

# CS-562 Project2 Assignment Report

jinyun.choi / Jinyun Choi

For this assignment, I mainly implemented Moment Shadow Map, and Convolution filter with a Gaussian weight kernel.

This is the instruction for each feature's implemented code files or line numbers.

1. Moment Shadow Map:

- SampleScene.cpp with 382~389 line(CreateShader), 526~559 line(UpdateShadow), 120~133 line (sending shadow map and matrix data)
- In shaders file - shadow.vert, shadow.frag, lightStage.frag with 107~160 line (cholesky, CaluateShadow)

2. Convolution filter with a Gaussian weight kernel:

- SampleScene.cpp with 393~420 line (Create Shader), 325~344 line (UpdateGaussianWeight), 561~605 line(UpdateBlur)
- In shader file - computeVertical.comp, computeHorizontal.comp

+ ) It takes a little bit taking time to start the project. Because I used a high polygon obj file of the Bunny model.

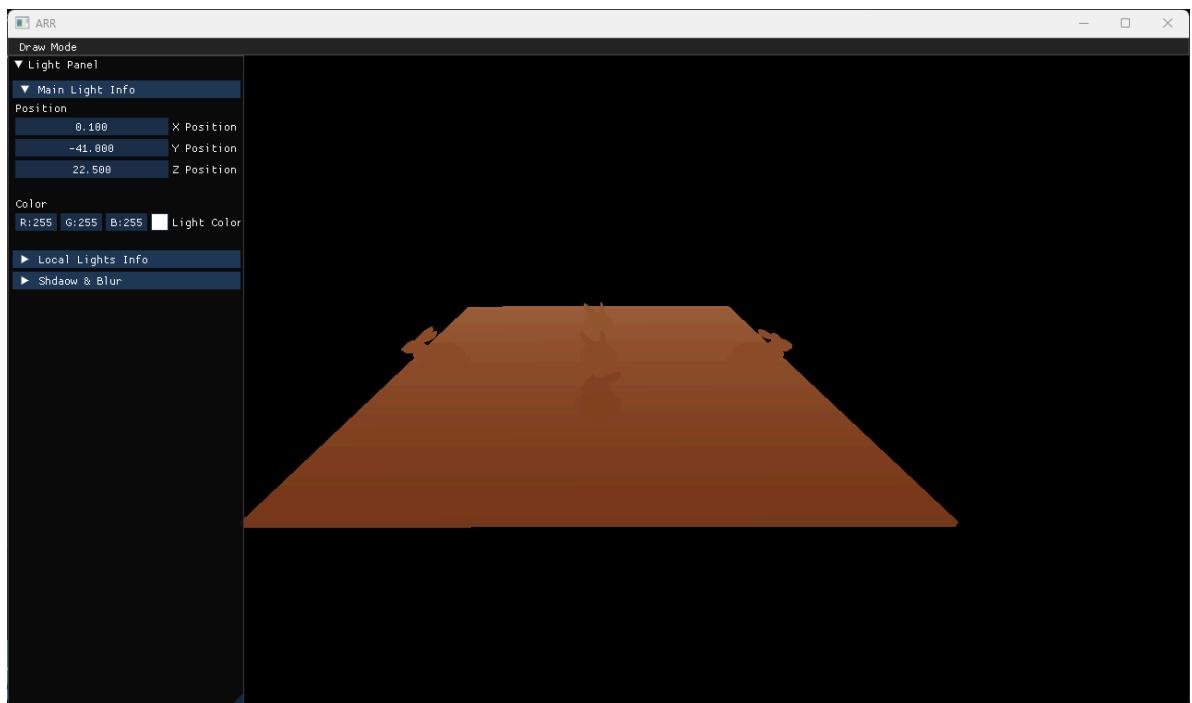
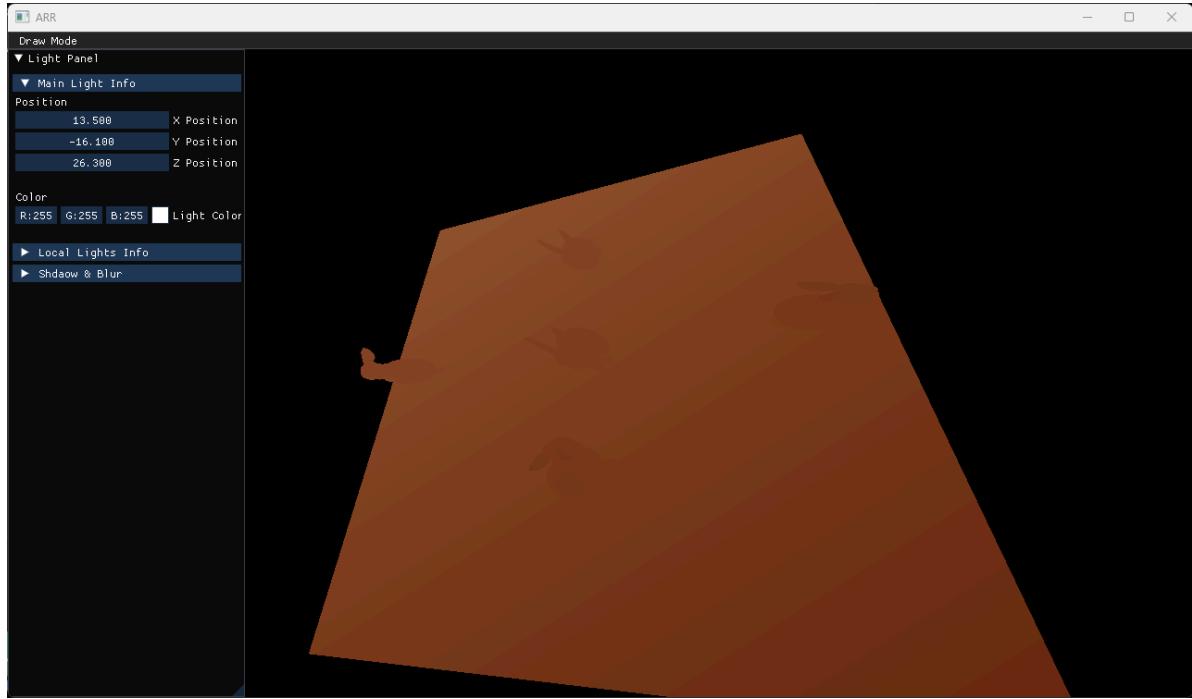
+ ) I set only the x64 version.

