

Information Visualization

W10: Exercise - Data Model and Transfer Function

Graduation School of System Informatics

Department of Computational Science

Naohisa Sakamoto, Akira Kageyama

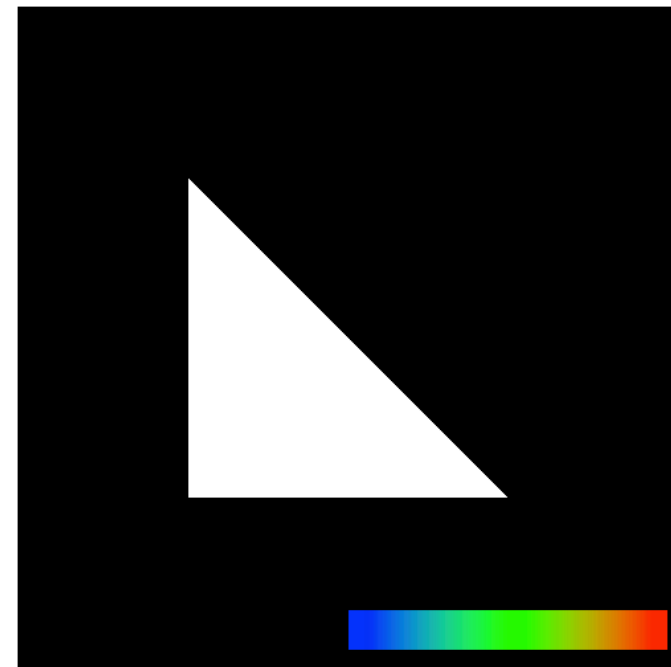
May.17, 2017

Schedule

- W01 4/11 Guidance
- W02 4/12 Setup
- W03 4/18 Introduction to Data Visualization
- W04 4/19 CG Programming
- W05 4/25 Rendering Pipeline
- W06 4/26 Coordinate Systems and Transformations
- W07 5/09 Shading
- W08 5/10 Shader Programming
- W09 5/16 Visualization Pipeline
- W10 5/17 Data Model and Transfer Function
- W11 5/23 Scalar Data Visualization 1 (Isosurface Extraction)
- W12 5/24 Implementation of Isosurface Extraction
- W13 5/30 Scalar Data Visualization 2 (Volume Rendering)
- W14 5/31 Implementation of Volume Rendering
- W15 6/06 Student Presentations

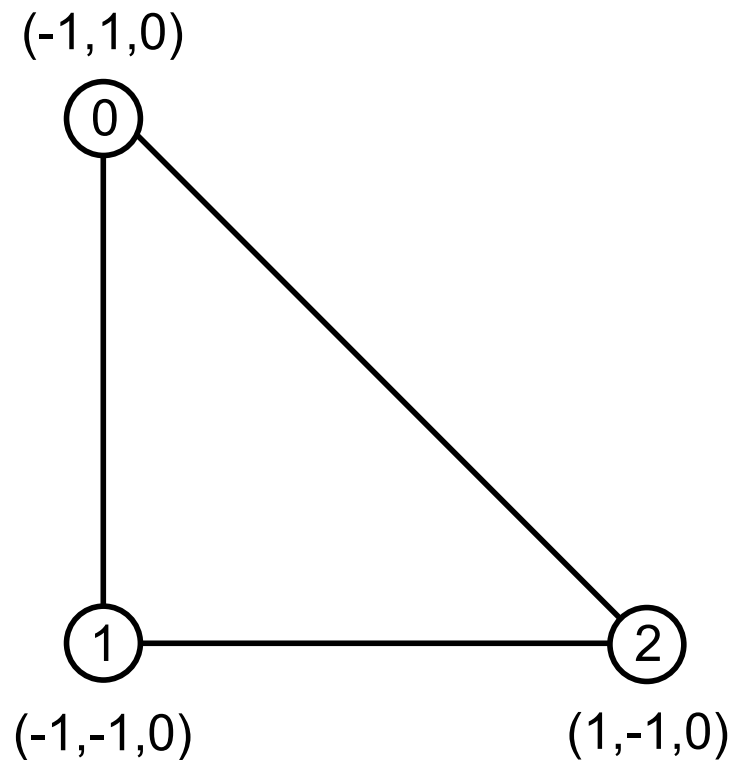
Ex01: Rainbow color map

- Draw a white triangle and a rainbow color map on the scene.
 - Download
 - `w10_main_ex01.js`
 - `w10_index_ex01.html`
 - Open
 - `w10_index_ex01.html`



