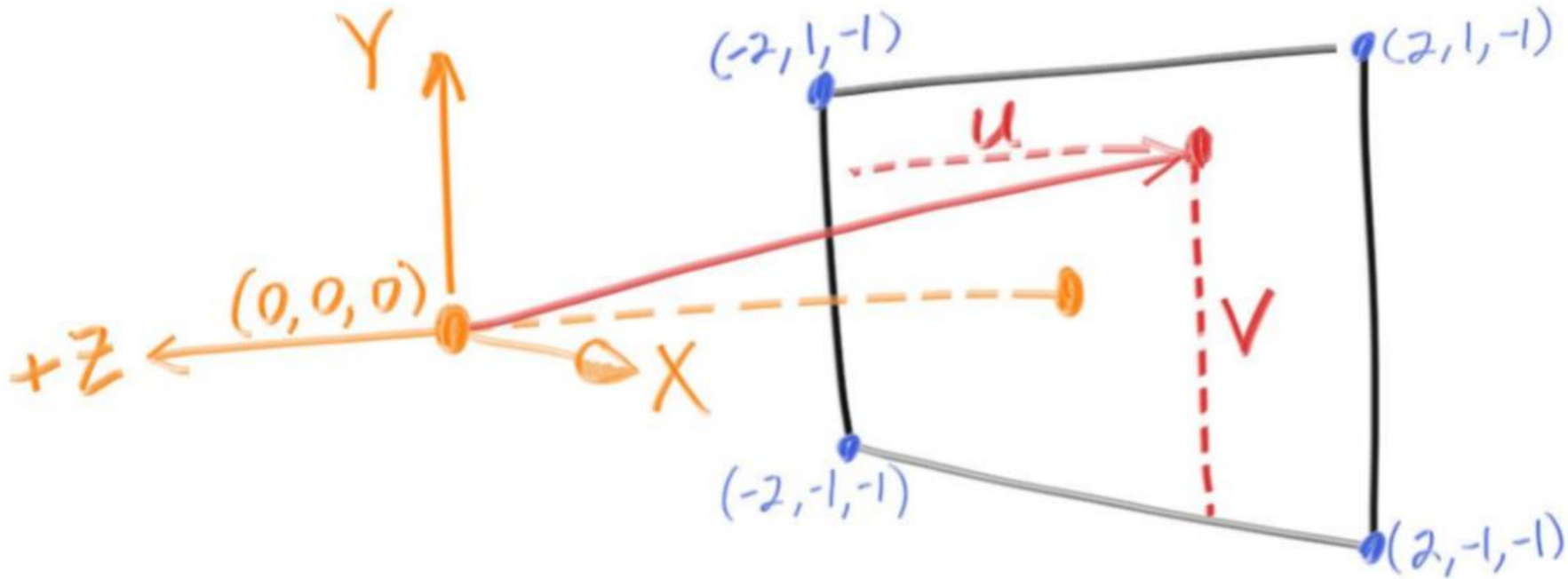
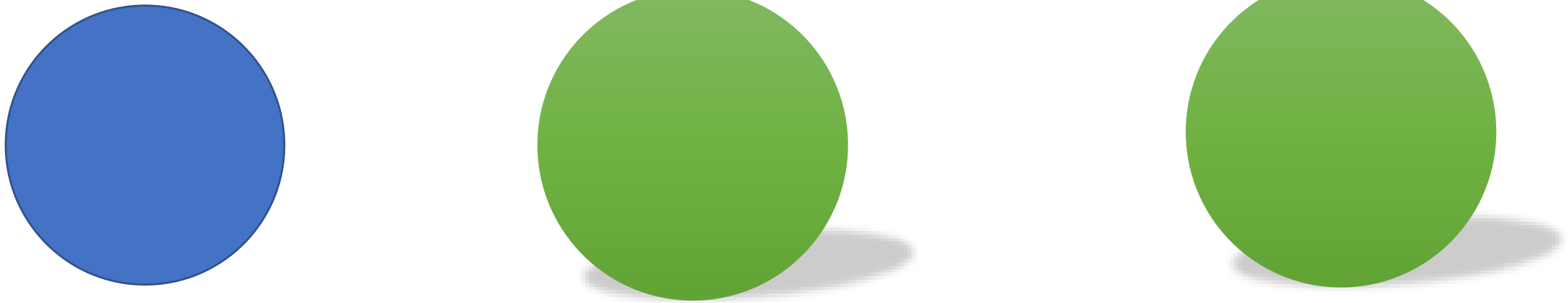
Chapter_7
Ray Tracer 1.0

