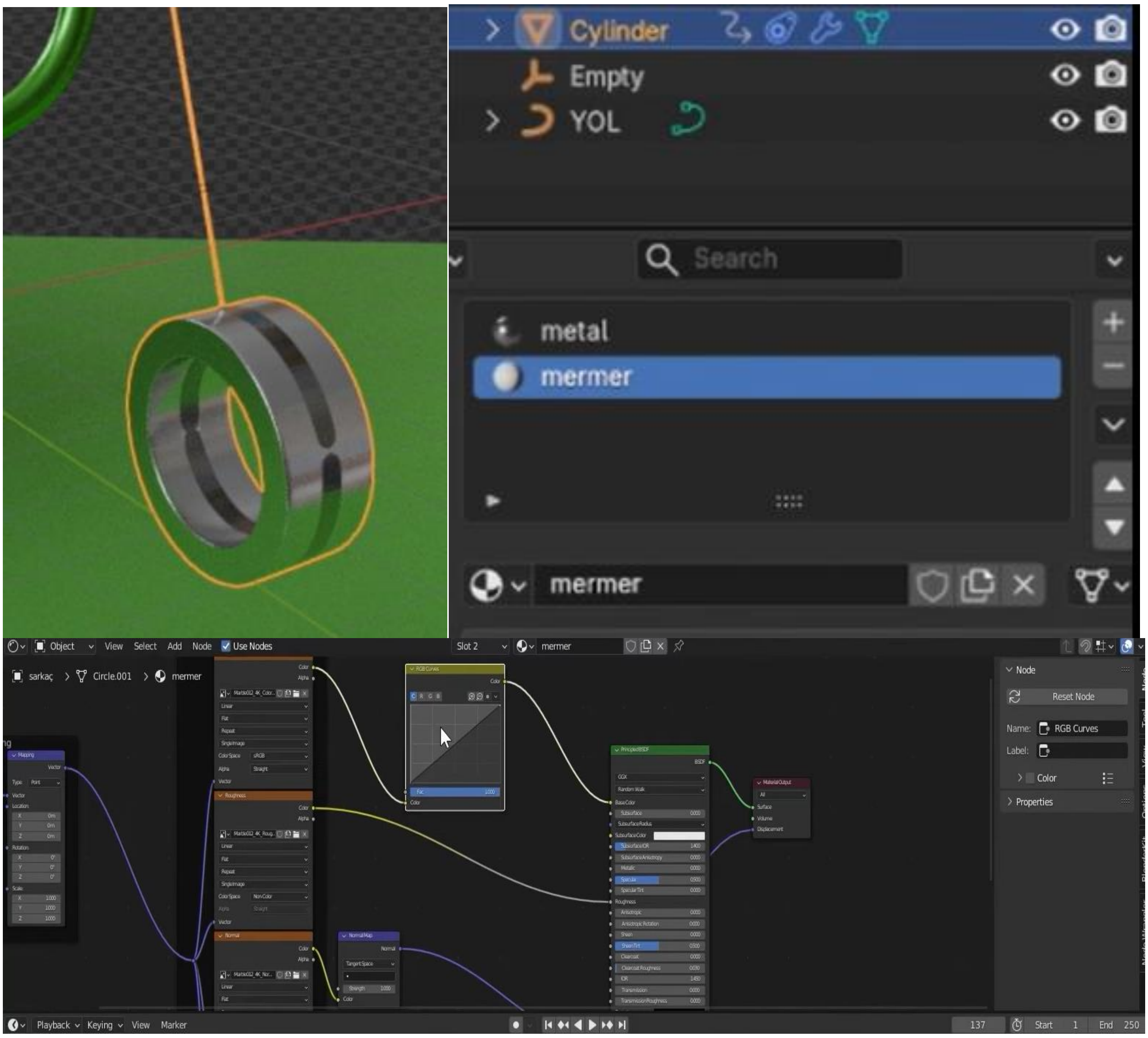
- **Metallic Surface Configurations:** A metallic material was preferred for the carrier rod and pipeline connections.

