

# | X3D Chromakey

Web3D Korea Standardization Meeting at SIGGRAPH

July 31, 2017

Myeong Won Lee  
The University of Suwon

# Chroma Key

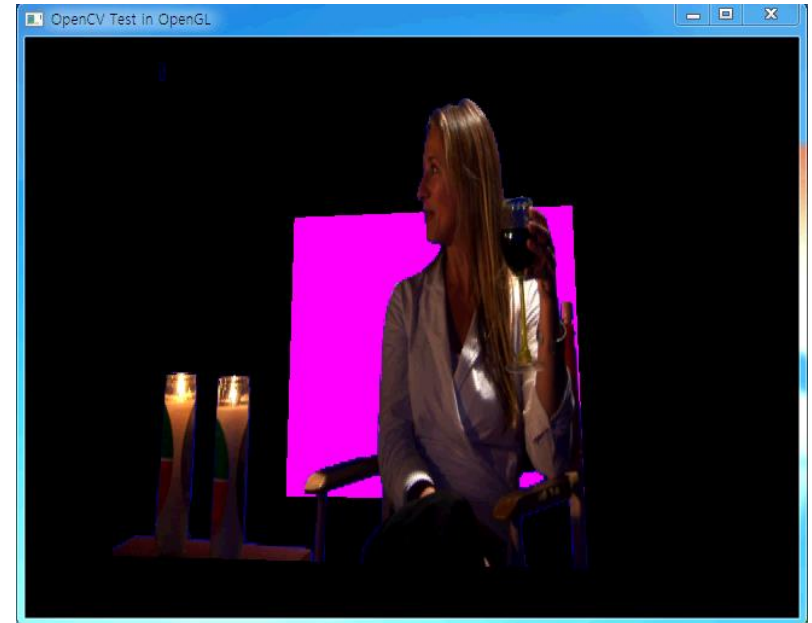
- Original concept: 2D chroma key
  - Chroma key is a special effects / post-production technique for compositing (layering) two images or video streams together based on color hues (chroma range)
  - The technique has been used heavily in many fields (newscasting, motion picture, videogaming) to remove a background from the subject of a photo or video
  - A color range in the top layer is made transparent, revealing another image behind. The chroma keying technique is commonly used in video production and post-production
- 3D chroma key
  - We define 3D chroma key as a 2D chroma key image or video located in a 3D scene

# Chroma Key Video Example

Before chroma key: video  
with blue screen



After chroma key: video  
extracted into a 3D scene

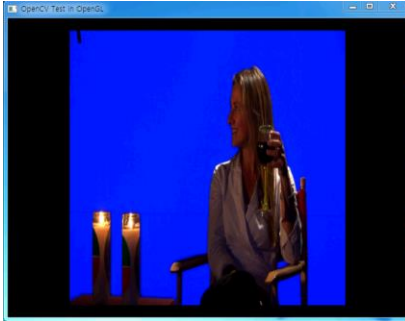


# 3D Chroma Key

- 3D chroma keying procedure
  - Capture a video with object motion in front of a blue or green background screen
  - A chroma difference (range) is extracted as a key value from the blue or green background screen using RGB values
  - Separate the video object from the background video using the key value
  - Combine the separated video object with a 3D scene
- New proposal for 3D chroma keying
  - Both 2D and 3D chroma key texture nodes should be defined
  - X3D scene structure should define a geometric or physical location for the chroma key image or video
  - Texture node should have parameters for chroma key video and image

# Transparency Processing

- A color range is not necessary with a single unique background color



- A color range should be provided when the background color is not unique due to the effects of light or other environmental conditions



# Chroma Key Parameters

- Additional parameters for TextureProperties Node
  - SFBool [in,out] chromakey FALSE
  - SFColorRGBA [in,out] minChromakeyColor 0 0 0 0 [0,1]
  - SFColorRGBA [in,out] maxChromakeyColor 0 0 0 0 [0,1]
- Chroma key
  - Select processing of MovieTexture node with chroma key if TRUE
- Chroma key color
  - Change the color to transparent if chroma key processing is selected

CASE 1:  
Define chromakey parameters in  
Texture properties

# TextureProperties

```
TextureProperties : X3DNode {  
  SFFloat      [in,out] anisotropicDegree  1.0    [1,∞)  
  SFColorRGBA [in,out] borderColor        0 0 0 0  [0,1]  
  SFInt32     [in,out] borderWidth       0      [0,1]  
  SFString    [in,out] boundaryModeS     "REPEAT" [see Table 18.7]  
  SFString    [in,out] boundaryModeT     "REPEAT" [see Table 18.7]  
  SFString    [in,out] boundaryModeR     "REPEAT" [see Table 18.7]  
  SFString    [in,out] magnificationFilter "FASTEST" [see Table 18.8]  
  SFNode      [in,out] metadata          NULL    [X3DMetadataObject]  
  SFString    [in,out] minificationFilter "FASTEST" [see Table 18.9]  
  SFString    [in,out] textureCompression "FASTEST" [see Table 18.10]  
  SFFloat     [in,out] texturePriority   0      [0,1]  
  SFBool      []    generateMipMaps     FALSE  
}
```