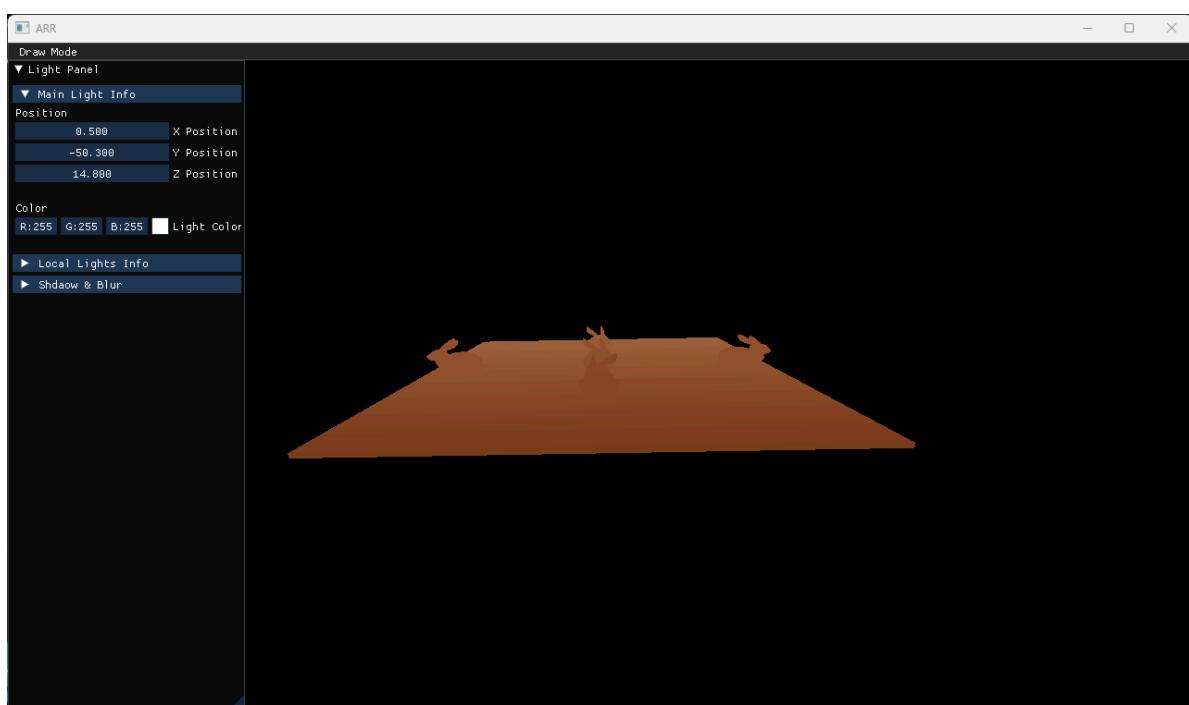
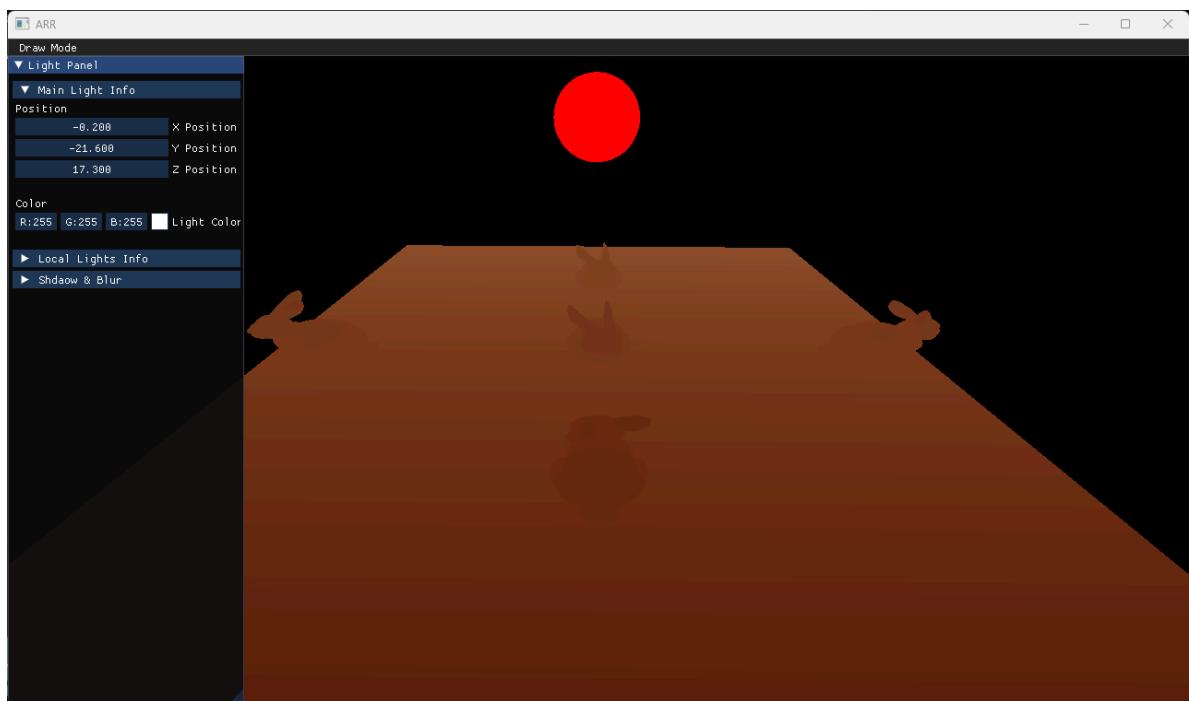
## Implementations:

