X3D SHADOW ANALYSIS

Dragan Knežević
Vesna Stojaković

FIT 2009, Novi Sad

Presentation contet

- **INTRODUCTION**
  - Assigment goel
  - Application
- **SHADOW ANALYSES**
  - The method presentation
  - Shadow matrix
  - X3D code
- **CONCLUSION**
  - Limitations
  - References

INTRODUCTION

Assigment goel

Application

Assignment goal

- The goal was to **simulate shadow** of an arbitrary object casted on a horizontal plane and **analyze transformation** needed to generate coordinates of a shadow vertex based on coordinates of an object vertex.
- Presented transformation must be applied for every vertex of an object in order to produce **realistic shadow**.

Application

- Shadows are very important for creation of **realistic 2D images**, since they determinate 3D object's geometry.
- Since shadows are part of real world images, they are also necessary for creation of **photorealistic rendering**.
- Type of matrix analysis applied in this research, can be relevant for **any central projection** of point to a plane.

SHADOW ANALYSES

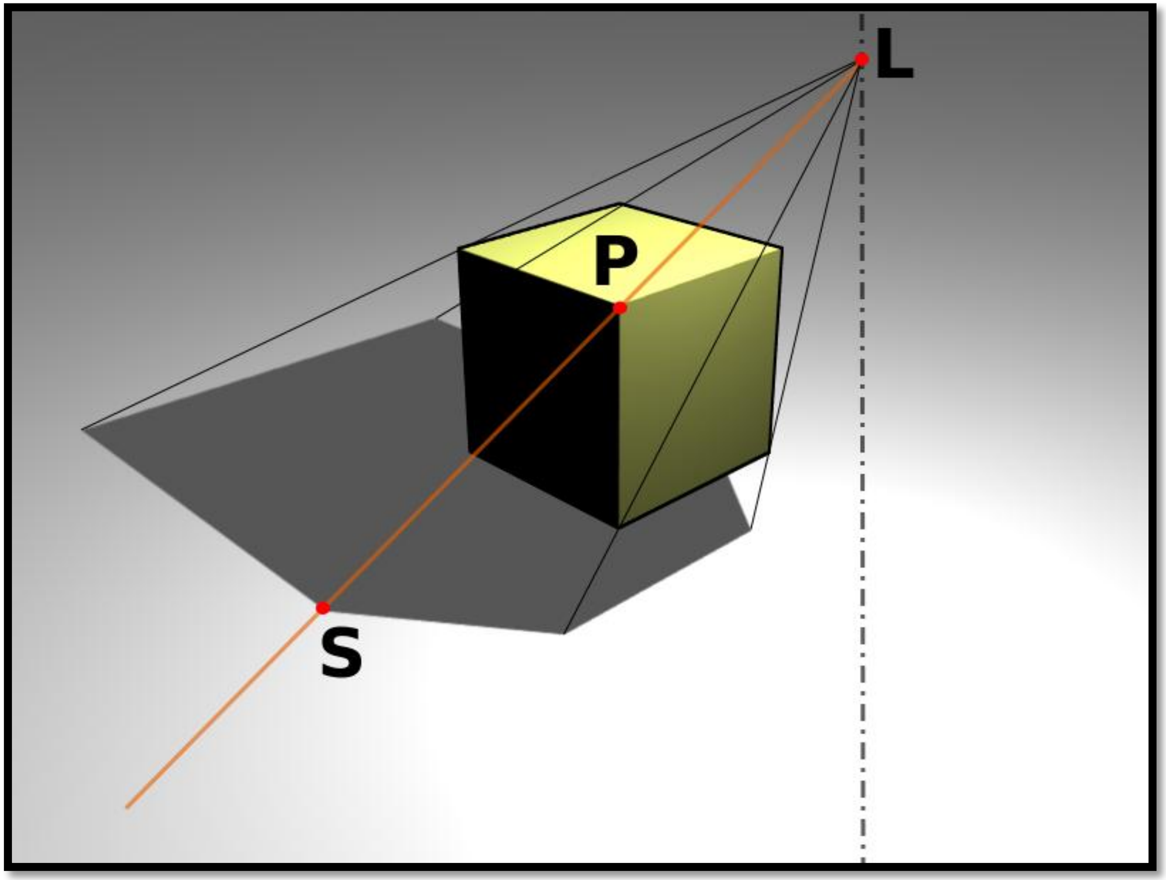
The method presentation

Shadow matrix

Relavant X3D code

The method presentation

- To explain relation between coordinates of point and coordinate of it's shadow, it is useful to introduce characteristic vectors S, P and L.
- **L** - the **light source** vector
- **P** - the **point** vector
- **S** - the **shadow** vector.