Ex01: Rainbow color map

- White triangle



```
var vertices = [  
    [-1, 1, 0], // v0  
    [-1, -1, 0], // v1  
    [ 1, -1, 0]  // v2  
];  
  
var faces = [  
    [0, 1, 2] // f0: v0-v1-v2  
];
```

Ex01: Rainbow color map

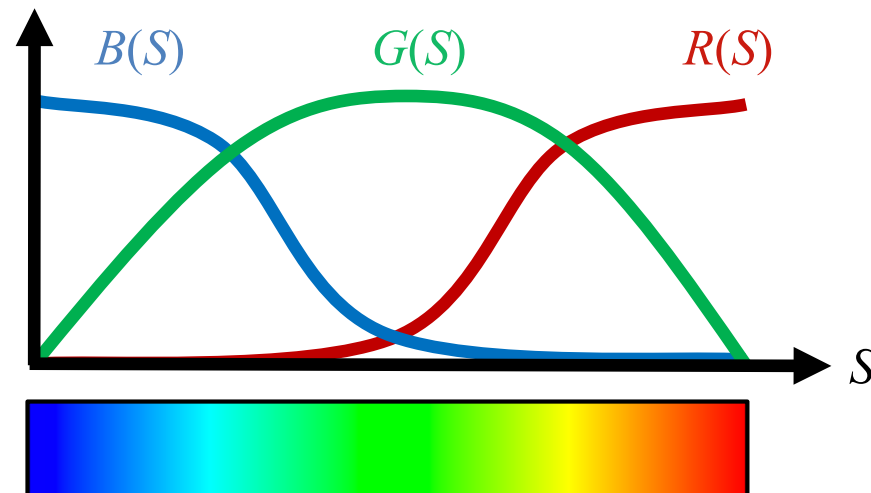
- White triangle

```
material.vertexColors = THREE.FaceColors;

var nfaces = faces.length;
for ( var i = 0; i < nfaces; i++ )
{
    geometry.faces[i].color = new THREE.Color( 1, 1, 1 );
}
```

Ex01: Rainbow color map

- Create the rainbow color map
 - Cosine functions for each color channel



Ex01: Rainbow color map

- Create the rainbow color map

```
var cmap = [];  
for ( var i = 0; i < 256; i++ )  
{  
    var S = i / 255.0; // [0,1]  
    var R = Math.max( Math.cos( ( S - 1.0 ) * Math.PI ), 0.0 );  
    var G = Math.max( Math.cos( ( S - 0.5 ) * Math.PI ), 0.0 );  
    var B = Math.max( Math.cos( S * Math.PI ), 0.0 );  
    var color = new THREE.Color( R, G, B );  
    cmap.push( [ S, '0x' + color.getHexString() ] );  
}
```

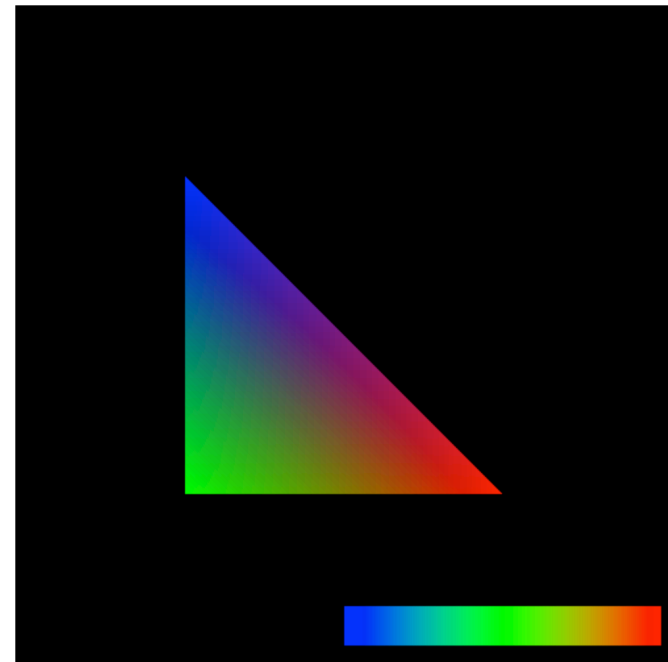
Ex01: Rainbow color map

- Draw the color map with THREE.Lut
 - <http://mrdoob.github.io/three.js/examples/js/math/Lut.js>

```
var lut = new THREE.Lut( 'rainbow', cmap.length );
lut.addColorMap( 'mycolormap', cmap );
lut.changeColorMap( 'mycolormap' );
scene.add( lut.setLegendOn( {
    'layout': 'horizontal',
    'position': { 'x': 0.6, 'y': -1.1, 'z': 2 },
    'dimensions': { 'width': 0.15, 'height': 1.2 }
} ) );
```