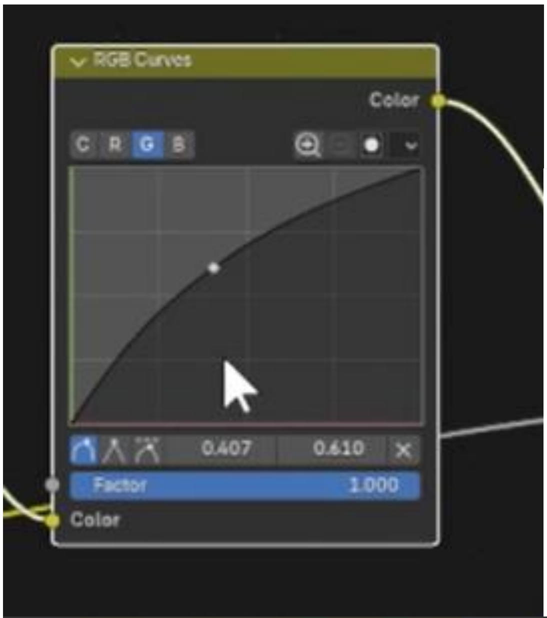
- **Metallic Value:** The **Metallic** parameter was set to **1.0** (full metallic) to ensure the surface behaves like metal.
- **Angular Reflections (Fresnel):** To handle reflections based on the viewing angle, the **IOR (Index of Refraction)** was set to **1.52** for marble and kept at higher values for metallic components.

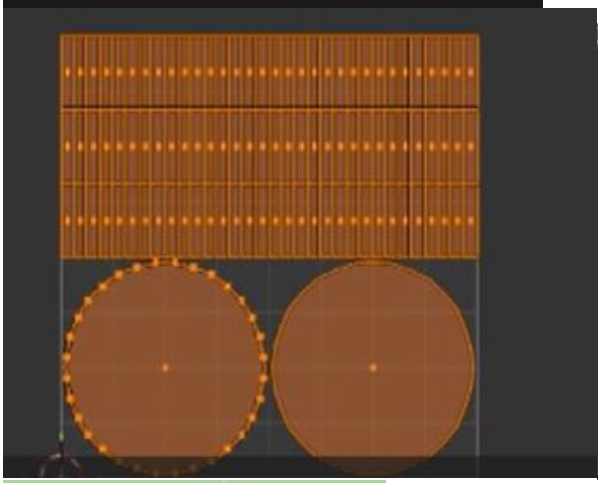
- **Color Integration with Lighting:** The HDRI lighting from Poly Haven enhanced the saturation and realism of the material colors. Bounce light from the green floor provided subtle green tints to the underside of the marble rings, improving scene cohesion.

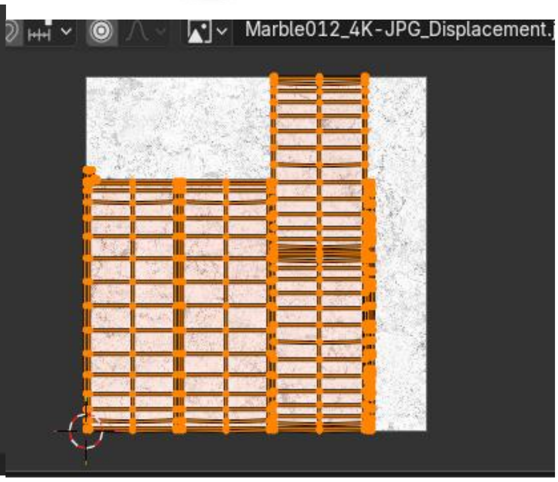
3.10.1. Marble Texturing of the Pendulum Cylinder:

To provide the pendulum ring with a realistic marble appearance, a PBR (Physically Based Rendering) material workflow and precise UV projection techniques were utilized.









- **Texture Sourcing (ambientCG):** To ensure photorealism, a high-resolution (4K) marble texture set was downloaded from the **ambientCG** platform. This set includes maps for Base Color, Roughness, and surface detail via Normal Maps.

- **UV Unwrapping:** To ensure the texture sits perfectly on the cylindrical and vertical faces of the ring, all faces were selected in **Edit Mode**. Subsequently, the **"Smart UV Project"** command was executed by pressing the **"U"** key.

- **Purpose of Smart UV Project:** This projection method was chosen to prevent texture stretching on the complex geometry of the ring and to ensure that marble veins appear seamless across a 360 view.