## TextureProperties (Update)

```
TextureProperties : X3DNode {  
    SFFloat      [in,out] anisotropicDegree  1.0    [1,∞)  
    SFColorRGBA [in,out] borderColor        0 0 0 0  [0,1]  
    SFInt32     [in,out] borderWidth        0      [0,1]  
    SFString    [in,out] boundaryModeS     "REPEAT" [see Table 18.7]  
    SFString    [in,out] boundaryModeT     "REPEAT" [see Table 18.7]  
    SFString    [in,out] boundaryModeR     "REPEAT" [see Table 18.7]  
    SFString    [in,out] magnificationFilter "FASTEST" [see Table 18.8]  
    SFNode      [in,out] metadata          NULL    [X3DMetadataObject]  
    SFString    [in,out] minificationFilter "FASTEST" [see Table 18.9]  
    SFString    [in,out] textureCompression "FASTEST" [see Table 18.10]  
    SFFloat     [in,out] texturePriority    0      [0,1]  
    SFBool      []    generateMipMaps      FALSE  
    SFBool      [in,out] chromakey         FALSE  
    SFColor     [in,out] minChromakeyColor 0 0 0 [0,1]  
    SFColor     [in,out] maxChromakeyColor 0 0 0 [0,1]  
}
```

# MovieTexture

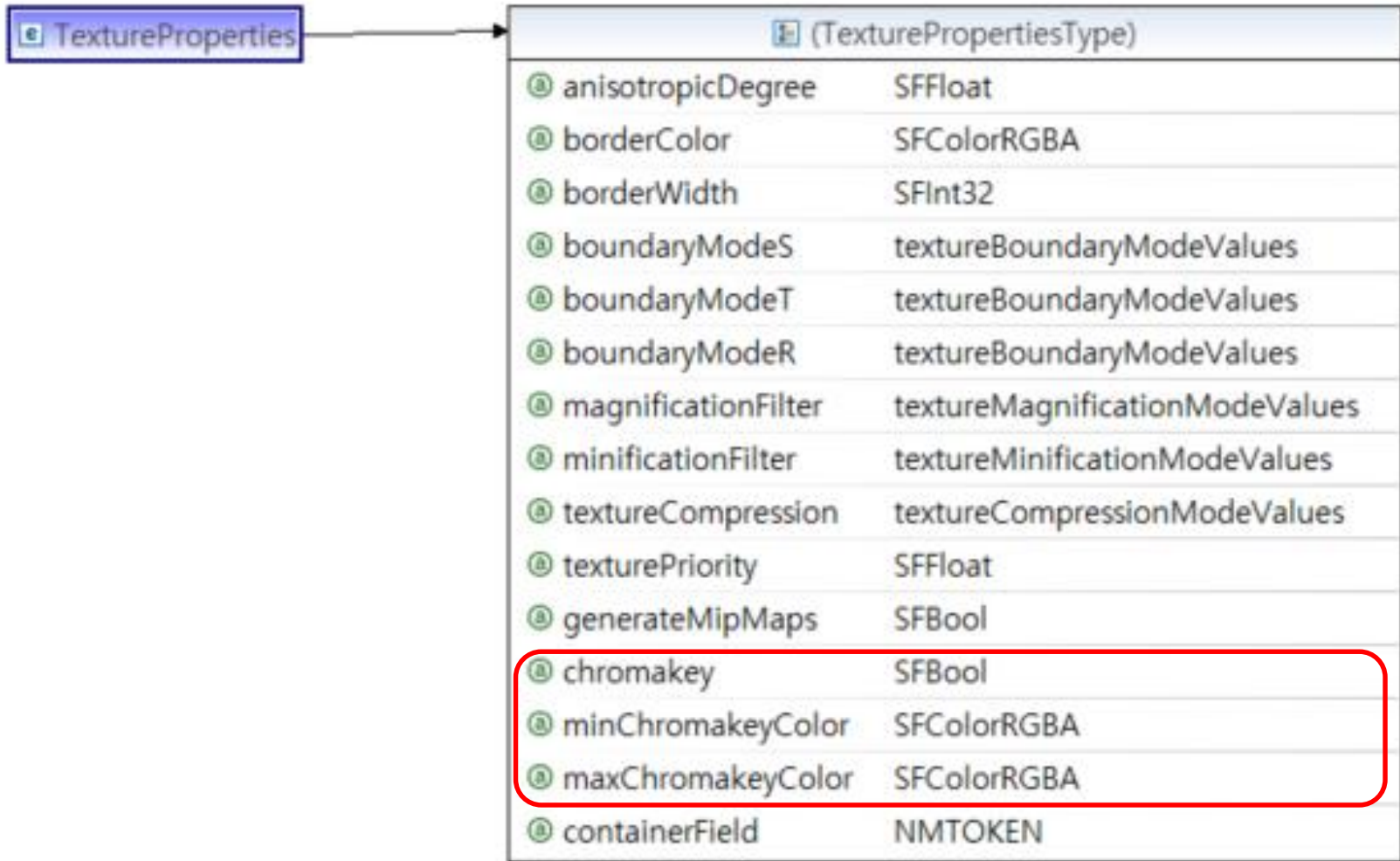
```
MovieTexture : X3DTexture2DNode, X3DSoundSourceNode, X3DUrlObject {  
  SFString [in,out] description    ""  
  SFBool   [in,out] loop           FALSE  
  SFNode   [in,out] metadata       NULL [X3DMetadataObject]  
  SFTime   [in,out] pauseTime      0 (-∞,∞)  
  SFFloat  [in,out] pitch          1.0 (0,∞)  
  SFTime   [in,out] resumeTime     0 (-∞,∞)  
  SFFloat  [in,out] speed          1.0 (-∞,∞)  
  SFTime   [in,out] startTime      0 (-∞,∞)  
  SFTime   [in,out] stopTime       0 (-∞,∞)  
  MFString [in,out] url            [] [URI]  
  SFTime   [out]   duration_changed  
  SFTime   [out]   elapsedTime  
  SFBool   [out]   isActive  
  SFBool   [out]   isPaused  
  SFBool   []      repeatS         TRUE  
  SFBool   []      repeatT         TRUE  
  SFNode   []      textureProperties NULL [TextureProperties]  
}
```

