1 Introduction

1.1 Name

The name of this component is "ProjectiveTextureMapping". This name shall be used when referring to this component in the COMPONENT statement.
2 Concepts

2.1 Overview

This component provides additional texturing extensions to the basic capabilities defined in X3D. Generally, 2D texture mapping has been used to enhance the quality of an image generated with a camera or to speed up the generation of an image with respect to a given scene including several geometric models. However, there are some constraints for mapping region and shape of textures over objects. As an extension of texture mapping, projective texture mapping has been proposed which allows the texture image to be projected onto a scene in the view volume constructed from a specific viewpoint with a location and an orientation. Figure 1 shows two example screen shots for projective texture mapping.

![Projective Texture Mapping Diagram]

Figure 1. An example of projective texture mapping [1]

Additionally, several texture images can be projected onto the same scene like multi-texturing. Moreover, we can provide multiple projective texture mapping over a common scene with specific objectives such as photogrammetry or reconstruction of endoscope
images, visualization of images captured by CCTVs in a virtual space. As shown Figure 2, assume that several images are provided, each of which is taken with a different camera. Construction of a terrain surface from those images can be performed by displaying overlapping images obtained after applying several projective textures to the surface model.

Figure 2. A construction of a terrain surface [2]

Figure 3 describes an example for reconstructing endoscope images over a cylinder by applying projective texture mapping. In a similar manner, each image is captured from an endoscope with perspective view information inside human body.

Figure 3. An example for reconstructing endoscope images by applying projective texture mapping [3]
2.2 Projective Texture Mapping Concepts

Projective Texture Mapping can be defined with a viewpoint, a view volume, a texture image and scene objects, where a viewpoint and a view volume can be specified similar to camera specification information. In the other hand, projective texture mapping allows the texture image to be projected onto a scene in the view volume constructed from a specific viewpoint with a location and an orientation as shown Figure 1. The texture image projected onto scene objects can be changed according to transformation of specific viewpoints and scene objects.

2.3 Projective texture map image formats

Node types specifying projective texture maps may supply data with a number of color components between one and four. The valid types and interpretations of projective textures are identical to that for 2D textures. The definition of texture formats is defined in 18.2.1 Texture map formats.

3 Specify projective texture mapping into X3D specification

3.1 Overview

As illustrated in Section 2, projective texture mapping can be specified by using the following three items: a texture image or multitexture image, a slider projector including a viewpoint and a view volume, and scene objects. Since there are the existing nodes in X3D specification related to texture images and scene such as Texture nodes (ImageTexture, MovieTexture, MultiTexture, MultiTextureCoordinate, MultiTextureTransform, PixelTexture, TextureCoordinate, TextureCoordinateGenerator, TextureProperties, TextureTransform) and Object Nodes, there is no any problem in specifying texture images and objects for projective texture mapping. For a slider projector, the existing nodes in X3D such as X3DViewpointNode may be used. Clearly, the node defines a specific location in the local coordinate system from which the user
may view the scene and is used as Bindable children nodes (using stack). There are three nodes including OrthoViewpoint: for an orthographic view, Viewpoint for a perspective view, and ViewpointGroup for supporting a group. Assume the navigation nodes are directly used to specify a slide projector. There are the following problems: first, confuse to distinguish navigators in browsers and a slider projector in projective texture mapping, and second cannot identify between both.

Due to the reasons, a specification of projective texture images within specific slide projector cannot be performed into X3D by using the existing nodes of X3D. Therefore, new nodes for projective texture mapping must be defined additionally as follows.

- **X3DProjectiveTextureNode**
  - OrthoTexture
  - PerspectiveTexture
  - ProjectiveTextureGroup

4 The proposed nodes

```plaintext
X3DProjectiveTextureNode
{
  SFString       [in, out] description ""
  SFVec4f/d     [in, out] position 0 0 10 1
  SFVec3f/d     [in, out] direction 0 0 1
}
```
4.1 OrthoTexture

- OrthoTexture : X3DProjectiveTextureNode {
  SFString [in, out] description ""
  SFVec4f/d [in, out] position 0 0 1 0
  SFVec3f/d [in, out] direction 0 0 1
  MFFloat [in,out] viewVolume -1 1 -1 1 -1 1 // (viewvolume)
  SFNode [in,out] texture NULL [TextureNode]
  SFNode [in,out] textureTransform NULL [TextureTransformNode]
}

The OrthoTexture node defines a projective texture map by specifying general parameters for orthogonal texture mapping.

