アドバンストCG　第一回レポート課題

# 実行環境

1. OS

macOS Big Sur version11.2.3

1. プログラム実行時に表示される文字列情報

OpenGL version: 4.1 ATI-4.2.15

GLSL version: 4.10

Vendor: ATI Technologies Inc.

Renderer: AMD Radeon Pro 5300M OpenGL Engine

# 実行結果

## 課題01

1. GLSLのソースコード

[scene01\_chcker2D.frag]

#version 150 core

in vec2 outTexCoord;

out vec4 fragColor;

// TODO: uncomment these lines

uniform vec4 checkerColor0;

uniform vec4 checkerColor1;

uniform vec2 checkerScale;

void main()

{

// TODO: write an appropriate code here

float s\_ration = mod(outTexCoord.s, checkerScale.s)/checkerScale.s;

float t\_ration = mod(outTexCoord.t, checkerScale.t)/checkerScale.t;

if((s\_ration < 0.5f && t\_ration < 0.5f) || (s\_ration >= 0.5 && t\_ration >= 0.5)){

fragColor = checkerColor0;

} else {

fragColor = checkerColor1;

}

}

[scene01\_checker2D.vert]

#version 150 core

in vec4 vertexPosition;

in vec2 inTexCoord;

out vec2 outTexCoord;

void main()

{

// TODO: write an appropriate code here

// gl\_Positon: バーテックスシェーダの出力変数

gl\_Position = vertexPosition;

outTexCoord = inTexCoord;

}

1. 実行結果のスクリーンキャプチャ画像

グラフィカル ユーザー インターフェイス が含まれている画像

自動的に生成された説明

## 課題02

1. GLSLのソースコード

[scene02\_image\_smoothing.frag]

#version 150 core

in vec2 outTexCoord;

out vec4 fragColor;

// TODO: uncomment these lines

uniform sampler2D tex;

uniform int halfKernelSize;

uniform float uScale;

uniform float vScale;

void main()

{

// TODO: write an appropriate code here

// change from texture2D() to texture()

// https://stackoverflow.com/questions/26266198/glsl-invalid-call-of-undeclared-identifier-texture2d

if(halfKernelSize == 0){

fragColor = texture(tex, outTexCoord);

}else {

float sum\_weight = 0; // ガウス関数で計算した重みの総和

for(int i = -halfKernelSize; i <= halfKernelSize; i++){

for(int j = -halfKernelSize; j<= halfKernelSize; j++){

// ガウス関数の重みを計算

float x = i / halfKernelSize, y = j / halfKernelSize;

float d = x \* x + y \* y;

float weight = exp(-d/(0.5f\*0.5f));

vec2 targetTexCoord = vec2(outTexCoord.s + uScale\*i, outTexCoord.t + vScale\*j);

fragColor += weight \* texture(tex, targetTexCoord);

sum\_weight += weight;

}

}

fragColor /= sum\_weight;

}

}

[scene02\_image\_smoothing.vert]

#version 150 core

in vec4 vertexPosition;

in vec2 inTexCoord;

out vec2 outTexCoord;

void main()

{

// TODO: write an appropriate code here

gl\_Position = vertexPosition;

outTexCoord = inTexCoord;

}

1. 実行結果のスクリーンキャプチャ画像

白いバックグラウンドの前に座っている猫

自動的に生成された説明[Half Kernel = 0の場合]

[Half Kernel = 10の場合]

猫のミーム

低い精度で自動的に生成された説明

[Half Kernel = 30の場合]

グラフィカル ユーザー インターフェイス

自動的に生成された説明

## 課題03

1. GLSLのソースコード

[scene03\_wave\_animation.frag]

#version 150 core

out vec4 fragColor;

// TODO: uncomment these lines

uniform vec4 lineColor;

void main()

{

// TODO: write an appropriate code here

fragColor = lineColor;

}

[scene03\_wave\_animation.vert]

#version 150 core

in vec4 vertexPosition;

// TODO: uncomment these lines

uniform float temporalSignal;

uniform mat4 projModelViewMatrix;

void main()

{

// TODO: write an appropriate code here

float initY = (sin(vertexPosition.x\*2)+cos(vertexPosition.z\*2))\*0.8f;

float scale = (cos(vertexPosition.x\*2)+sin(vertexPosition.z\*2))/2.0f\*cos(temporalSignal\*4);

vec4 animVPosition = vec4(vertexPosition.x, initY\*scale, vertexPosition.z, vertexPosition.w);

gl\_Position = projModelViewMatrix \* animVPosition;

}

1. 実行結果のスクリーンキャプチャ画像

グラフィカル ユーザー インターフェイス

自動的に生成された説明グラフィカル ユーザー インターフェイス が含まれている画像

自動的に生成された説明グラフィカル ユーザー インターフェイス

自動的に生成された説明

## 課題04

1. GLSLのソースコード

[scene04\_pseudo\_normal.frag]

#version 150 core

in vec3 vVertexNormal;

out vec4 fragColor;

void main()

{

// TODO: write an appropriate code here

fragColor = vec4(0.5f \* vVertexNormal + 0.5f, 1.f);

}

[scene04\_pseudo\_normal.vert]

#version 150 core

in vec4 vertexPosition;

in vec3 vertexNormal;

out vec3 vVertexNormal;

uniform mat4 modelViewMatrix;

uniform mat4 projMatrix;

// TODO: uncomment these lines

uniform mat3 modelViewInvTransposed;

void main()

{

// TODO: write an appropriate code here

gl\_Position = projMatrix \* modelViewMatrix \* vertexPosition;

vVertexNormal = normalize(modelViewInvTransposed \* vertexNormal);

}

1. 実行結果のスクリーンキャプチャ画像

グラフィカル ユーザー インターフェイス

自動的に生成された説明

## 課題05

1. GLSLのソースコード

[scene05\_envmap.frag]

#version 150 core

#define PI 3.141592653589793

in vec3 vWorldEyeDir;

in vec3 vWorldNormal;

out vec4 fragColor;

// TODO: uncomment these lines

uniform sampler2D envmap;

float atan2(in float y, in float x)

{

return x == 0.0 ? sign(y)\*PI/2 : atan(y, x);

}

void main()

{

// TODO: write an appropriate code here

// 正反射ベクトルを計算

vec3 vWorldRefDir = normalize(reflect(vWorldEyeDir, vWorldNormal));

// テクスチャ座標の計算

float theta = asin(vWorldRefDir.y);

float psi = atan2(vWorldRefDir.z, vWorldRefDir.x);

// 角度を正規化し、テクスチャ座標に変換

float normalTheta = (theta + PI/2.f) / PI;

float normalPsi = psi / PI;

fragColor = texture(envmap, vec2(normalPsi, normalTheta));

}

[scene05\_envmap.vert]

#version 150 core

in vec4 vertexPosition;

in vec3 vertexNormal;

out vec3 vWorldEyeDir;

out vec3 vWorldNormal;

uniform mat4 projModelViewMatrix;

// TODO: uncomment these lines

uniform vec3 eye;

void main()

{

// TODO: write an appropriate code here

gl\_Position = projModelViewMatrix \* vertexPosition;

vWorldEyeDir = normalize(vec3(vertexPosition) - eye);

vWorldNormal = vertexNormal;

}

1. 実行結果のスクリーンキャプチャ画像

男, 座る, 空気, 民衆 が含まれている画像

自動的に生成された説明