## Texture3D Node

```
X3DTexture3DNode : X3DTextureNode {  
  SFNode [in,out] metadata      NULL [X3DMetadataObject]  
  SFBool []    repeatS          FALSE  
  SFBool []    repeatT          FALSE  
  SFBool []    repeatR          FALSE  
  SFNode []    textureProperties NULL [TextureProperties]  
}
```

# TextureProperties (Update)

- XML Schema Diagram



# TextureProperties (Update)

- XML Schema Code

```
<xs:element name="TextureProperties">
  <xs:complexType>
    <xs:complexContent>
      <xs:extension base="X3DAppearanceChildNode">
        <xs:attribute name="anisotropicDegree" type="SFFloat" default="0"/>
        <xs:attribute name="borderColor" type="SFColorRGBA" default="0 0 0 0"/>
        <xs:attribute name="borderWidth" type="SFInt32" default="0"/>
        <xs:attribute name="boundaryModeS" type="textureBoundaryModeValues" ... />
        <xs:attribute name="boundaryModeT" type="textureBoundaryModeValues" ... />
        <xs:attribute name="boundaryModeR" type="textureBoundaryModeValues" ... />
        <xs:attribute name="magnificationFilter" type="textureMagnificationModeValues" ... />
        <xs:attribute name="minificationFilter" type="textureMinificationModeValues" ... />
        <xs:attribute name="textureCompression" type="textureCompressionModeValues" ... />
        <xs:attribute name="texturePriority" type="SFFloat" default="0"/>
        <xs:attribute name="generateMipMaps" type="SFBool" default="false"/>
        <xs:attribute name="chromakey" type="SFBool" default="false"/>
        <xs:attribute name="minChromakeyColor" type="SFColorRGBA" default="0 0 0 0"/>
        <xs:attribute name="maxChromakeyColor" type="SFColorRGBA" default="0 0 0 0"/>
        <xs:attribute name="containerField" type="xs:NMTOKEN" default="..."/>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
</xs:element>
```

# MovieTexture

```
MovieTexture : X3DTexture2DNode, X3DSoundSourceNode, X3DUrlObject {  
  SFString [in,out] description    ""  
  SFBool   [in,out] loop           FALSE  
  SFNode   [in,out] metadata       NULL [X3DMetadataObject]  
  SFTime   [in,out] pauseTime      0 (-∞,∞)  
  SFFloat  [in,out] pitch          1.0 (0,∞)  
  SFTime   [in,out] resumeTime     0 (-∞,∞)  
  SFFloat  [in,out] speed          1.0 (-∞,∞)  
  SFTime   [in,out] startTime      0 (-∞,∞)  
  SFTime   [in,out] stopTime       0 (-∞,∞)  
  MFString [in,out] url            [] [URI]  
  SFTime   [out]   duration_changed  
  SFTime   [out]   elapsedTime  
  SFBool   [out]   isActive  
  SFBool   [out]   isPaused  
  SFBool   []      repeatS         TRUE  
  SFBool   []      repeatT         TRUE  
  SFNode   []      textureProperties NULL [TextureProperties]  
}
```

CASE 2:  
Define chromakey parameters in  
Texture 2D Node and Texture 3D Node

## X3DTexture2DNode

```
X3DTexture2DNode : X3DTextureNode {  
  SFNode [in,out] metadata      NULL [X3DMetadataObject]  
  SFBool []   repeatS          TRUE  
  SFBool []   repeatT          TRUE  
  SFNode []   textureProperties NULL [TextureProperties]  
}
```