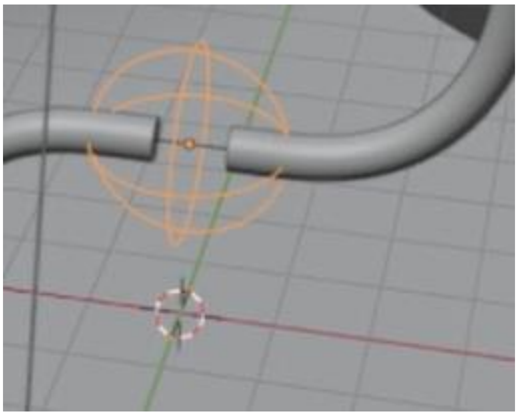
- **Node Editor Configuration:** The downloaded images were linked to the **Principled BSDF** node within Blender's **Shading** panel.

    - **Mapping:** Scale values were adjusted via the **Mapping** node to control the density and distribution of the marble veins.
    - **RGB Curves:** To achieve color harmony with the green background, an **RGB Curves** node was utilized to fine-tune the color tones and contrast of the marble.

- **Material Assignment:** The finalized "mermer" material was added to the object's material slot and assigned to the pendulum ring. The final result is a surface that reflects light like natural stone, featuring smooth transitions between veins.

3.11. Three-Point Lighting and Light Source Configuration:

To accentuate the details of the marble and metallic surfaces and to create a sense of depth, two primary light sources were integrated into the system in addition to the HDRI environmental lighting.