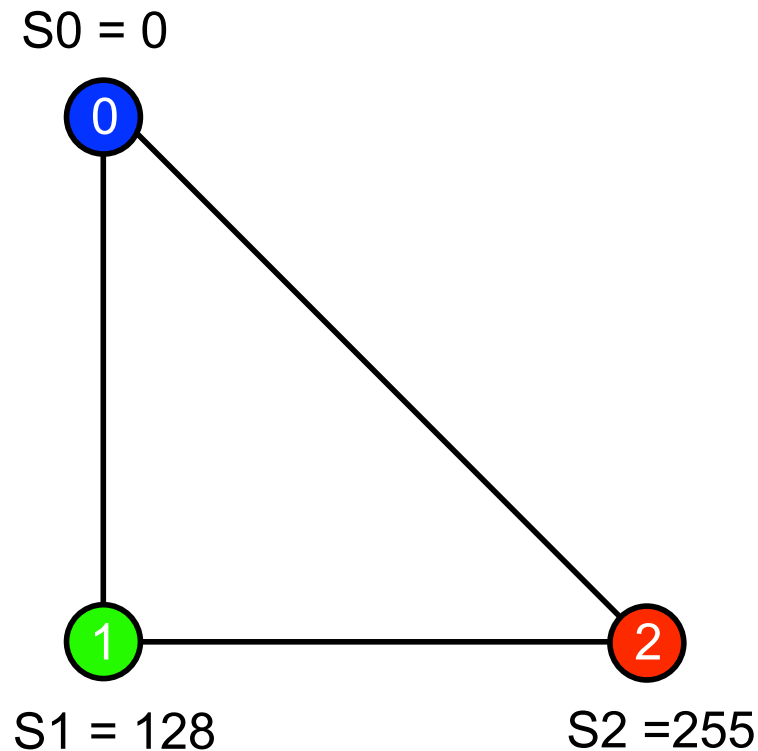

Ex02: Colored triangle geometry

- Draw a triangle geometry colored by the rainbow color map.
 - Download
 - `w01_main_ex02.js`
 - `w10_index_ex02.html`
 - Open
 - `w10_index_ex02.html`

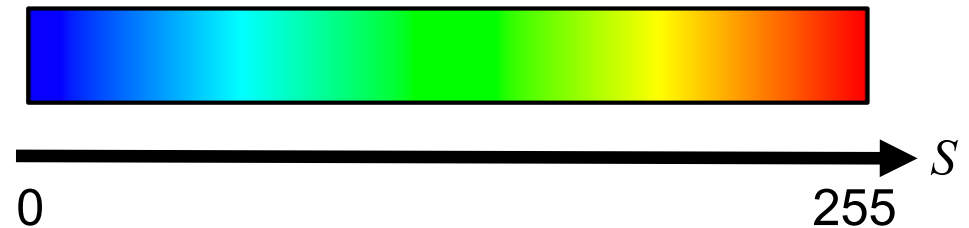


Ex02: Colored triangle geometry

- Triangle with scalar values.



```
var scalars = [  
    0,      // S0  
    128,    // S1  
    255     // S2  
];
```



Ex02: Colored triangle geometry

- Assign the colors for each vertex using the rainbow color map.

```
material.vertexColors = THREE.VertexColors;
for ( var i = 0; i < nfaces; i++ )
{
    var id = faces[i];
    var S0 = scalars[ id[0] ];
    var S1 = scalars[ id[1] ];
    var S2 = scalars[ id[2] ];
    var C0 = new THREE.Color().setHex( cmap[ S0 ][1] );
    var C1 = new THREE.Color().setHex( cmap[ S1 ][1] );
    var C2 = new THREE.Color().setHex( cmap[ S2 ][1] );
    geometry.faces[i].vertexColors.push( C0 );
    geometry.faces[i].vertexColors.push( C1 );
    geometry.faces[i].vertexColors.push( C2 );
}
```

Task 1

- Modify the scalar values and then draw the triangle with the rainbow color map.