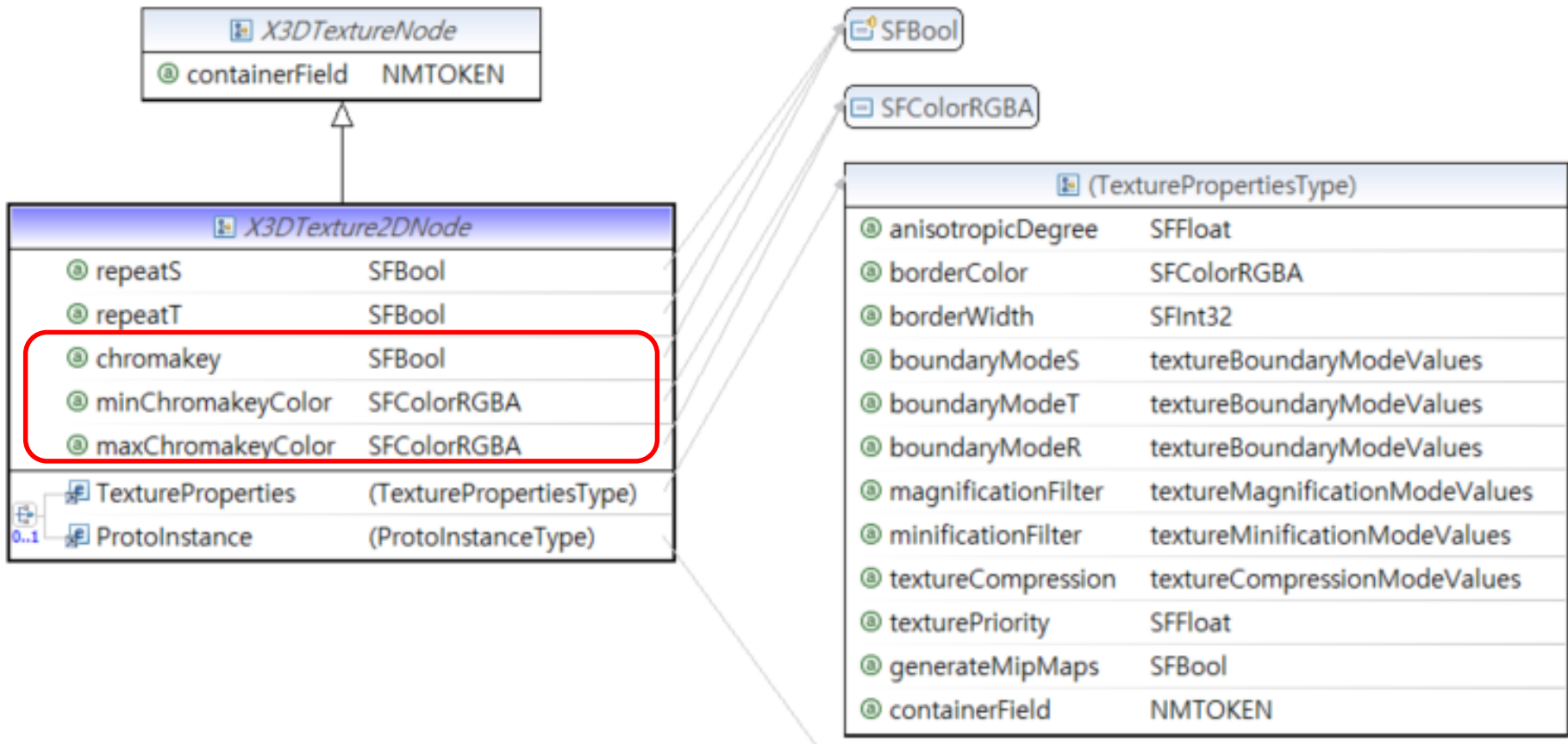


## X3DTexture2DNode (Update)

```
X3DTexture2DNode : X3DTextureNode {  
  SFNode [in,out] metadata      NULL [X3DMetadataObject]  
  SFBool []   repeatS          TRUE  
  SFBool []   repeatT          TRUE  
  SFNode []   textureProperties NULL [TextureProperties]  
  SFBool      [in,out] chromakey FALSE  
  SFColorRGBA [in,out] minChromakeyColor 0 0 0 0 [0,1]  
  SFColorRGBA [in,out] maxChromakeyColor 0 0 0 0 [0,1]  
}
```