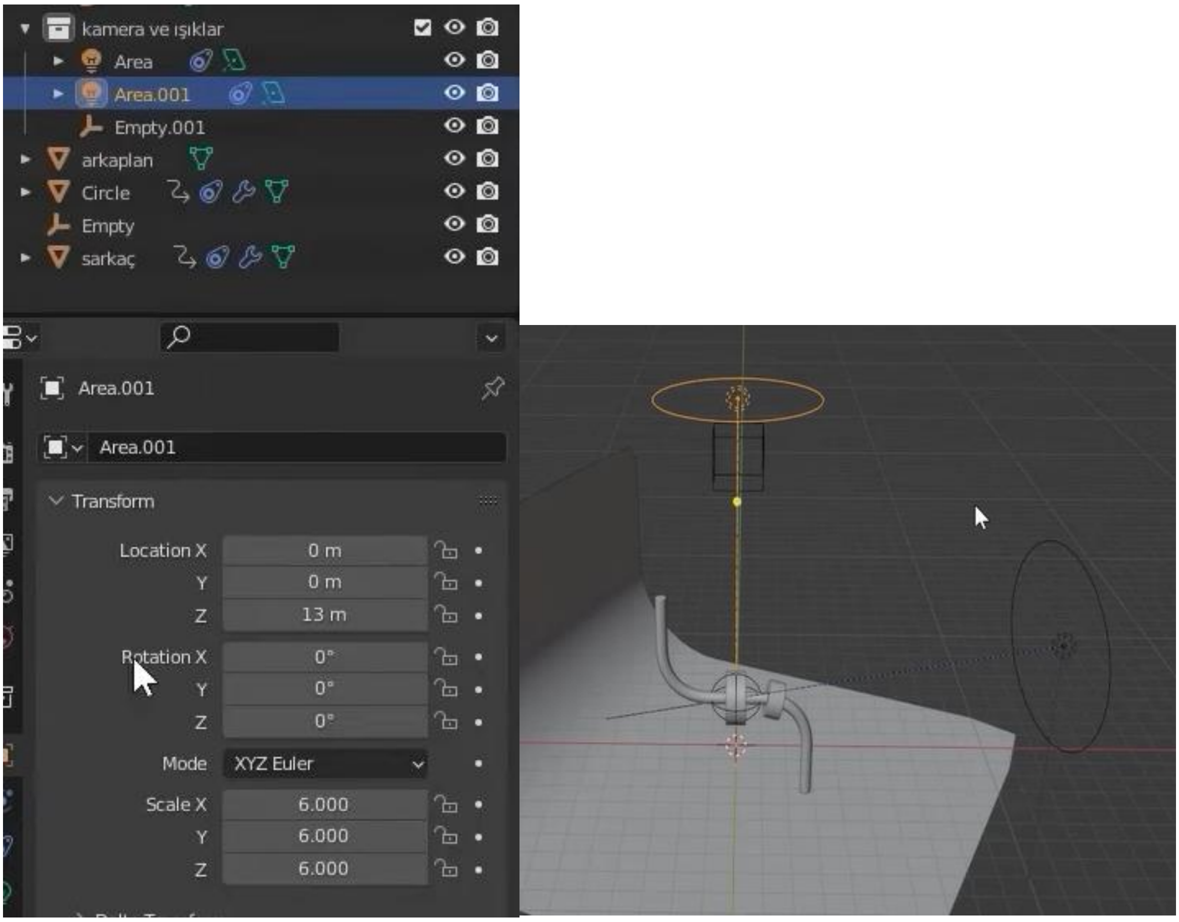


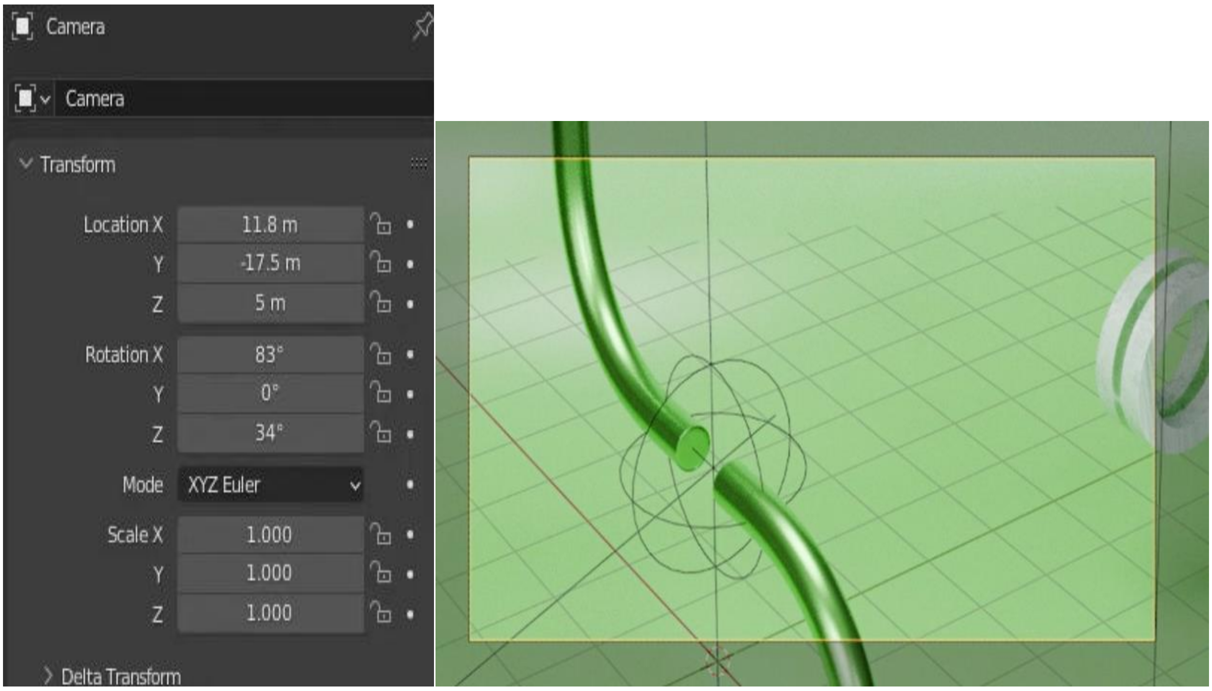
Figure 4.11: Two-point Area Light placement and lighting hierarchy

- **Primary Light Source (Key Light):** An **"Area Light"** was positioned at an upper diagonal point to provide general illumination and define the main contours of the objects.
    - **Position and Angle:** The light was angled at approximately **45** degree from above, facing the pendulum mechanism. This angle was selected to generate soft shadows on the marble rings, thereby enhancing depth.
- **Secondary Light Source (Fill/Rim Light):** A second light source was placed at the rear/side of the pendulum to separate the object from the green backdrop.
    - **Purpose:** To create a subtle highlight on the edges of the marble rings, preventing the object from blending into the transparent background and highlighting engineering details.
- **Power and Color Configuration:**
    - The light intensity (Power) was kept at a balanced Wattage to prevent bounce light from the green floor from overexposing the marble texture.
    - A slightly warm white tone was chosen for the light color, maintaining harmony with the natural daylight provided by the HDRI.

• **Shadow Softening (Size):** The **"Size"** parameter of the Area Lights was increased to ensure shadows transitioned smoothly and softly, mimicking real-world physics rather than appearing unnaturally sharp.

3.12. Camera Positioning and Composition Techniques:

To present the pendulum simulation to the viewer in the most impactful manner, the camera angle, perspective, and focal length settings were configured with engineering precision.