The method presentation

- According to definitions of these vectors, we can see that:

$$\overrightarrow{S} = \overrightarrow{P} + \alpha\left(\overrightarrow{P} - \overrightarrow{L}\right)$$

- Considering the given const

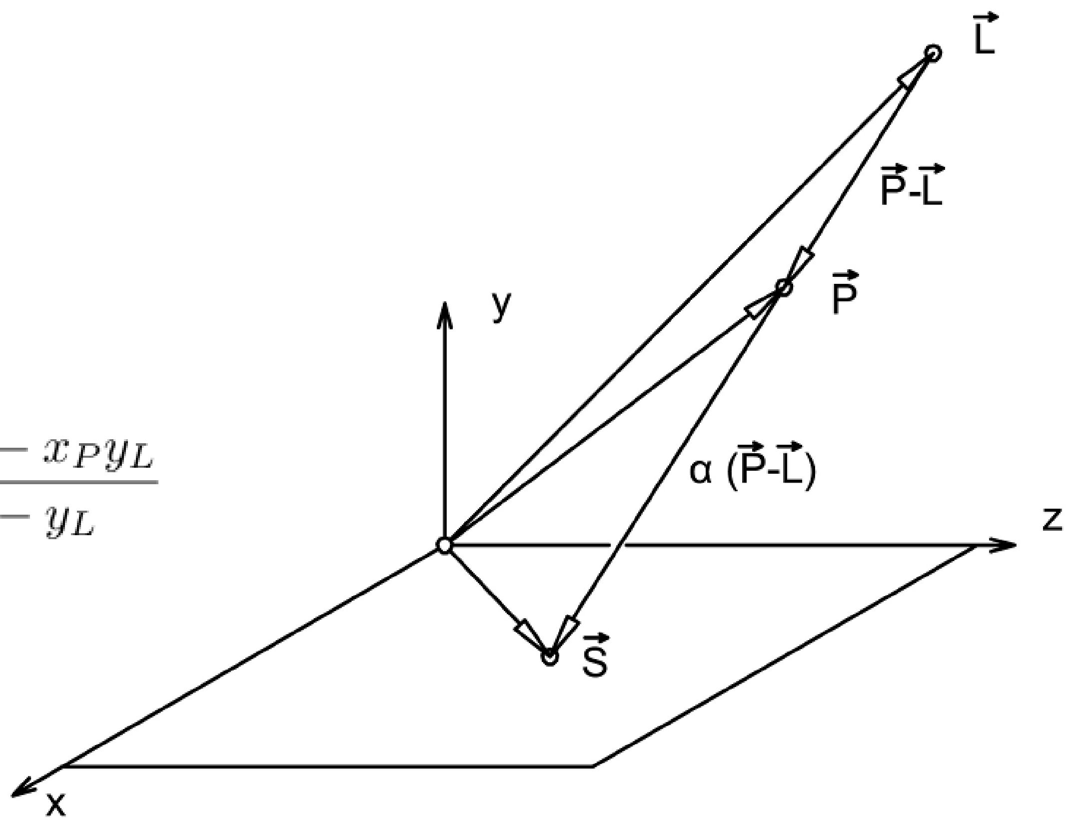
$$y_S = 0$$

- We can see that:

$$x_S = x_P - y_P \frac{x_P - x_L}{y_P - y_L} = \frac{x_L y_P - x_P y_L}{y_P - y_L}$$

- And similarly:

$$z_S = \frac{z_L y_P - z_P y_L}{y_P - y_L}$$



The method presentation

- **Relation** **S=P*M** allows extraction of transformation matrix M using the following procedure:

$$\vec{S} = \vec{P} \cdot M \qquad [x_S \; y_S \; z_S \; 1] = [x_P \; y_P \; z_P \; 1]\, M$$

$$[x_P \; y_P \; z_P \; 1]\, M = \left[\frac{x_L y_P - x_P y_L}{y_P - y_L} \; 0 \; \frac{z_L y_P - z_P y_L}{y_P - y_L} \; 1 \right]$$

$$\left[\begin{array}{cccc} x_S & y_S & z_S & 1 \end{array}\right] = \frac{1}{y_P - y_L} \left[\begin{array}{cccc} x_P & y_P & z_P & 1 \end{array}\right] \left[\begin{array}{cccc} -y_L & 0 & 0 & 0 \\ x_L & 0 & z_L & 0 \\ 0 & 0 & -y_L & 0 \\ 0 & 0 & 0 & y_P - y_L \end{array}\right]$$

Shadow matrix

Shadow matrix, **M**, defines transformation from coordinates of any point **P(x_P,y_P,z_P)** to its shadow S which appears on the plane **y=0** considering **light source L(x_L,y_L,z_L)** .