# X3DTexture2DNode (Update)

- XML Schema Diagram



# X3DTexture2DNode (Update)

- XML Schema Code

```
<xs:complexType name="X3DTexture2DNode" abstract="true" mixed="false">
  <xs:complexContent>
    <xs:extension base="X3DTextureNode">
      <xs:choice minOccurs="0">
        <xs:annotation>
          <xs:documentation>textureProperties</xs:documentation>
        </xs:annotation>
        <xs:element ref="TextureProperties"/>
        <xs:element ref="ProtoInstance">
          <xs:annotation>
            <xs:documentation>Appropriately typed substitute node</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:choice>
      <xs:attribute name="repeatS" type="SFBool" default="true"/>
      <xs:attribute name="repeatT" type="SFBool" default="true"/>
      <xs:attribute name="chromakey" type="SFBool" default="false"/>
      <xs:attribute name="minChromakeyColor" type="SFColorRGBA" default="0 0 0 0"/>
      <xs:attribute name="maxChromakeyColor" type="SFColorRGBA" default="0 0 0 0"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

## Texture3D Node

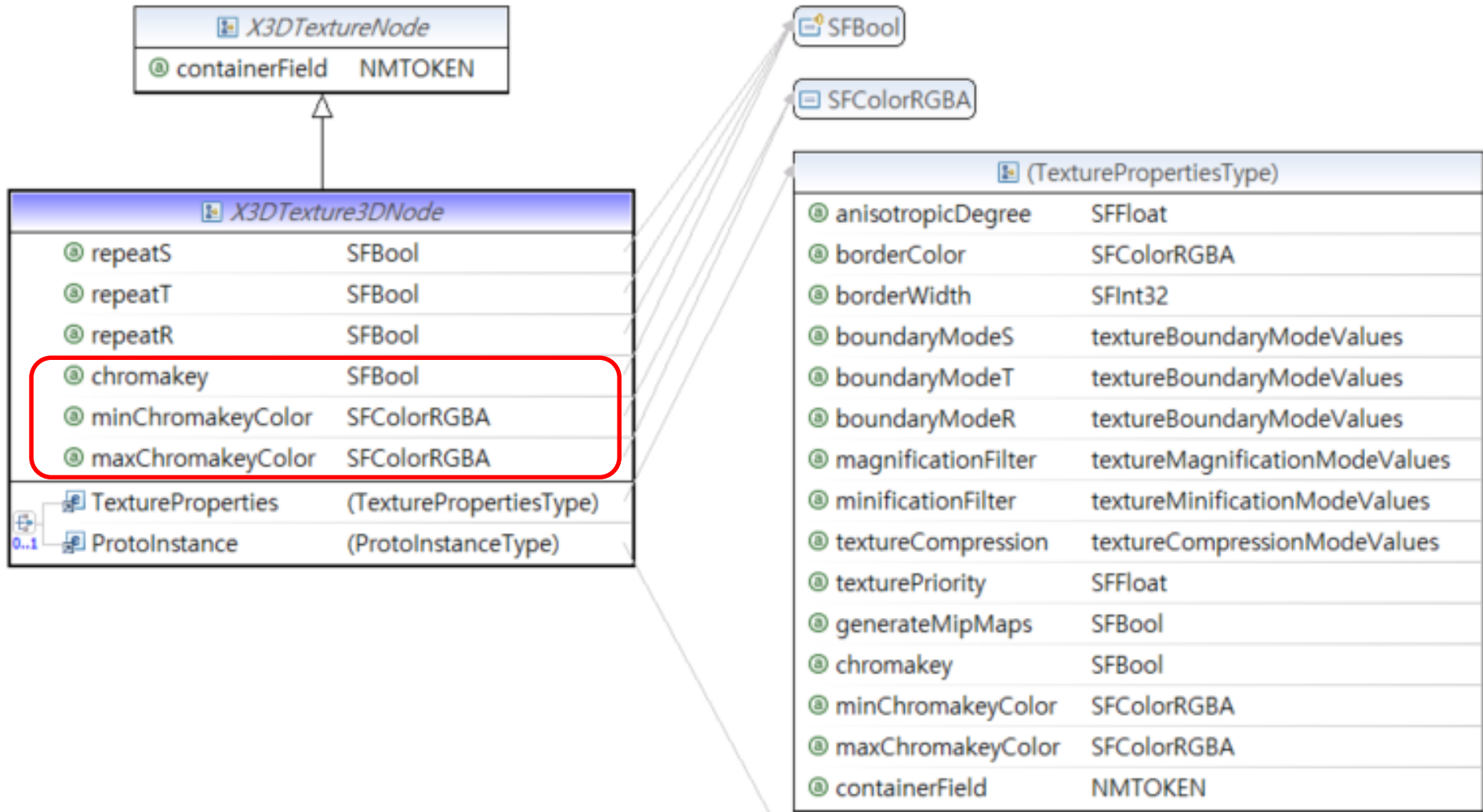
```
X3DTexture3DNode : X3DTextureNode {  
  SFNode [in,out] metadata      NULL [X3DMetadataObject]  
  SFBool []      repeatS        FALSE  
  SFBool []      repeatT        FALSE  
  SFBool []      repeatR        FALSE  
  SFNode []      textureProperties NULL [TextureProperties]  
}
```

## X3DTexture3D Node (update)