- Moment Shadow Map
- Convolution filter with a Gaussian weight kernel

## Moment Shadow Map:

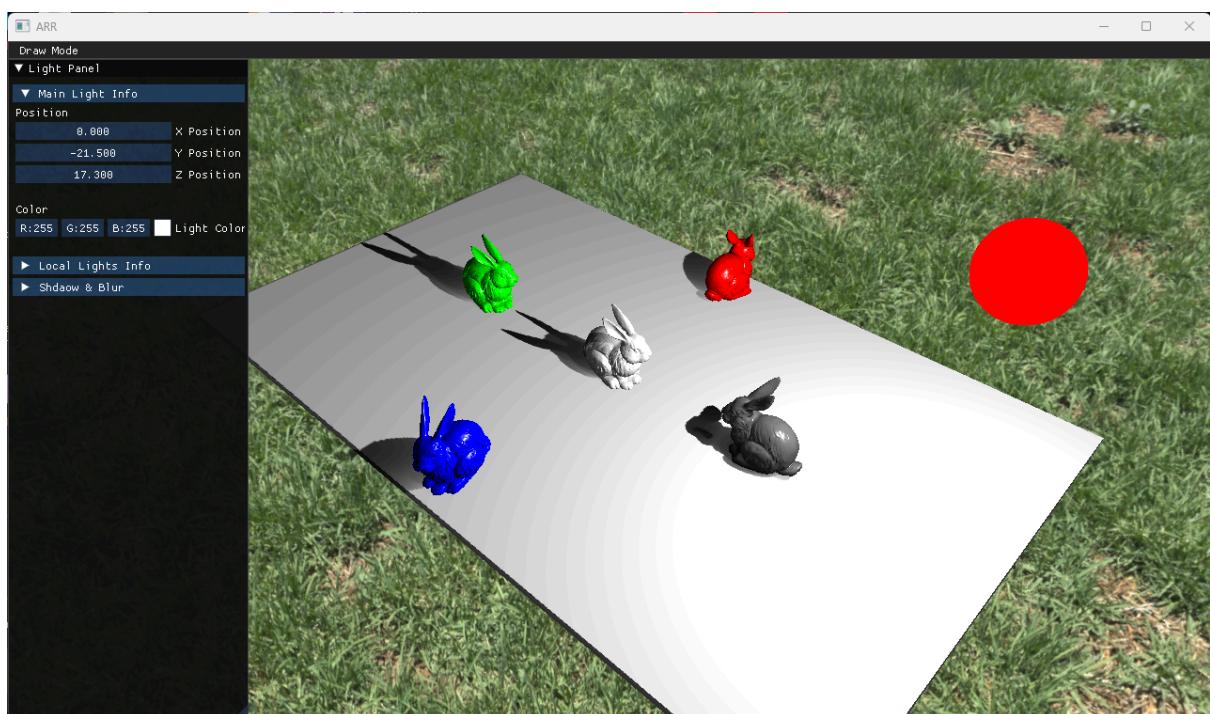
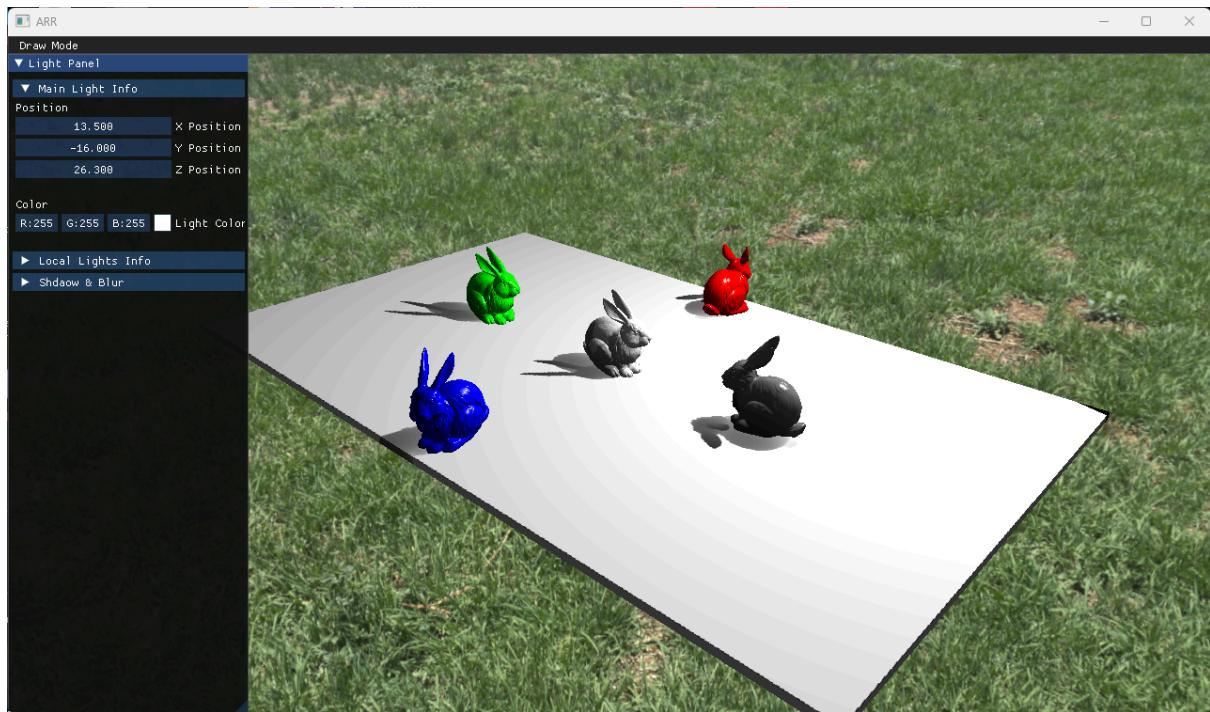
### 1. Shadow Map (with out blur, only MSM)