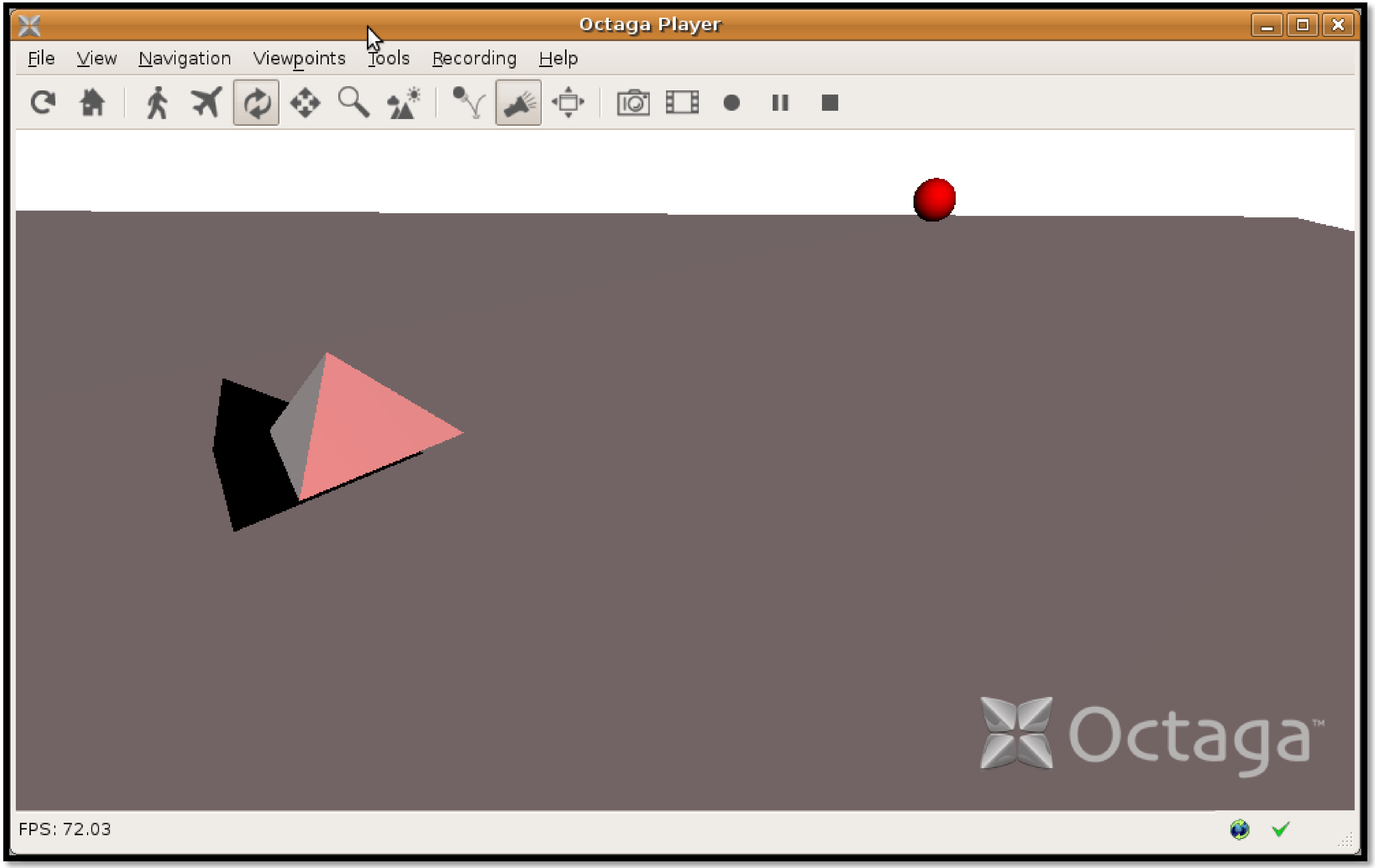
$$\begin{bmatrix} -y_L & 0 & 0 & 0 \\ x_L & 0 & z_L & 0 \\ 0 & 0 & -y_L & 0 \\ 0 & 0 & 0 & y_P - y_L \end{bmatrix}$$

Relevant X3D code

- Shape **”shadow”** contains resulting coordinate values and represents ”the shadow object”.

```
.
.
<Shape DEF="pyramid">
    <IndexedFaceSet coordIndex="1 0 3 -1 2 1 3 -1 0 2 3 -1 0 1 2">
            <Coordinate point="-2 3 -1, 2 3 -1, 0 3 2, 0 5 0" />
    </IndexedFaceSet>
    <Appearance>
            <Material/>
    </Appearance>
</Shape>
<Shape DEF="shadow">
    <IndexedFaceSet coordIndex="1 0 3 -1 2 1 3 -1 0 2 3 -1 0 1 2" solid="false">
            <Coordinate point="-7 0 -6.5, -1 0 -6.5, -4 0 -2, -10 0 -12.5" />
    </IndexedFaceSet>
    <Appearance>
            <Material diffuseColor="0 0 0"/>
    </Appearance>
</Shape>
.
.
.
```

Rendered output



CONCLUSION

Limitations

References

Limitations

- If **light source is positioned between highest point of object and plane** incorrect imaginary shadows which will appear
- Additionally, there is **"division by zero"** problem if $y_P = y_L$ is satisfied.
- If **plane** figure on which shadow is casted is **finite,** the shadow can be partially placed outside its borders.
- Presented procedure is **limited to y = 0 plane.**

References

- Introduction to 3D game programming with DirectX 9.0c
  http://books.google.com/books?id=is6Wc3hKSWIC
  Author: Frank D. Luna
- Two Shadow Rendering Algorithms
  http://web.cs.wpi.edu/~matt/courses/cs563/talks/shadow/shadow.html
  Author: Chris Bently

THANK YOU FOR YOUR ATENTION! ☺