```
X3DTexture3DNode : X3DTextureNode {  
  SFNode [in,out] metadata      NULL [X3DMetadataObject]  
  SFBool []    repeatS          FALSE  
  SFBool []    repeatT          FALSE  
  SFBool []    repeatR          FALSE  
  SFNode []    textureProperties NULL [TextureProperties]  
  SFBool      [in,out] chromakey FALSE  
  SFColorRGBA [in,out] minChromakeyColor 0 0 0 0 [0,1]  
  SFColorRGBA [in,out] maxChromakeyColor 0 0 0 0 [0,1]  
}
```

# X3DTexture3DNode (Update)

- XML Schema Diagram



# X3DTexture3DNode (Update)

- XML Schema Code

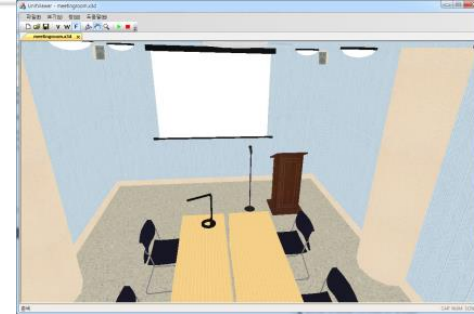
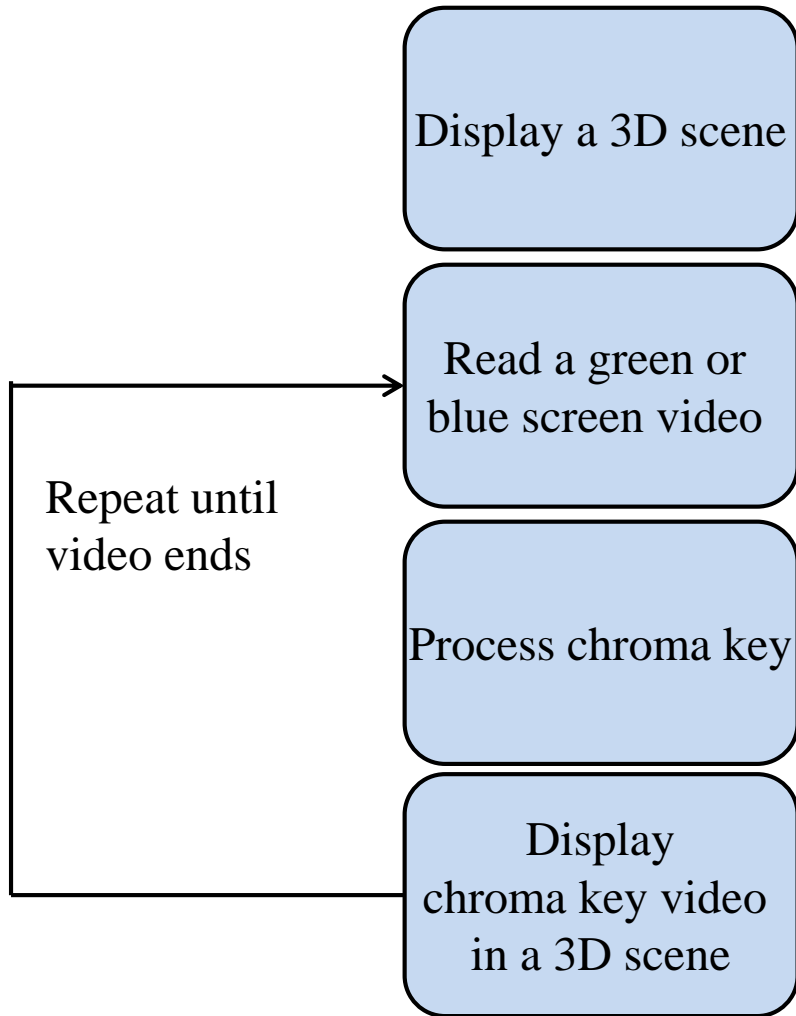
```
<xs:complexType name="X3DTexture3DNode" abstract="true" mixed="false">
  <xs:complexContent>
    <xs:extension base="X3DTextureNode">
      <xs:choice minOccurs="0">
        <xs:annotation>
          <xs:documentation>textureProperties</xs:documentation>
        </xs:annotation>
        <xs:element ref="TextureProperties"/>
        <xs:element ref="ProtoInstance">
          <xs:annotation>
            <xs:documentation>Appropriately typed substitute node</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:choice>
      <xs:attribute name="repeatS" type="SFBool" default="true"/>
      <xs:attribute name="repeatT" type="SFBool" default="true"/>
      <xs:attribute name="repeatR" type="SFBool" default="true"/>
      <xs:attribute name="chromakey" type="SFBool" default="false"/>
      <xs:attribute name="minChromakeyColor" type="SFColorRGBA" default="0 0 0 0"/>
      <xs:attribute name="maxChromakeyColor" type="SFColorRGBA" default="0 0 0 0"/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

## Chroma Key Video Display

- Generate RGBA image after transparency processing for a specific color or color range
- Texture mapping to a rectangle object using alpha blending



# Procedure for X3D Chroma Key Video Generation



an X3D scene

```
frame = cvQueryFrame( m_capture );  
...  
...
```

```
if(pData[i] < 150 && pData[i+1] > 127 &&  
    pData[i + 2] < 150 )  
    pData[i + 3] = 0xFF;  
...  
...
```

