

CZ2003

Computer Graphics and Visualization

Lab 5 Report: Morphing

SSR2

U1821610C

Liew Zhi Li (Sherna)

Contents

[1 – Morphing 2](#_Toc24789796)

[2 – Code Overview 5](#_Toc24789797)

# 1 – Morphing

Given two parametric formulas that define some surfaces, we can create animated transformation called morphing. We can create a surface A that gradually morphs into surface B. In this lab, we will explore how it is done.

My index number in the lab attendance is 18.

According to the specified algorithm, my formula numbers will be:

formula\_number\_1 = 18

formula\_number\_2 = 20

|  |  |
| --- | --- |
| **Snapshot (Smooth)** | **Snapshot (Wireframe)** |
| Formula 18 | Formula 18 |
| Formula 20 | Formula 20 |

The following snapshots show how the morphing takes place. Alternatively, you can open the file lab5-official.wrl file to examine in greater detail.

|  |
| --- |
| **Snapshot (Smooth)** |
|  |
|  |
|  |
|  |

In order to perform this morphing, the parameters of the formulae should be in the same range for both surfaces. However, the original formulae given has different parameters and intervals. As such, we need to modify the formulae so that both parameters are .

|  |  |  |
| --- | --- | --- |
| **Formula 18 (Original)** | **Parameters (Original)** | **Image** |
|  |  |  |
| **Formula 18 (Modified)** | **Parameters (Modified)** |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Formula 20 (Original)** | **Parameters (Original)** | **Image** |
|  |  |  |
| **Formula 20 (Modified)** | **Parameters (Modified)** |
|  |  |

When I set the resolution of the morphing to [100 100], the quality of shape improved at the expense of the quality of the animation. The transition animation is jittery and less smooth.

As such, I have selected a resolution of [50 50] to achieve an ideal balance of the quality of shape and animation. This allows the scene to be rendered within 2 seconds.

When I set the cycleInterval to 2, the shape morphs much rapidly. When I set the the cycleInterval to 10, the shape morphs much slowly.

When I set loop to TRUE, the morphing animation is enabled. When I set loop to FALSE, the morphing animation is disabled.

# 2 – Code Overview

**#VRML V2.0 utf8**

# The following EXTERNPROTO definitins remain the same for all FVRML scenes.

EXTERNPROTO FGeometry [

    exposedField SFString definition

    exposedField MFFloat parameters

    exposedField MFInt32 resolution

    exposedField SFVec3f bboxCenter

    exposedField SFVec3f bboxSize

    exposedField SFString type

    exposedField MFNode parents

    exposedField SFVec2f timeSpan

] "http://www3.ntu.edu.sg/home/assourin/FVRML/FVRML.wrl#FGeometry"

EXTERNPROTO FMaterial [

    exposedField SFString diffuseColor

    exposedField SFString type

    exposedField MFColor patternColor

    exposedField MFFloat patternKey

    exposedField SFString ambientIntensity

    exposedField SFString emissiveColor

    exposedField SFString shininess

    exposedField SFString specularColor

    exposedField SFString transparency

    exposedField MFFloat parameters

    exposedField MFNode parents

    exposedField SFVec2f timeSpan

] "http://www3.ntu.edu.sg/home/assourin/FVRML/FVRML.wrl#FMaterial"

EXTERNPROTO FTexture3D [

    exposedField SFString definition

    exposedField SFString type

    exposedField MFFloat parameters

    exposedField MFNode parents

    exposedField SFVec2f timeSpan

] "http://www3.ntu.edu.sg/home/assourin/FVRML/FVRML.wrl#FTexture3D"

EXTERNPROTO FAppearance [

    exposedField SFNode material

    exposedField SFNode texture

    exposedField SFNode textureTransform

    exposedField SFNode texture3D

    exposedField MFNode parents

    eventIn SFBool refresh

] "http://www3.ntu.edu.sg/home/assourin/FVRML/FVRML.wrl#FAppearance"

EXTERNPROTO FShape [

    exposedField SFNode geometry

    exposedField SFNode appearance

    exposedField SFString polygonizer

    exposedField MFNode parents

    exposedField SFTime startTime

    exposedField SFTime stopTime

    exposedField SFTime cycleInterval

    exposedField SFBool loop

    exposedField SFBool enabled

    eventOut SFTime cycleTime

    eventIn SFFloat set\_fraction

    field SFInt32 frames

] "http://www3.ntu.edu.sg/home/assourin/FVRML/FVRML.wrl#FShape"