• **Camera Object Initialization:** A **"Camera"** object was added to the scene via the **"Shift + A"** menu. The "Perspective" mode was preferred to allow the viewer to observe the pendulum motion with both depth and a comprehensive flow.

• **3D Positioning (Location):** The camera's position was fixed at the following coordinates to keep the pendulum mechanism perfectly centered:

- **X: 11.8 m / Y: -17.5 m / Z: 5 m**
- These values were determined to prevent the pendulum from swinging out of the frame and to maintain a composition following the rule of thirds.

• **Angular Orientation (Rotation):** The camera's viewing angle was configured in **XYZ Euler** mode with the following angles to provide a slight top-down (bird's eye) effect:

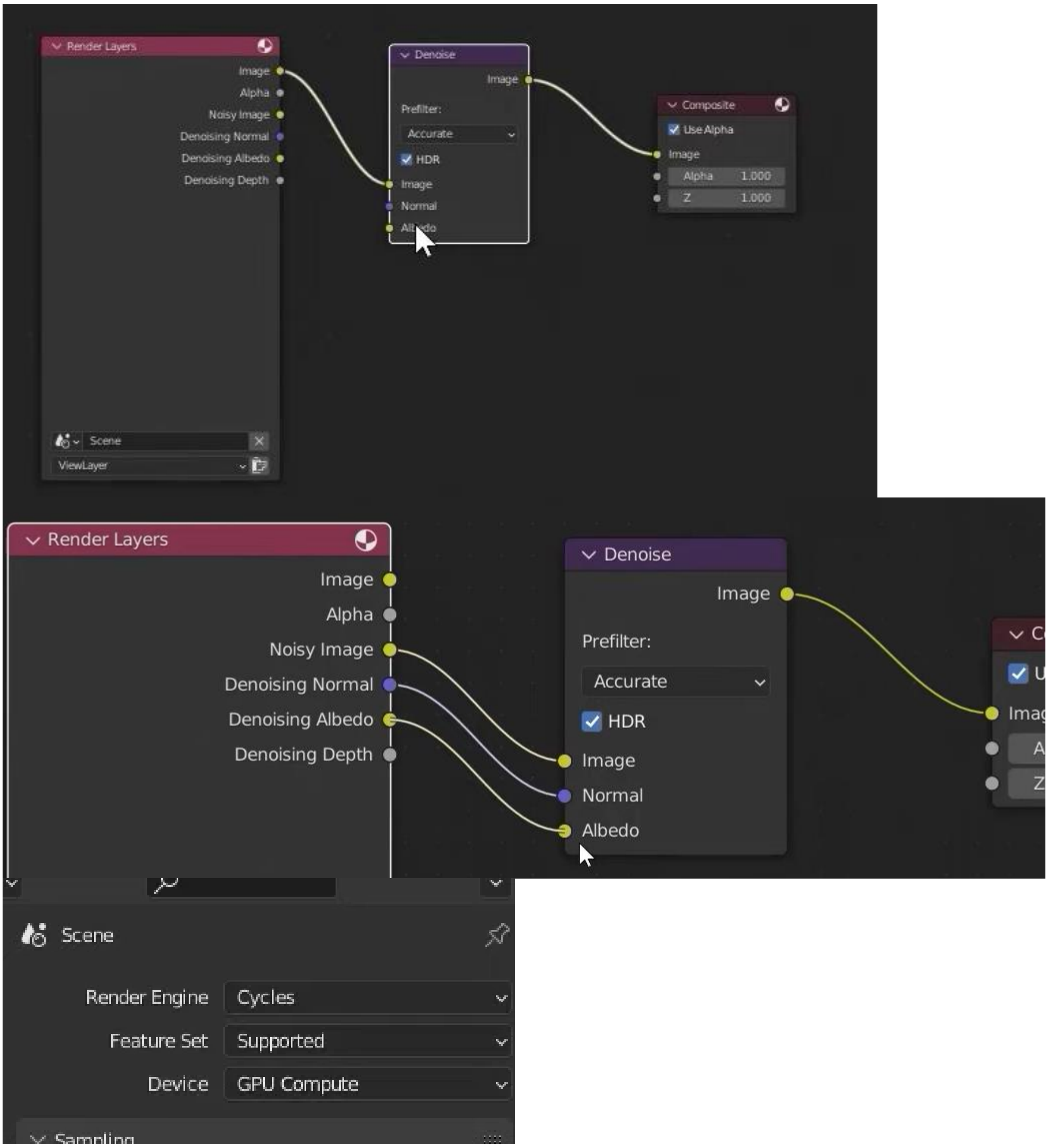
- **Rotation X: 83°:** Tilts the camera slightly toward the floor to emphasize the depth of the marble pipes.
- **Rotation Z: 34°:** Provides a diagonal perspective of the mechanism to make the 3D volume of the simulation more prominent.