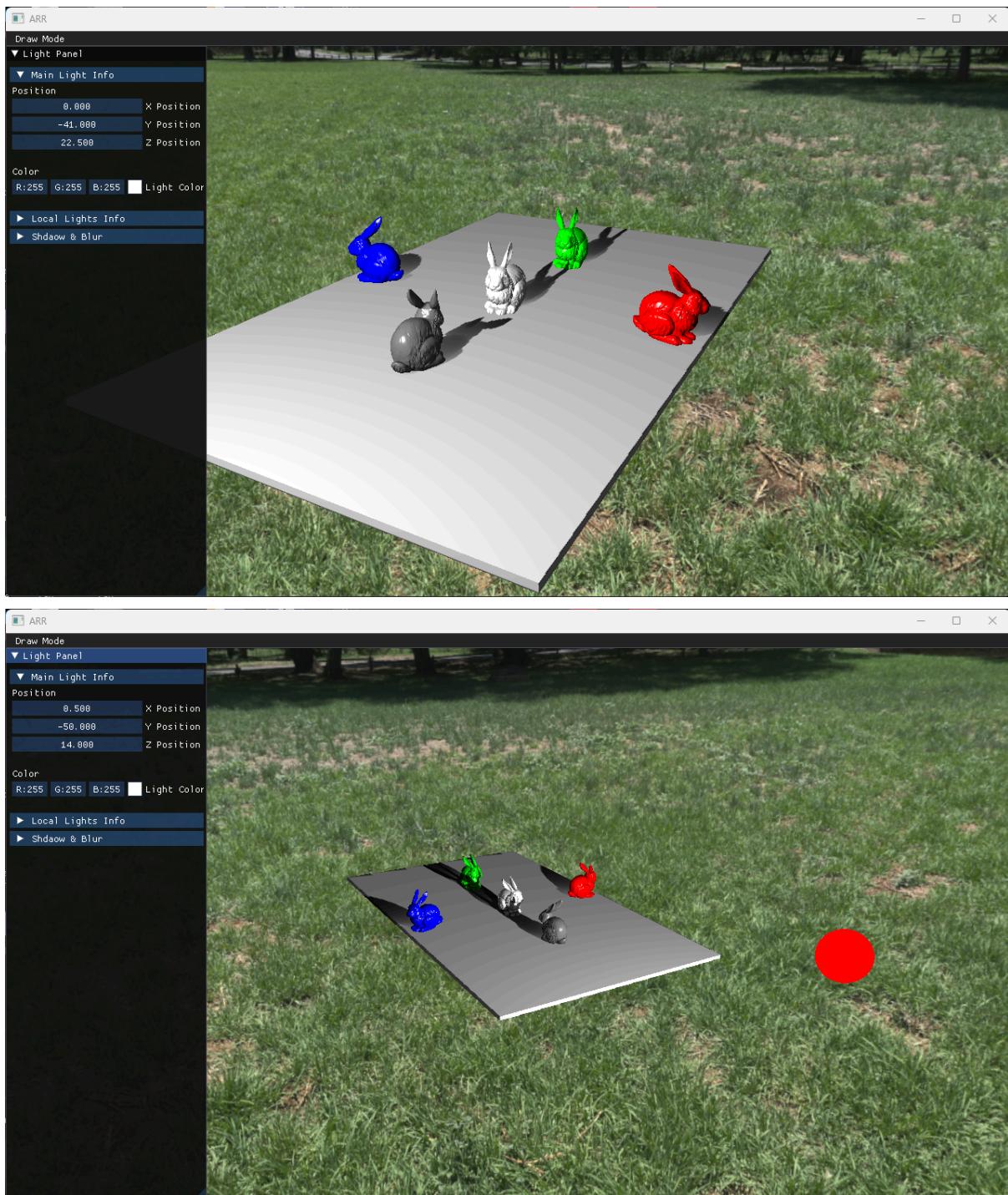




These pictures shows that the shadow map texture image.

## 2. Total Result (with out blur, only MSM)





This scene shows that the total result of Light + Shadow result.

+ ) I used the “Log Depth Buffer Algorithm”. I want to use a high value of the far plane for shadow maps. But as we know the high value of shadow far plane gives incorrect depth value because the value between near and far plane is so big. So I used the Log Depth Buffer Algorithm to solve this problem.