The position and direction fields define position and direction for doing orthogonal projection. The 4th value of the position field is always zero.

The viewVolume field specify the view volume of orthogonal projection.

The texture field contains a single texture node such as ImageTexture, MovieTexture, MultiTexture, and PixelTexture.
4.2 PerspectiveTexture

```plaintext
PerspectiveTexture : X3DProjectiveTextureNode {
    SFString [in, out] description ""
    SFVec4f/d [in, out] position 0 0 10 1
    SFVec3f/d [in, out] direction 0 0 1
    SFFloat [in, out] fieldOfView n/4 (0, n) // field of view
    SFFloat [in, out] aspectRatio 1 // aspect ratio
    MFFloat [in, out] minmax 1 10 // min-max z value
    SFNode [in, out] texture NULL [TextureNode]
    SFNode [in, out] texture Transform NULL [TextureTransformNode]
}
```

The PerspectiveTexture node defines a projective texture map by specifying general parameters for perspective texture mapping.

The position and direction fields define position and direction for doing perspective projection. The 4th value of the position field must be always one.

The fovy, aspectRatio, and depth fields define the values of fovy, aspect ratio, and depth of perspective projection.

The texture field contains a single texture node such as ImageTexture, MovieTexture, MultiTexture, and PixelTexture.
4.3 ProjectiveTextureGroup

- ProjectiveTextureGroup:X3DChildNode{
  
  SFFloat [in,out] alpha 1 [0,1]
  
  SFCOLOR [in,out] color 1 1 1 [0,1]
  
  MFString [in,out] function []
  
  SFNode [in,out] metadata NULL
  
  MFString [in,out] mode []
  
  MFString [in,out] source []
  
  MFNode [in,out] projectiveTexture [] [X3DProjectiveTextureNode]

}

The ProjectiveTextureGroup node specifies the application of several individual projective textures to 3D objects in a virtual world to achieve a more complex visual effect.

The projectiveTextures field contains a list of projective texture nodes (e.g., OrthoTexture, PerspectiveTexture). The projectiveTextures field may not contain another ProjectiveTextureGroup node.

The color and alpha fields define base RGB and alpha values for SELECT mode operations.

The mode field controls the type of blending operation of projected overlapping geometry area in a virtual world from two or more than two projective texture mappings. The available modes include MODULATE for a lit Appearance, REPLACE for an unlit Appearance, and several variations of the two. The mode field may contain an additional blending mode for the alpha channel.

EXAMPLE The mode value "MODULATE","REPLACE" specifies Color = (Arg1.color × Arg2.color, Arg1.alpha).

The source and function fields are equally defined in those in MultiTexture node.
5 The Examples

5.1 Example 1

The following figure is visualizing one example of perspective texture mapping with position="8.0 8.0 -22.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = "1", minmax = "7 30" and <PixelTexture url="sample1 rgba"/> with respect to a given virtual world.

Figure 4. The result of perspective texture mapping with respect to a wall
The following X3D file is used to obtain the result for above perspective texture mapping.

```xml
<Scene><!--Scene graph nodes are added here -->
  <Transform>
    <Shape>
      <PerspectiveTexture description="sample1", position="8.0 8.0 -22.0", direction="-6 - 6 -8", fieldOfView="45.0", aspectRatio = "1", minmax = "7 30">
        <PixelTexture url="sample1.rgba"/>
      </PerspectiveTexture>
      <Appearance>
        <Material diffuseColor="0 0 0"/>
      </Appearance>
      <!--define walls by using IndexedFaceSet -->
      .................................................................
    </Shape>
  </Transform>
</Scene>
```
5.2 Example 2

The following figure is visualizing one example of perspective texture mapping with position="8.0 8.0 -22.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = “1”, minmax = “7 30”> <PixelTexture url="sample2.rgba"/> with respect to a given virtual world containing a cube.