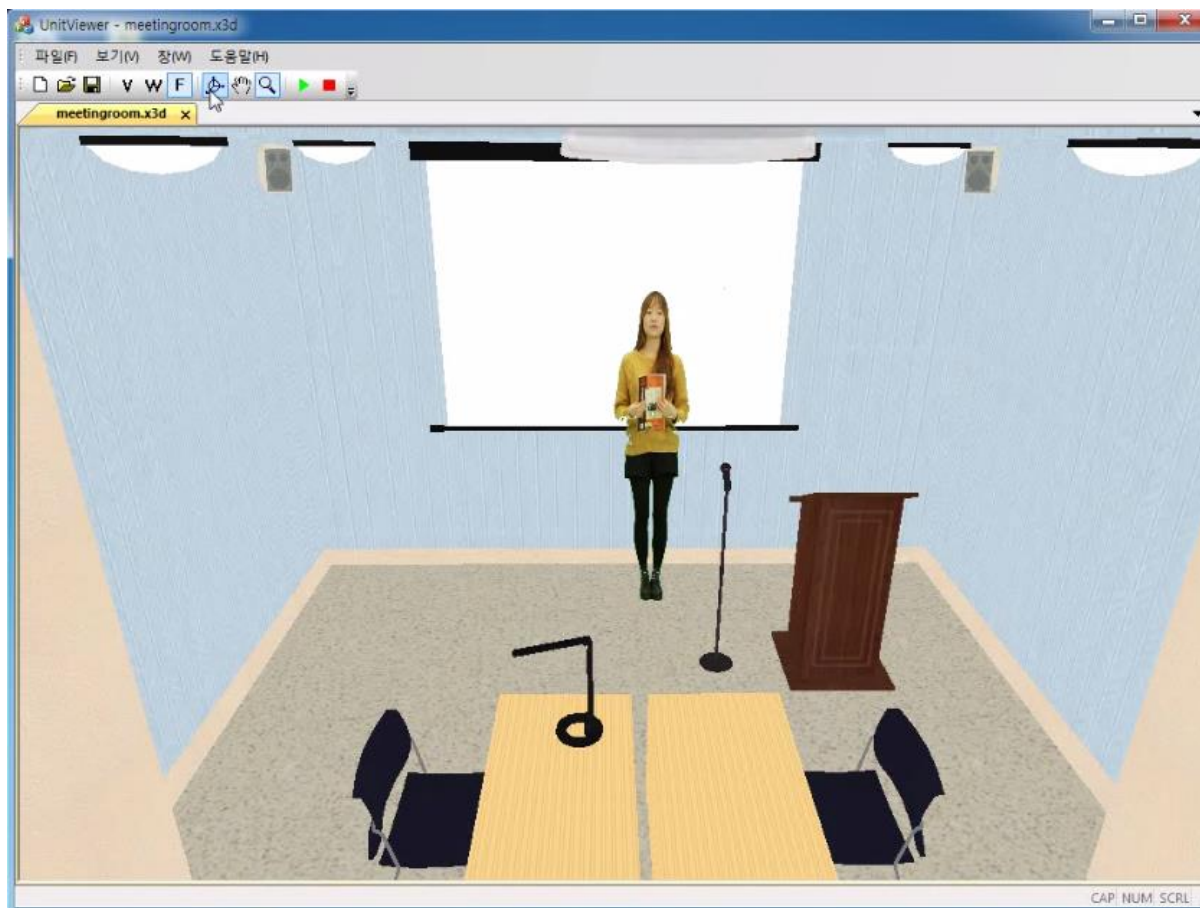


an X3D scene with chromakey video

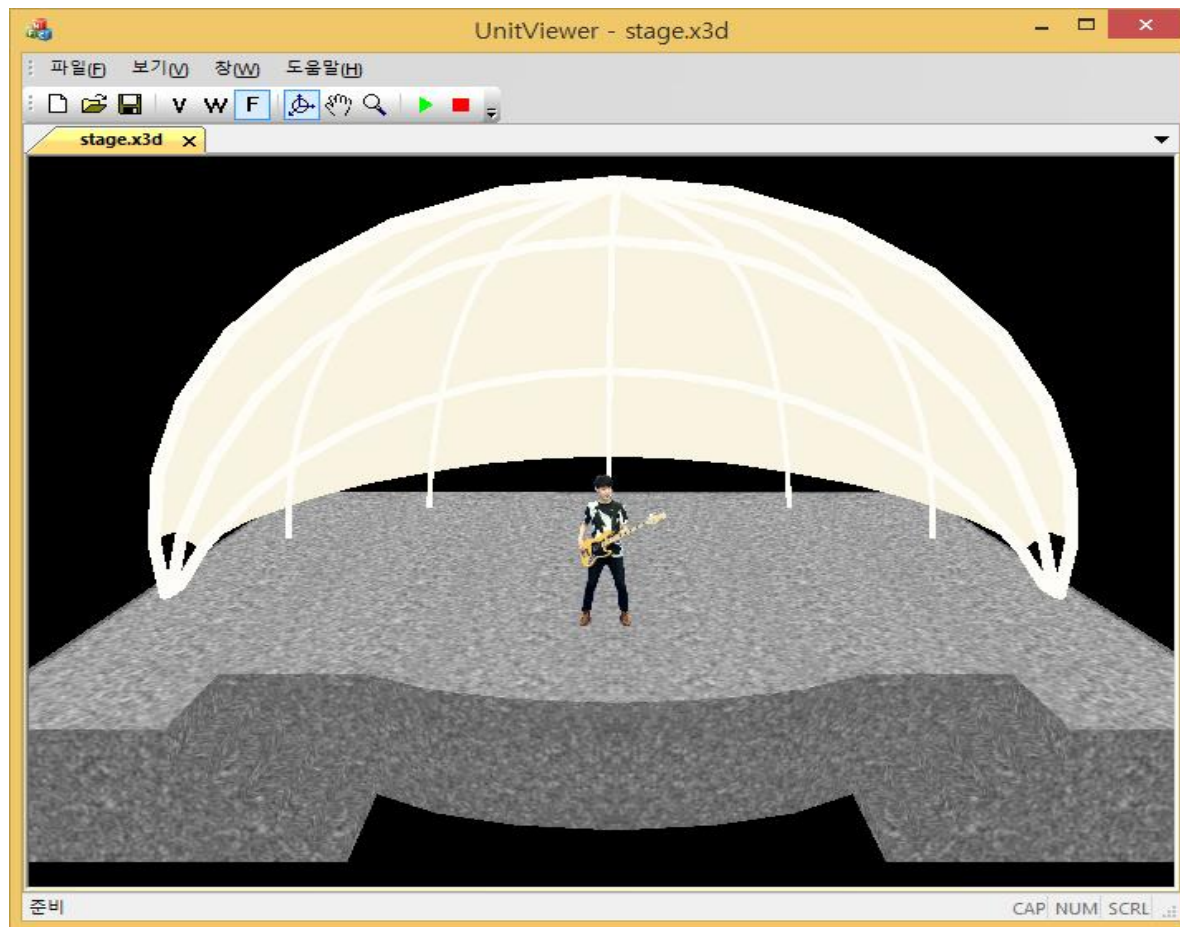
## Green Screen Video (2D)