This is the code for it:

```

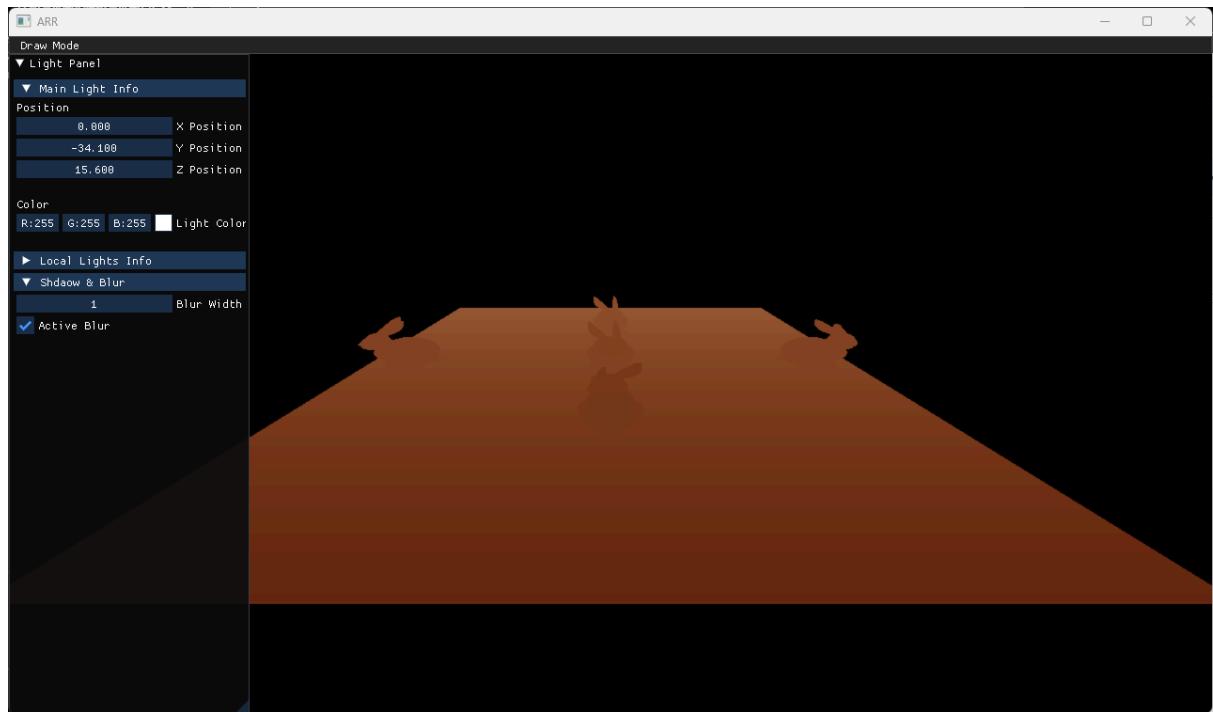
float z = position.w;
 $\star$ 
float logDepth = log2(z - near + 1.0f) / log2(far - near+1.0f);
shadowData = vec4(logDepth, logDepth*logDepth, logDepth*logDepth*logDepth, logDepth*logDepth*logDepth*logDepth);

```

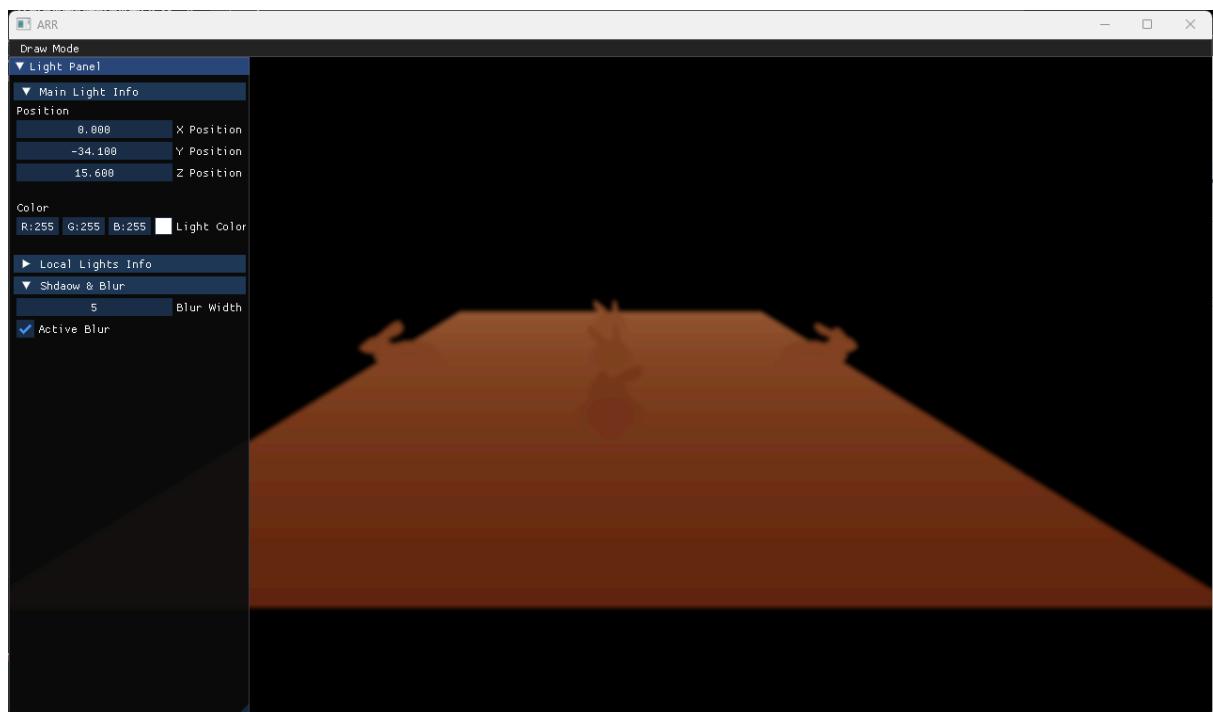
## Convolution filter with a Gaussian weight kernel:

### 1. Shadow Map (with blur)

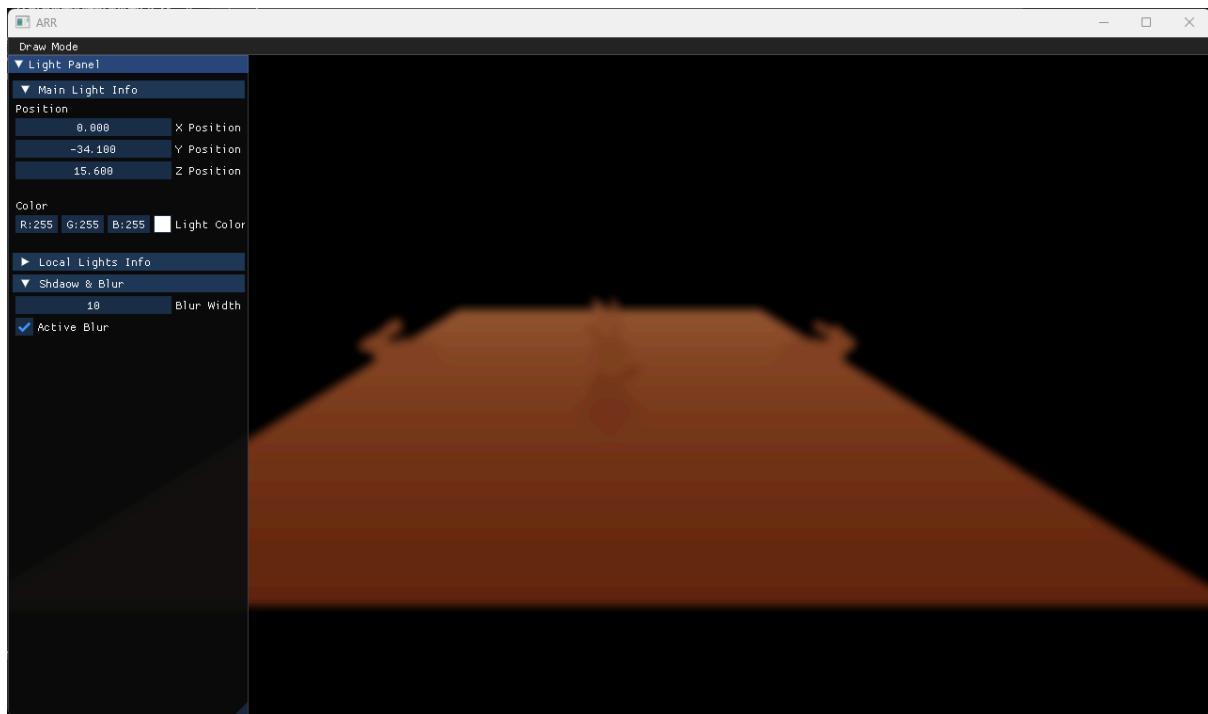
- blur width : 1



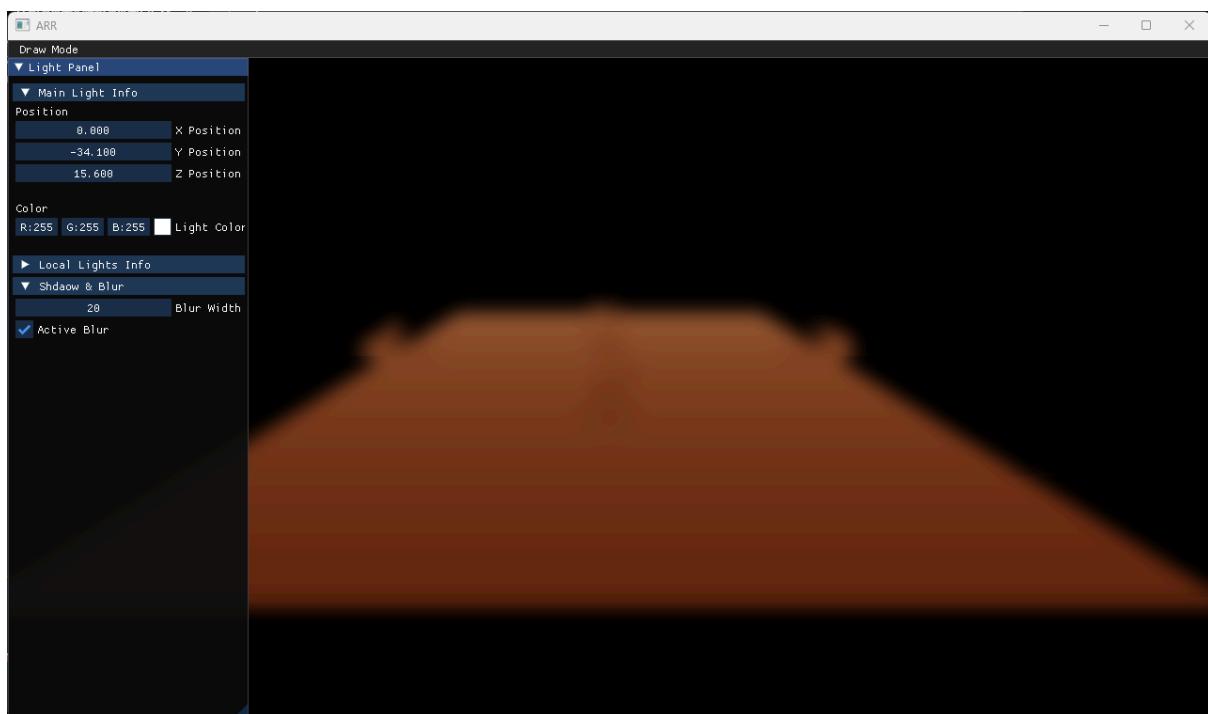
- blur width : 5



- blur width : 10