![Figure 5. The result of perspective texture mapping with respect to a solid cube](slide_projector.png)
<Scene><!--Scene graph nodes are added here -->
  <Transform>
    <Shape>
      <PerspectiveTexture description="sample1", position="8.0 8.0 -22.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = “1” minmax = “7 30”>
        <PixelTexture url="sample2.rgba"/>
      </PerspectiveTexture>
      <Appearance>
        <Material diffuseColor="0 0 0"/>
      </Appearance>
      <!--define cylinder -->
      ……………………………………………………….
    </Shape>
  </Transform>
</Scene>
5.3 Example 3

The following figure is visualizing one example of perspective texture mapping with position=“8.0 8.0 -22.0”, direction=“-6 -6 -8”, fieldOfView=“45.0”, aspectRatio = “1”, minmax = “7 30”> <PixelTexture url=“sample2.rgba”/> with respect to a given virtual world containing walls and a flower object.

Figure 6. The result of perspective texture mapping with respect to two objects.
<Scene><!--Scene graph nodes are added here -->
  <Transform>
    <Shape>
      <PerspectiveTexture description="sample1", position="8.0 8.0 -22.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = "1" minmax = "7 30">
        <PixelTexture url="sample2.rgba"/>
      </PerspectiveTexture>
      <Appearance>
        <Material diffuseColor="0 0 0"/>
      </Appearance>
      <!--define wall -->
      ..........................................................
      <!--define model -->
      ..........................................................
    </Shape>
  </Transform>
</Scene>
5.4 Example 4

The following figure is visualizing one example of perspective texture mapping with position="8.0 8.0 -22.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = “1”, minmax = “7 30”> <PixelTexture url="sample2 rgba"/> with respect to a given virtual world containing a flower object and a airplane object.

Figure 7. The result of perspective texture mapping with respect to two objects.
<Scene> <!-- Scene graph nodes are added here -->
  <Transform>
    <Shape>
      <PerspectiveTexture description="sample1", position="8.0 8.0 -22.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = "1" minmax = "7 30">
        <PixelTexture url="spotlight.rgb"/>
      </PerspectiveTexture>
      <Appearance>
        <Material diffuseColor="0 0 0"/>
      </Appearance>
      <!--define model1 -->
      .................................................................
    </Shape>
    <!--define model2 -->
    .................................................................
  </Transform>
</Scene>
5.5 Example 5

The following figure is visualizing an example of two perspective texture mappings with

```xml
<PerspectiveTexture description="sample1", position="8.0 8.0 -22.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = "1" minmax = "7 30">
  <PixelTexture url="spotlight.rgba"/>
</PerspectiveTexture>,

<PerspectiveTexture description="sample1", position="6.0 6.0 -20.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = "1" minmax = "7 30">
  <PixelTexture url="spotlight.rgba"/>
</PerspectiveTexture>
```

with respect to a given virtual world containing a flower object and a airplane object.

Figure 8. The result for two perspective texture mapping with respect to two objects.
<Scene><!--Scene graph nodes are added here -->
  <Transform>
    <Shape>
      <PerspectiveTexture description="sample1", position="8.0 8.0 -22.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = "1" minmax = "7 30">
        <PixelTexture url="spotlight.rgba"/>
      </PerspectiveTexture>
      <Appearance>
        <Material diffuseColor="0 0 0"/>
      </Appearance>
      <!--define model1 -->
  </Shape>
  </Transform>
  <Transform>
    <Shape>
      <PerspectiveTexture description="sample1", position="6.0 6.0 -20.0", direction="-6 -6 -8", fieldOfView="45.0", aspectRatio = "1" minmax = "7 30">
        <PixelTexture url="spotlight.rgba"/>
      </PerspectiveTexture>
      <Appearance>
        <Material diffuseColor="0 0 0"/>
      </Appearance>
      <!--define model2 -->
Bibliography