Information Visualization

W08: Exercise - Data Model and Transfer Function

Graduation School of System Informatics

Department of Computational Science

Naohisa Sakamoto, Akira Kageyama

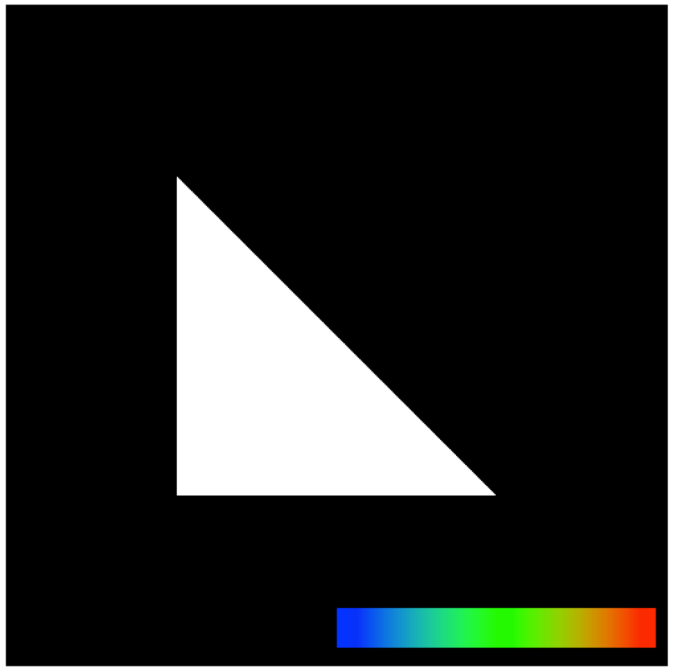
May.2, 2018

Schedule

- W01 4/10 Guidance
- W02 4/11 Exercise (Setup)
- W03 4/17 Introduction to Data Visualization
- W04 4/18 Exercise (JavaScript Programming)
- W05 4/24 Computer Graphics
- W06 4/25 Exercise (Shader Programming)
- W07 5/01 Visualization Pipeline
- W08 5/02 Exercise (Data Model and Transfer Function)
- W09 5/08 Volume Visualization
- W10 5/09 Exercise (Isosurfaces and Volume Rendering)
- W11 5/22 Flow Visualization
- W12 5/23 Exercise (Streamlines and Line Integral Convolution)
- W13 5/29 Workshops 1
- W14 5/30 Workshops 2
- W15 6/05 Presentations

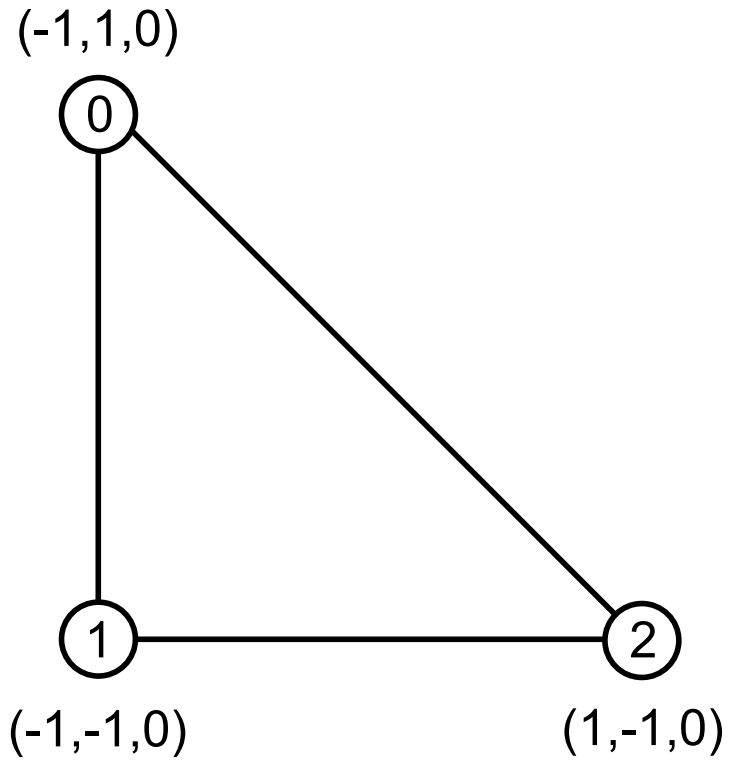
Ex01: Rainbow color map

- Draw a white triangle and a rainbow color map on the scene.
  - Download
    - w08_main_ex01.js
    - w08_index_ex01.html
  - Open
    - w08_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