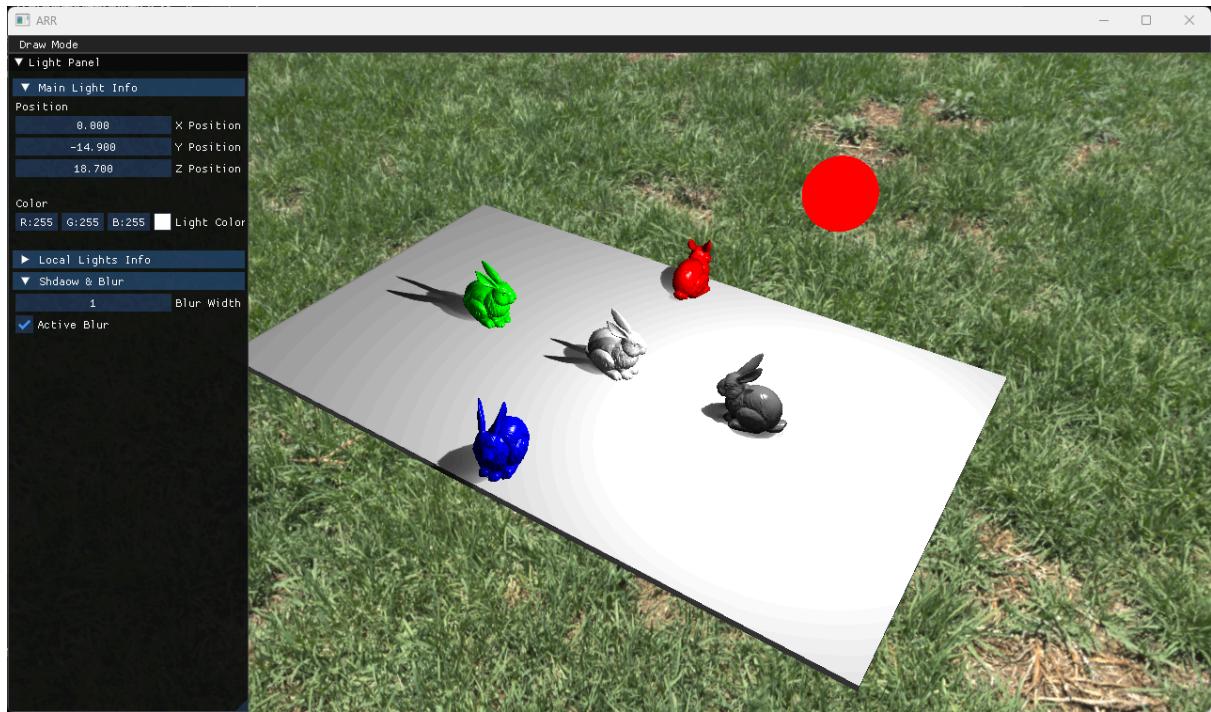
- blur width: 20



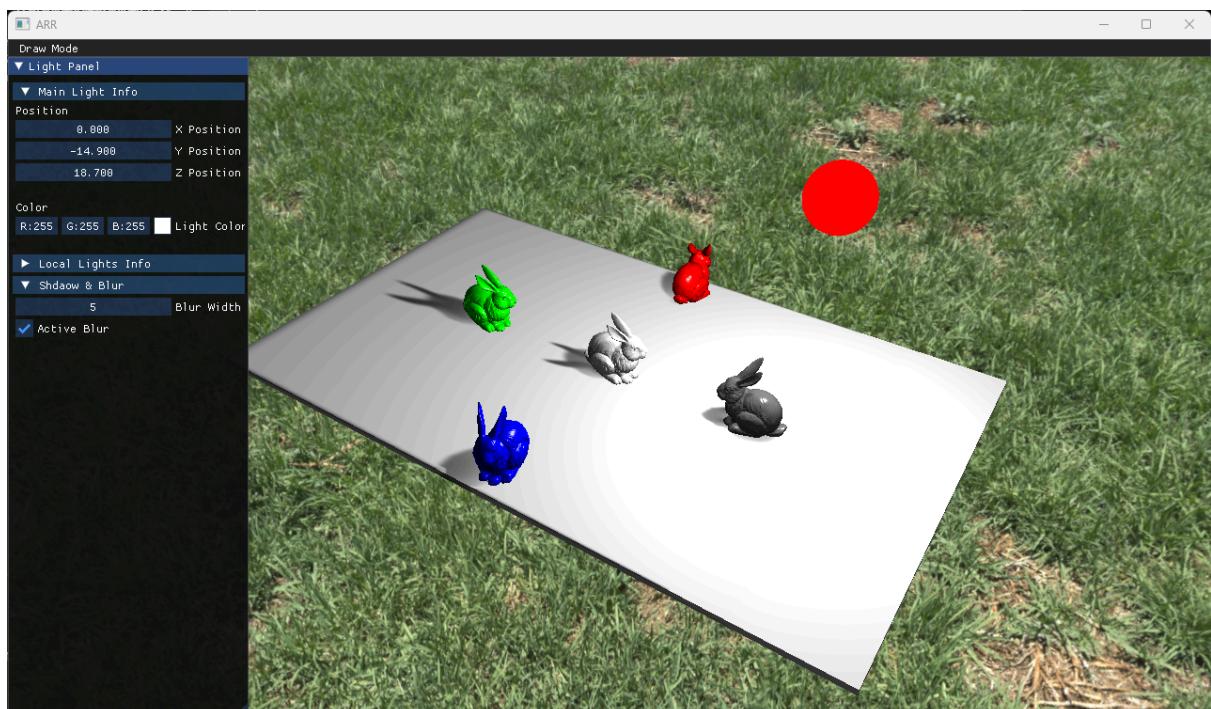
These pictures shows that the shadow map texture image with blur effect by compute shader.