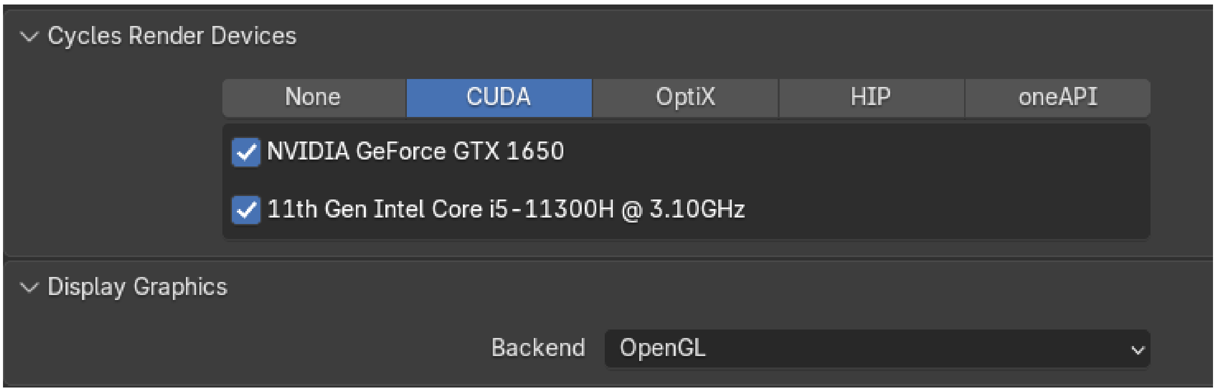
- **Framing and Safe Area:** The **"Numpad 0"** shortcut was used to toggle the camera view, and the **"View -> Lock Camera to View"** feature was enabled to fine-tune the frame.

- **Focal Length and Depth:** A standard wide-angle value was utilized to ensure the curved structure of the pipeline remained undistorted, keeping both the pendulum ring and the studio backdrop in clear focus.

3.13. Render Engine Configuration and Denoising Workflow:

To ensure the final output of the simulation is high-quality, smooth, and photorealistic, Blender's advanced rendering and compositor tools were utilized.





- **Cycles Render Engine and Hardware Acceleration:** The **"Cycles"** engine, which performs physics-based light calculations, was preferred for the rendering process. To optimize processing time, **CUDA/OptiX** hardware acceleration was enabled via the **GTX 1650** GPU.

- **Sampling Strategy:** A high sampling value was entered to prevent loss of detail and to sharpen the marble veins. However, as this increases rendering time, it was balanced using "Denoising" techniques.