# X3D Chroma Key Video Scene (3D)



# X3D Chroma Key Video Scene (3D)



## Location of Chromakey Image and Video

```
<Shape>  
  <IndexedFaceSet>  
    <Coordinate>  
    <TextureCoordinate>  
  </IndexedFaceSet>  
  <Appearance>  
    <MovieTexture>  
  </Appearance>  
</Shape>
```

## Location of Chromakey Image and Video

```
<Shape DEF='MovieShapeStandardDefinition'>
  <IndexedFaceSet DEF='Quadrilateral320x240' coordIndex='0 1 2 3'
    solid='false'>
    <Coordinate point='-29.53 -39.37 0 29.53 -39.37 0 -29.53 39.37 0 29.53
      39.37 0'/>
    <TextureCoordinate DEF='FullImageMapping' point='0 0 1 0 1 1 0 1'/>
  </IndexedFaceSet>
  <Appearance>
    <MovieTexture DEF='X3dQuipMovieStandardDefinition' repeatS='false'
      repeatT='false' url='base.avi'>
      <TextureProperties chromakey='true' minChromakeyColor='0 0 0.5
        0' maxChromakeyColor='0 0 1 0'/>
    </MovieTexture>
  </Appearance>
</Shape>
```

### 3 Terms, definitions, acronyms, and abbreviations

#### 3.1.17

#### **chroma-keying**

Chroma key chroma keying, is a visual effects technique for layering two images or video streams together based on color hues (chroma range). The technique has been used in many fields to remove a background from the subject of a photo or video. A color range in the foreground footage is made transparent, allowing separately filmed background footage or a static image to be inserted into the scene.

## 18 Texturing component

### 18.4.9 TextureProperties

TextureProperties : X3DNode

SFFloat [in,out] anisotropicDegree 1.0 [1,∞)  
SFColorRGBA [in,out] borderColor 0 0 0 0 [0,1]  
SFInt32 [in,out] borderWidth 0 [0,1]  
SFString [in,out] boundaryModeS "REPEAT" [see Table 18.7]  
SFString [in,out] boundaryModeT "REPEAT" [see Table 18.7]  
SFString [in,out] boundaryModeR "REPEAT" [see Table 18.7]  
SFString [in,out] magnificationFilter "FASTEST" [see Table 18.8]  
SFNode [in,out] metadata NULL [X3DMetadataObject]  
SFString [in,out] minificationFilter "FASTEST" [see Table 18.9]  
SFString [in,out] textureCompression "FASTEST" [see Table 18.10]  
SFFloat [in,out] texturePriority 0 [0,1]  
SFBool [] generateMipMaps FALSE  
**SFBool [in,out] chromakey FALSE**  
**SFColorRGBA [in,out] minChromakeyColor 0 0 0 0 [0,1]**  
**SFColorRGBA [in,out] maxChromakeyColor 0 0 0 0 [0,1]**



## 19775-1 IS Text Update

...

The chromakey field specifies the usage of chroma-keying image or video generation.

The minChromakeyColor specifies a minimum color value for a chroma-keying color.

The maxChromakeyColor specified a maximum color value for a chroma-keying color.

### **Example:**

```
<Shape DEF='MovieShapeStandardDefinition'>
  <IndexedFaceSet DEF='Quadrilateral320x240' coordIndex='0 1 2 3'
solid='false'>
    <Coordinate point='-29.53 -39.37 0 29.53 -39.37 0 -29.53 39.37 0 29.53
39.37 0'>
      <TextureCoordinate DEF='FullImageMapping' point='0 0 1 0 1 1 0
1'>
```

## 19775-1 IS Text Update

</IndexedFaceSet>

<Appearance>

<MovieTexture DEF='X3dQuipMovieStandardDefinition'  
repeatS='false' repeatT='false' url='base.avi'>

<TextureProperties chromakey='true' minChromakeyColor='0  
0 0.5 0' maxChromakeyColor='0 0 1 0'>

</MovieTexture>

</Appearance>

</Shape>

# Conclusion

- Chromakey image and video definition in X3D
- Chromakey definition
  - Solution (Case 1): Update TextureProperties node
    - Affect ImageTexture and MovieTexture nodes with TextureProperties
    - Affect chromakey parameters for ImageTexture and MovieTexture nodes
- Implementation
  - Chromakey X3D viewer: Visual C++, OpenGL, OpenCV
- 19775-1 IS text updates
  - 3 Terms, definitions, acronyms, and abbreviations
  - 18 Texturing component
- Future work
  - Web version