## 2. Total Result (with blur)

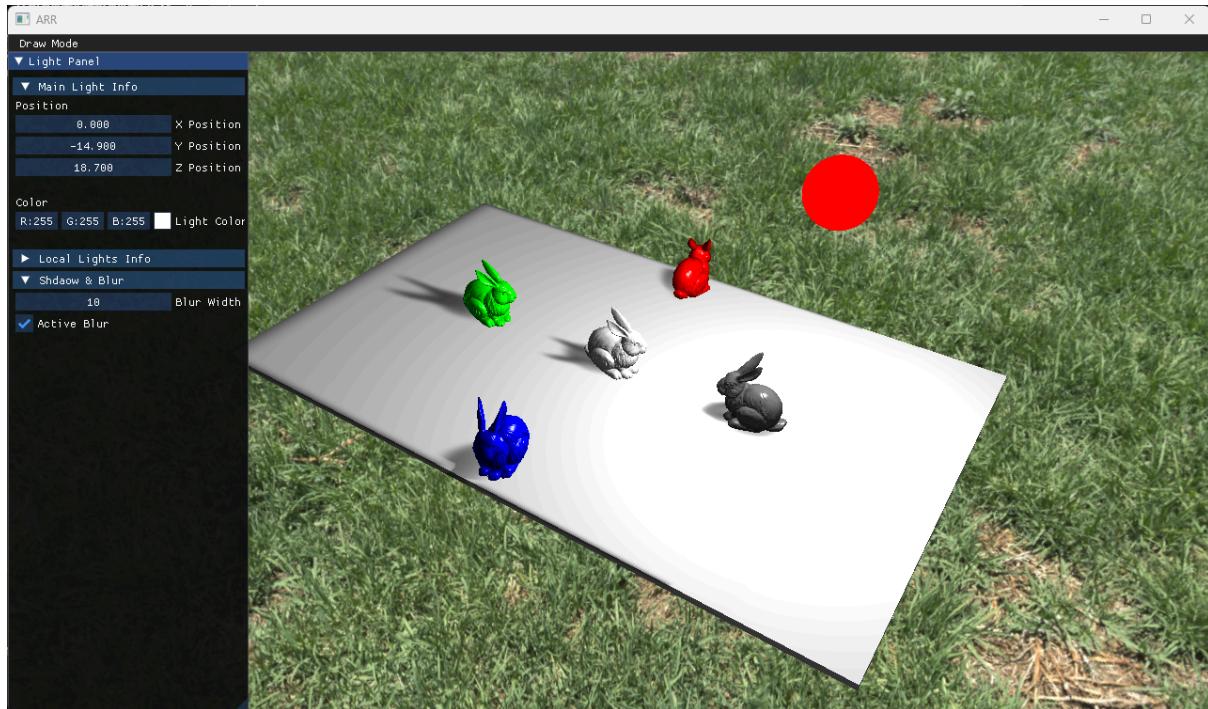
- blur width: 1



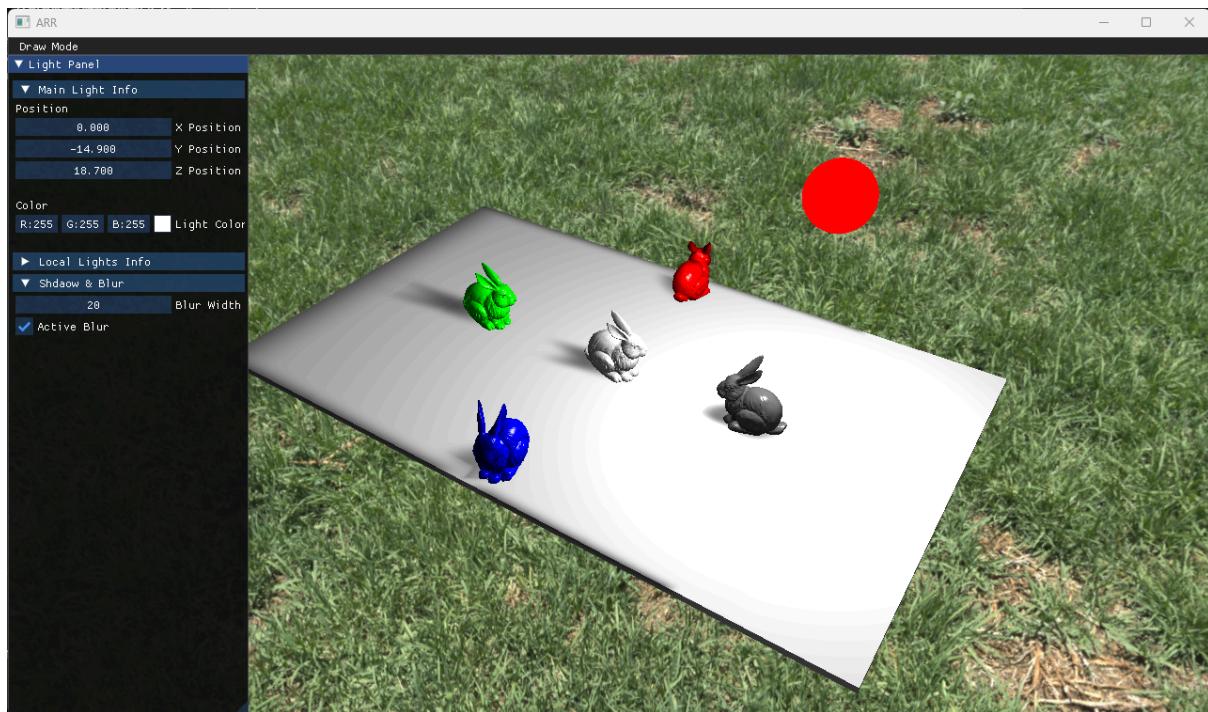
- blur width: 5



- blur width: 10



- blur width: 20



This scene shows that the total result of Light + Shadow result with blur effect by compute shader.