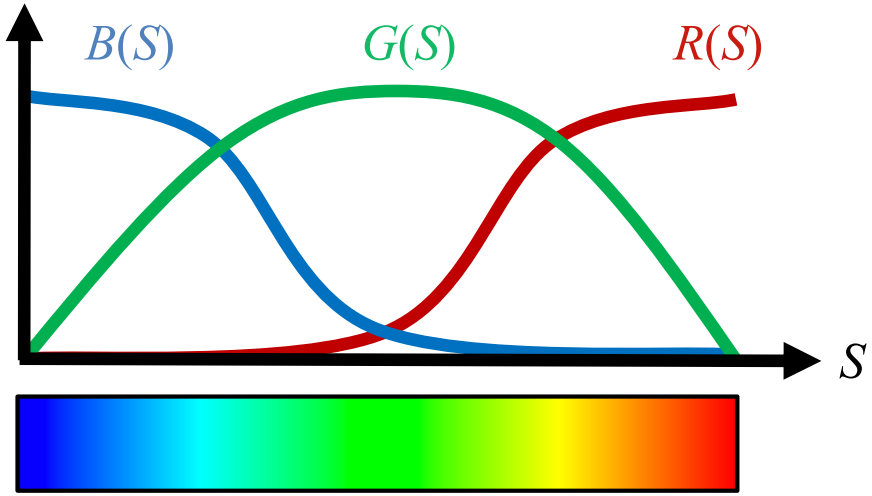
- White triangle

```
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() ] );
}
```