$S_0 = 0.1$

0

1

$S_1 = 0.2$

2

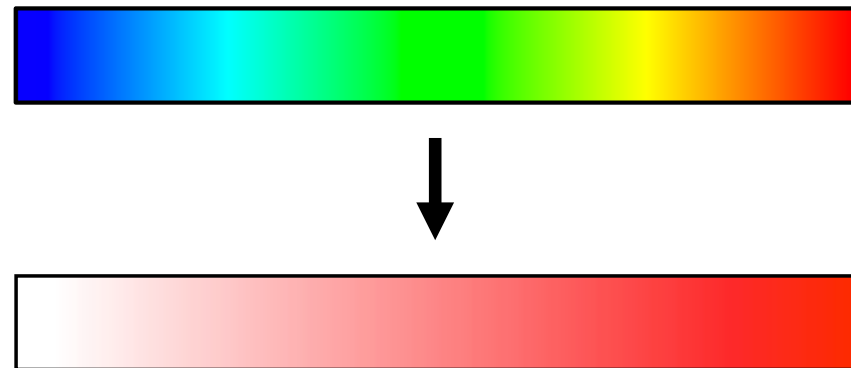
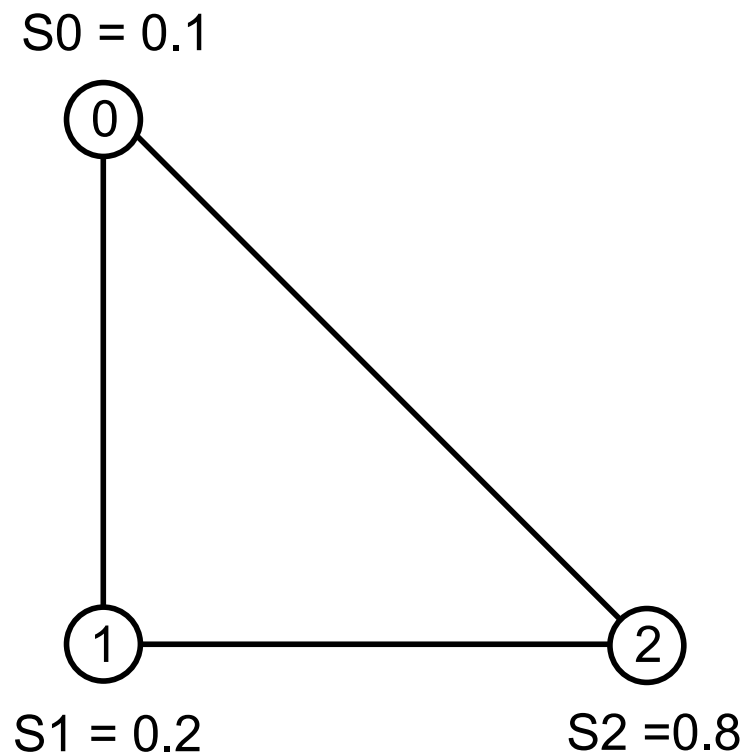
$S_2 = 0.8$

```
var scalars = [  
    0.1,    // S0  
    0.2,    // S1  
    0.8     // S2  
];
```



Task 2

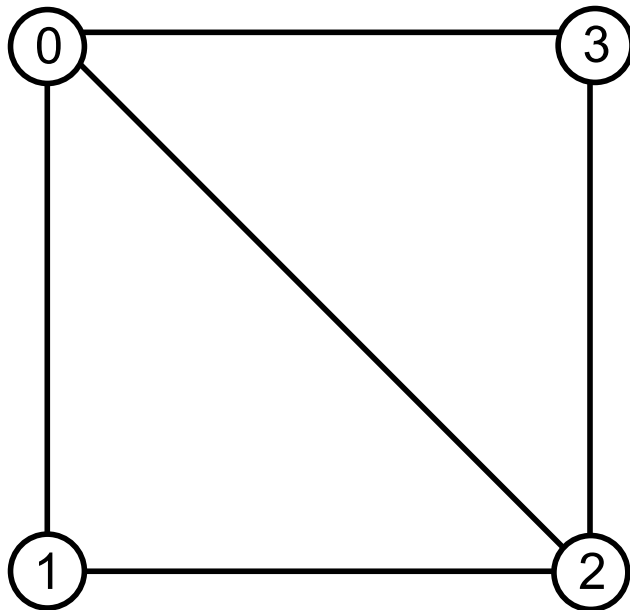
- Change the rainbow color map to the following white-red color map.



Task 3

- Draw the following square composed of two triangles with the color map.

S0 = 0.1



S1 = 0.2

S2 = 0.8

```
var scalars = [  
    0.1,    // S0  
    0.2,    // S1  
    0.8,    // S2  
    0.5     // S3  
];
```

Polling

- Take the poll
 - Student ID Number
 - Name
 - URL to Task 1
 - URL to Task 2
 - URL to Task 3