EXTERNPROTO FTransform [

    exposedField SFString operation

    exposedField SFString type

    exposedField SFString polygonizer

    exposedField MFFloat parameters

    exposedField SFString center

    exposedField SFString rotation

    exposedField SFString scale

    exposedField SFString scaleOrientation

    exposedField SFString translation

    exposedField MFNode children

    exposedField MFNode parents

    eventIn SFBool refresh

    exposedField SFVec2f timeSpan

    exposedField SFTime startTime

    exposedField SFTime stopTime

    exposedField SFTime cycleInterval

    exposedField SFBool loop

    exposedField SFBool enabled

    eventOut SFTime cycleTime

    eventIn SFFloat set\_fraction

    field SFInt32 frames

] "http://www3.ntu.edu.sg/home/assourin/FVRML/FVRML.wrl#FTransform"

Viewpoint {

    position 0 0 12

    orientation 0 0 -1 0

    description "view all"

}

Background {skyColor 0.75 0.75 0.75 }

# External VRML object "coordinate system" is included in the scene

Inline {url "http://www3.ntu.edu.sg/home/assourin/FVRML/CoordinateAxes.wrl"}

## the following codes for shape 1

Transform{

    translation -4.2 0 0

    children[

        FShape {

            geometry FGeometry {

                resolution [50 50]

                parameters [0 1 0 1]

                definition "

                function parametric\_x(u,v,w,t)

                { x1=(cos(2\*pi\*u) +1)\*(cos(2\*pi\*v));

                return x1;        }

                function parametric\_y(u,v,w,t)

                { y1=sin(2\*pi\*u)\*cos(2\*pi\*v);

                return y1;       }

                function parametric\_z(u,v,w,t)

                { z1=sin(2\*pi\*v);

                return z1;        }"

            }

            appearance FAppearance {

                material FMaterial {

                    diffuseColor "r=abs(cos(u\*pi)); g=0; b=1;"

                }

            }

        }

    ]

}

Transform{

    translation -5.5  -1.6 0

    scale 0.3 0.3 0.3

    children[

        Shape { geometry Text { string [ "Shape 1 - Formula 18" ]}

            appearance Appearance {

                material Material {

                    diffuseColor 0 0 0

                }

            }}

    ]

}

## the following codes for the animated surface

FShape {

    # Enabling cycled animation

    loop TRUE

    # Mapping the interval of the internal time t=[0,1] to the actual time in sec.

    cycleInterval 7

    geometry FGeometry {

        resolution [50 50]

        parameters [0 1 0 1]

        # Definition of the animated linear transformation (morphing)

        # of one surface defined by     x1(u,v), y1(u,v), z1(u,v)

        # to another surface defined by x2(u,v), y2(u,v), z2(u,v)

        definition "

        function parametric\_x(u,v,w,t)

        { x1=(cos(2\*pi\*u)+1)\*(cos(2\*pi\*v));

        x2=0.2\*2\*pi\*u\*sin(2\*pi\*v);

        return x1+(x2-x1)\*t;        }

        function parametric\_y(u,v,w,t)

        { y1=sin(2\*pi\*u)\*cos(2\*pi\*v);

        y2=0.2\*2\*pi\*v\*sin(2\*pi\*u);

        return y1+(y2-y1)\*t;       }

        function parametric\_z(u,v,w,t)

        { z1=sin(2\*pi\*v);

        z2=0.2\*2\*pi\*v\*cos(2\*pi\*u);

        return z1+(z2-z1)\*t;        }"

    }

    appearance FAppearance {

        material FMaterial {

            diffuseColor "r=abs(cos(u\*pi)); g=0; b=1;"

        }    }

}

Transform{

    translation -0.5  -1.6 0

    scale 0.3 0.3 0.3

    children[

        Shape { geometry Text { string [ "Morphing" ]}

            appearance Appearance {

                material Material {

                    diffuseColor 0 0 0

                }

            }}

    ]

}

## the following codes for shape 2

Transform{

    translation 3.2 0 0

    children[

        FShape {

            geometry FGeometry {

                resolution [50 50]

                parameters [0 1 0 1]

                definition "

                function parametric\_x(u,v,w,t)

                { x2=0.2\*2\*pi\*u\*sin(2\*pi\*v);

                return x2;        }

                function parametric\_y(u,v,w,t)

                { y2=0.2\*2\*pi\*v\*sin(2\*pi\*u);

                return y2;       }

                function parametric\_z(u,v,w,t)

                { z2=0.2\*2\*pi\*v\*cos(2\*pi\*u);

                return z2;        }"

            }

            appearance FAppearance {

                material FMaterial {

                    diffuseColor "r=abs(cos(u\*pi)); g=0; b=1;"

                }

            }

        }

    ]

}

Transform{

    translation 2.5  -1.6 0

    scale 0.3 0.3 0.3

    children[

        Shape {

            geometry Text {

                string [ "Shape 2 - Formula 20" ]

            }

            appearance Appearance {

                material Material {

                    diffuseColor 0 0 0

                }

            }

        }

    ]

}