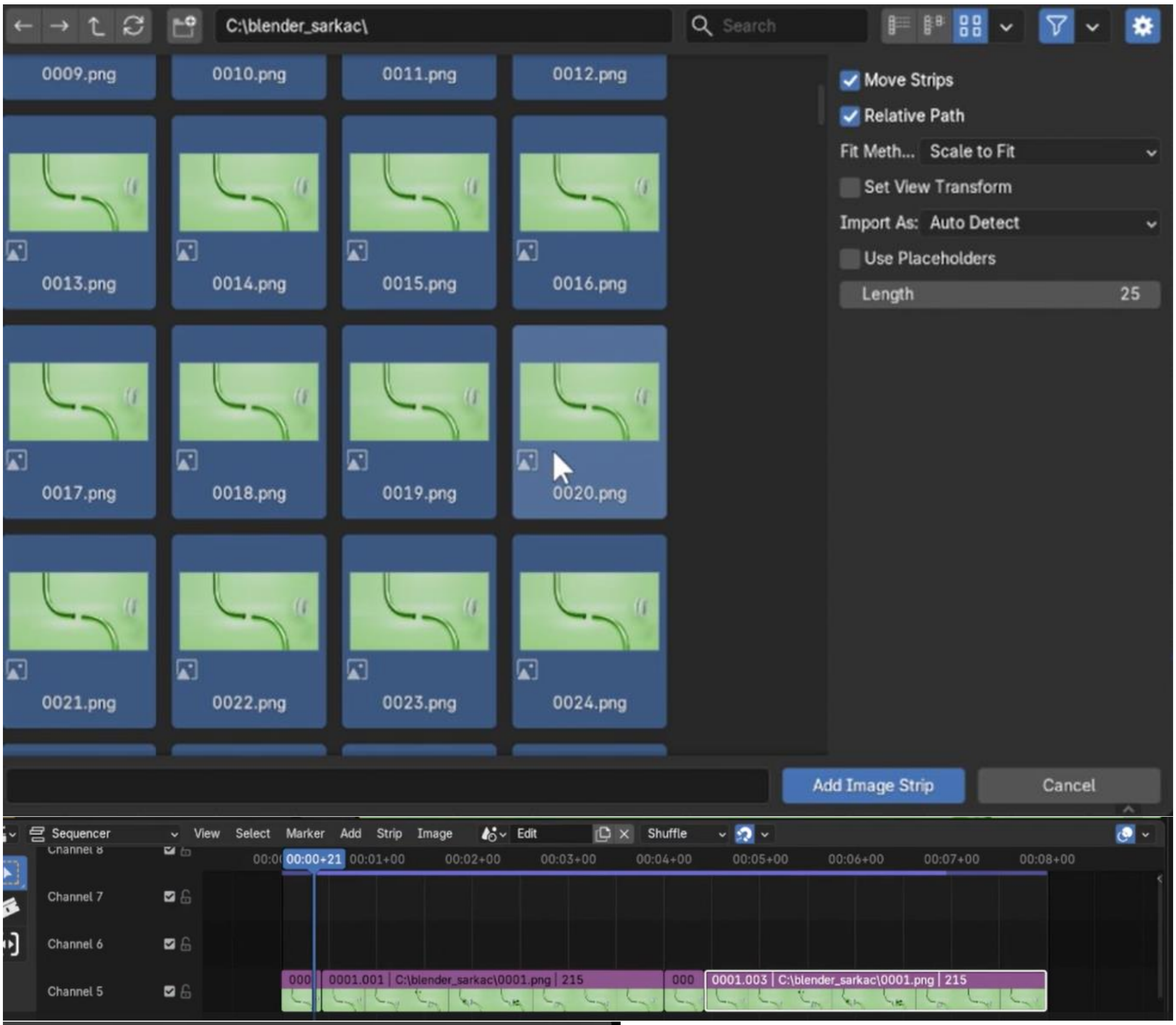
- **Compositing and Denoise Node:** Blender's **"Compositor"** panel was utilized to eliminate post-render graininess (noise).

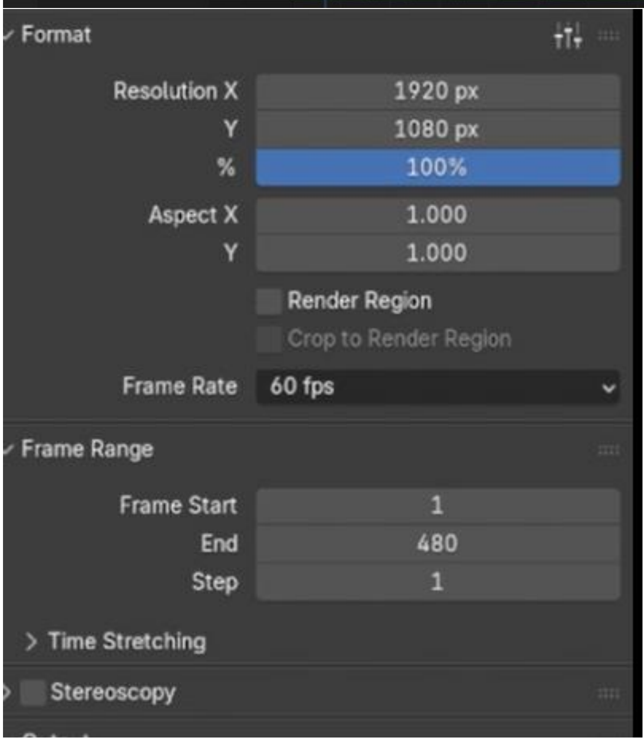
    - **Render Layers Node:** During rendering, data for the *Noisy Image*, *Denoising Normal*, and *Denoising Albedo* were separated for processing.
    - **Denoise Node (Accurate Mode):** This node compares the noisy image with albedo and normal maps to erase noise while preserving sharp details. The **"HDR"** option was enabled to ensure color information in high-light areas remained intact.