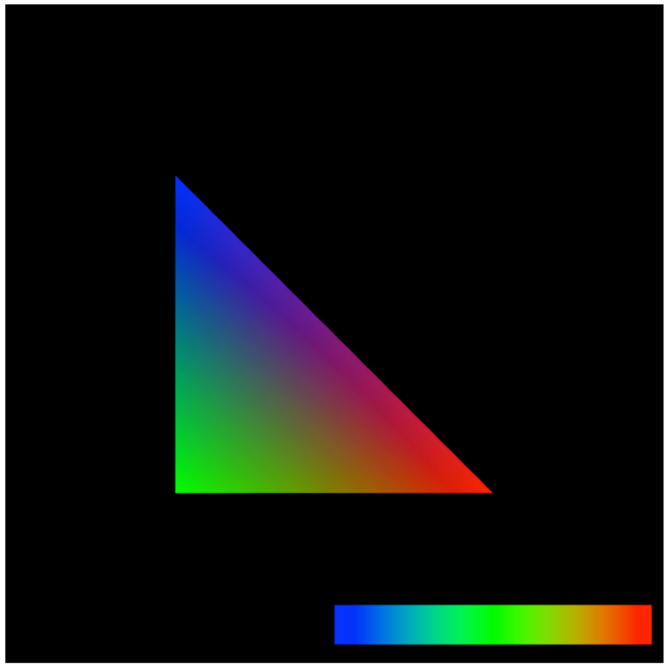
Ex01: Rainbow color map

- Draw the color map with THREE.Lut
  - https://threejs.org/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 }
} ) );
```

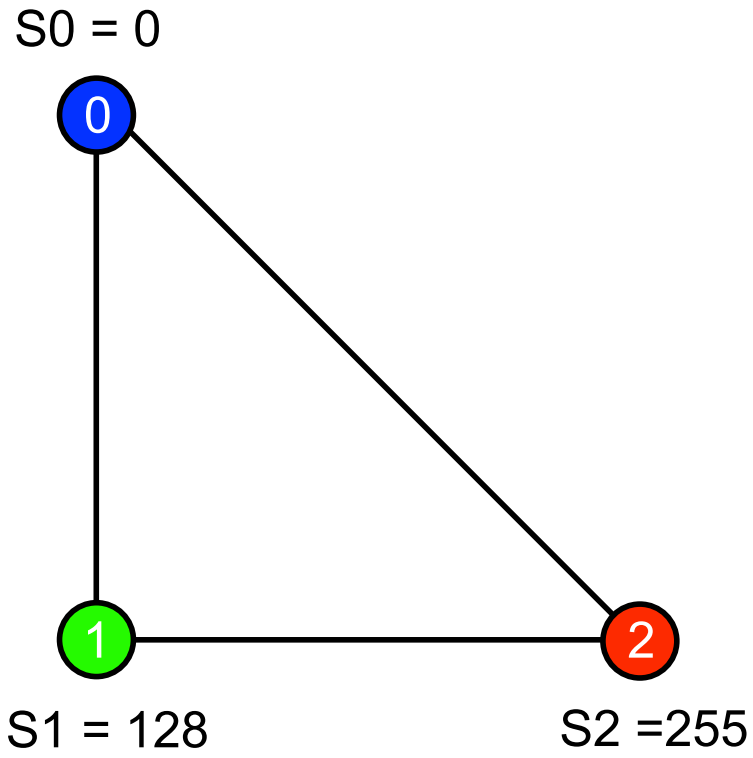
Ex02: Colored triangle geometry

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

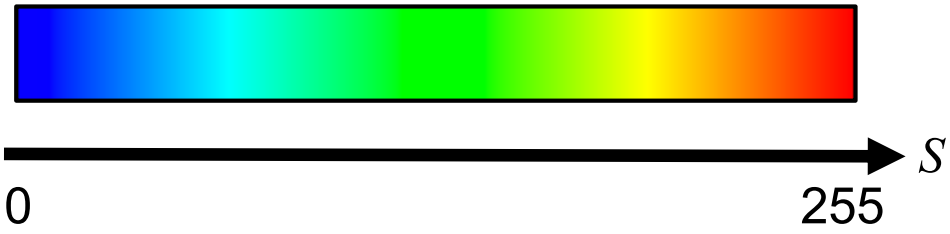


Ex02: Colored triangle geometry

- Triangle with scalar values.




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



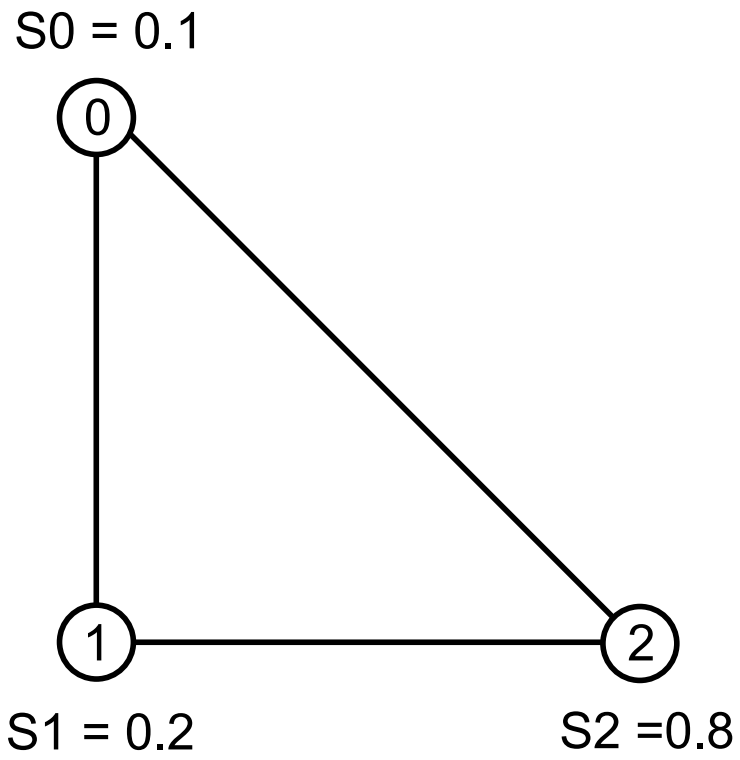
Ex02: Colored triangle geometry

- 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

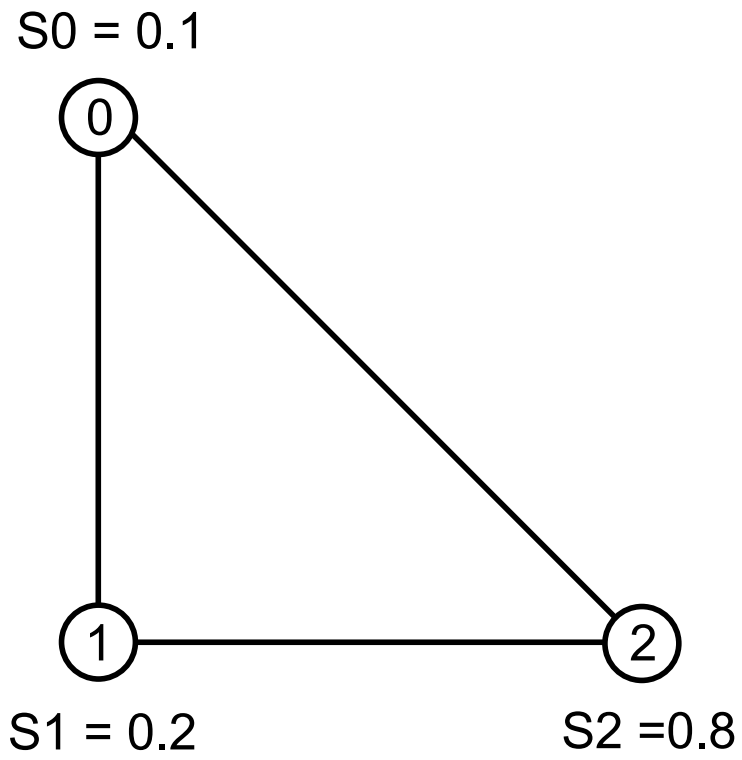
- Modify the scalar values and then draw the triangle with the rainbow color map.

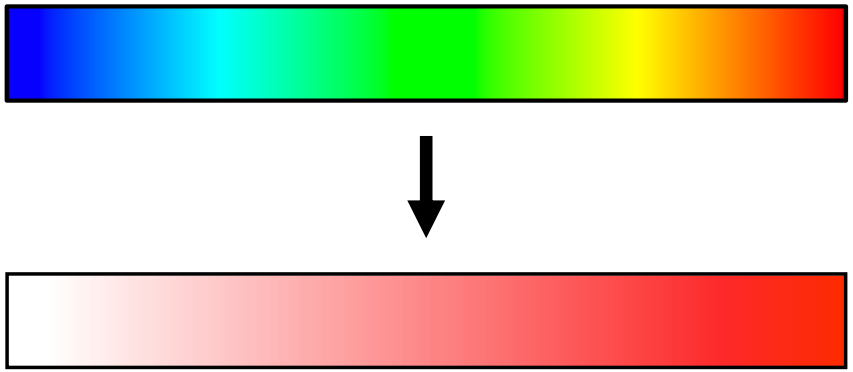