- **Resolution and FPS Calculations:** To achieve a smooth video, the output was set to **1920x1080 (Full HD)** resolution at a frame rate of **60 FPS**. The 240-frame animation was completed within a consistent timeframe using these hardware and render configurations.

3.14. Video Editing and Frame Sequence Integration:

In the final stage of the project, the rendered individual frames were merged using Blender's "Video Sequencer" panel and converted into a seamless loop.


C:\blender_sarkac\
Search
0009.png
0010.png
0011.png
0012.png
0013.png
0014.png
0015.png
0016.png
0017.png
0018.png
0019.png
0020.png
0021.png
0022.png
0023.png
0024.png
Move Strips
Relative Path
Fit Meth... Scale to Fit
Set View Transform
Import As: Auto Detect
Use Placeholders
Length 25
Add Image Strip
Cancel
Sequencer
View
Select
Marker
Add
Strip
Image
Edit
Shuffle
00:00+21
00:01+00
00:02+00
00:03+00
00:04+00
00:05+00
00:06+00
00:07+00
00:08+00
Channel 7
Channel 6
Channel 5
0001.001 | C:\blender_sarkac\0001.png | 215
0001.003 | C:\blender_sarkac\0001.png | 215


Format
Resolution X 1920 px
Y 1080 px
% 100%
Aspect X 1.000
Y 1.000
Render Region
Crop to Render Region
Frame Rate 60 fps
Frame Range
Frame Start 1
End 480
Step 1
Time Stretching
Stereoscopy

- **Utilization of Video Sequencer:** All frames rendered between 0001.png and 0240.png were imported into Blender's internal video editing panel using the **"Add -> Image Sequence"** command.

- **Frame Count and Loop Duration:** The simulation, initially planned as 240 frames, was extended to **480 frames** (double loop) to ensure motion continuity. This was achieved by duplicating the sequence and appending it to the end.

- **Resolution and Synchronization Resolution:**

    - **Problem Encountered:** Due to an accidental change in resolution or scaling settings during the rapid render process after the 25th frame, a size mismatch occurred in the video segments.
    - **Implemented Solution:** To rectify this error, Blender's **"Transform"** and **"Scale"** tools were employed. The mismatched clips were selected, and through **"Strip -> Transform -> Offset/Scale"** settings, they were millimetrically aligned with the primary video format (1920x1080). This eliminated the visual jump between the 25th and 26th frames.

- **Timeline Configurations:** To maintain a natural flow, the speed on the Timeline was synchronized with the original render rate of 60 FPS.

- **Final Output Format:** The 480-frame sequence was exported as a high-quality MP4 file using the **FFmpeg Video** container and the **H.264** codec.

- **Rendering:** The render process is initiated by pressing the F12 key.

4. RESULTS AND DISCUSSION

Within the scope of this project, the pendulum motion—one of the fundamental principles of classical mechanics—was successfully simulated using 3D modeling and physics-based rendering techniques.

- **Visual and Technical Achievement:** Upon project completion, the interaction of light with marble and metallic materials was achieved at a photorealistic level. The 240-frame looped animation was converted into a fluid video output at 60 frames per second (FPS).
- **Hardware Performance Analysis:** Rendering operations were performed on an **NVIDIA GeForce GTX 1650** GPU using **Cycles (OptiX)** hardware acceleration.
    - AI-based **Denoising** technology allowed for smooth images even at lower sampling rates, significantly optimizing the total rendering time.
- **Challenges and Resolutions:** The resolution mismatch encountered during the video editing phase (after the 25th frame) was resolved with millimetric precision using transform and scale tools within the Blender **Video Sequencer**, ensuring a seamless 480-frame loop.
- **Future Work:** This study can be evolved into interactive engineering simulations in the future by integrating it with real-time physics engines such as Unreal Engine.

5. REFERENCES

- **[1] Blender Documentation.** (2026). *Cycles Render Engine and Compositing Nodes*. https://docs.blender.org/
- **[2] Poly Haven.** (2026). *High Dynamic Range Imaging (HDRI) for Environmental Lighting*. https://polyhaven.com/
- **[3] AmbientCG.** (2026). *PBR Marble Textures and Material Maps*. https://ambientcg.com/
- **[4] Gemini AI (Google).** (2026). *Technical Documentation Writing and Reporting Assistant*. https://gemini.google.com/
- **[5] Youtube.** For training videos(2026). https://youtu.be/ZMsAuLevjdg?si=pH_07s6kx3NaL9vj