```
var scalars = [
    0.1,    // S0
    0.2,    // S1
    0.8     // S2
];
```

Task 2

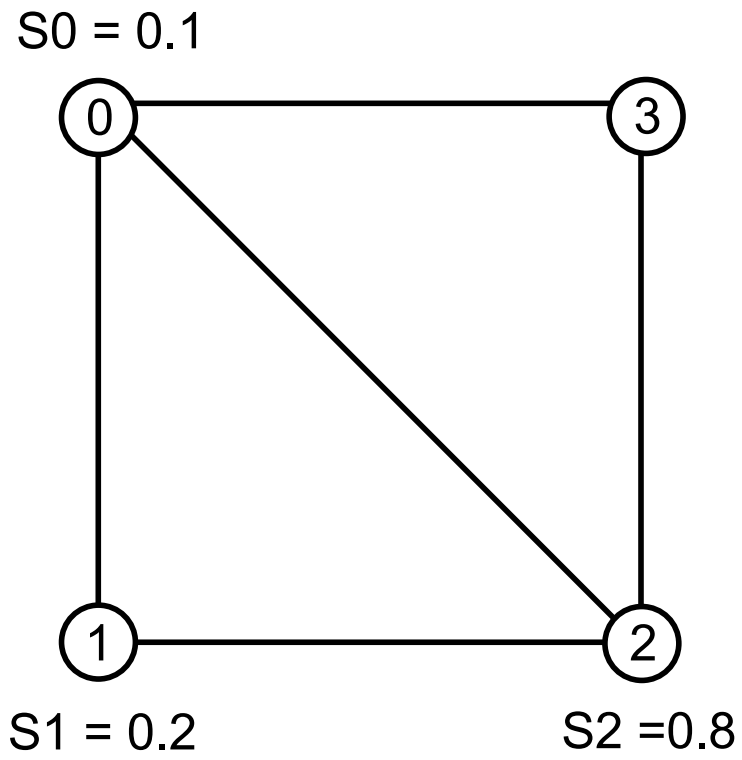
- Change the rainbow color map to the following white-red color map.





Task 3

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




```
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