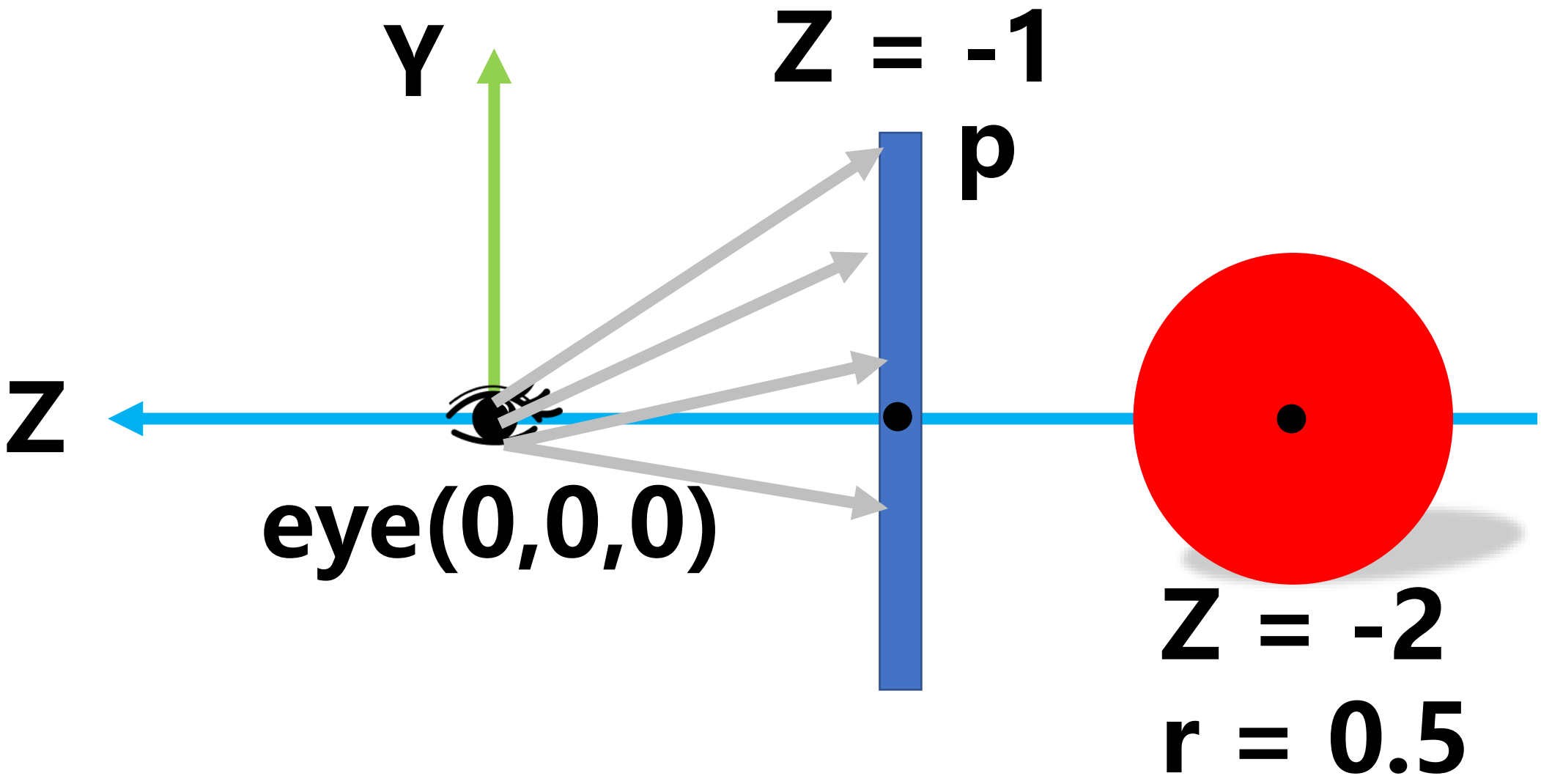
主讲人：王世元

西南石油大学 2020

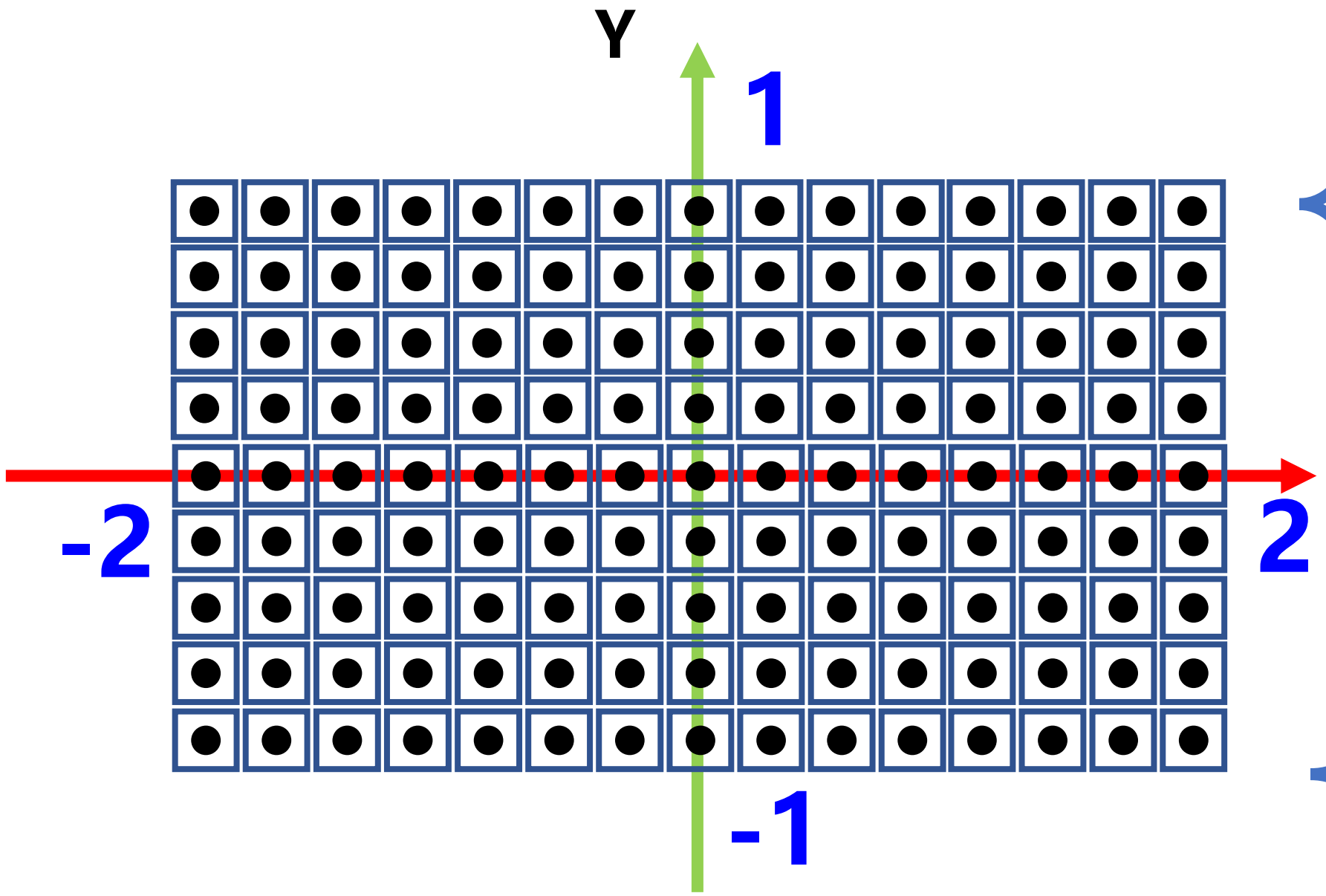

Y
(-2,1,-1)
(2,1,-1)
u
(0,0,0)
+Z
X
V
(-2,-1,-1)
(2,-1,-1)



成像平面



成像平面



X[-2,2]

Y[-1,1]

Z = -1

Res (200,100)

像素间隔

H = 4.0 / 200

V = 2.0 / 100

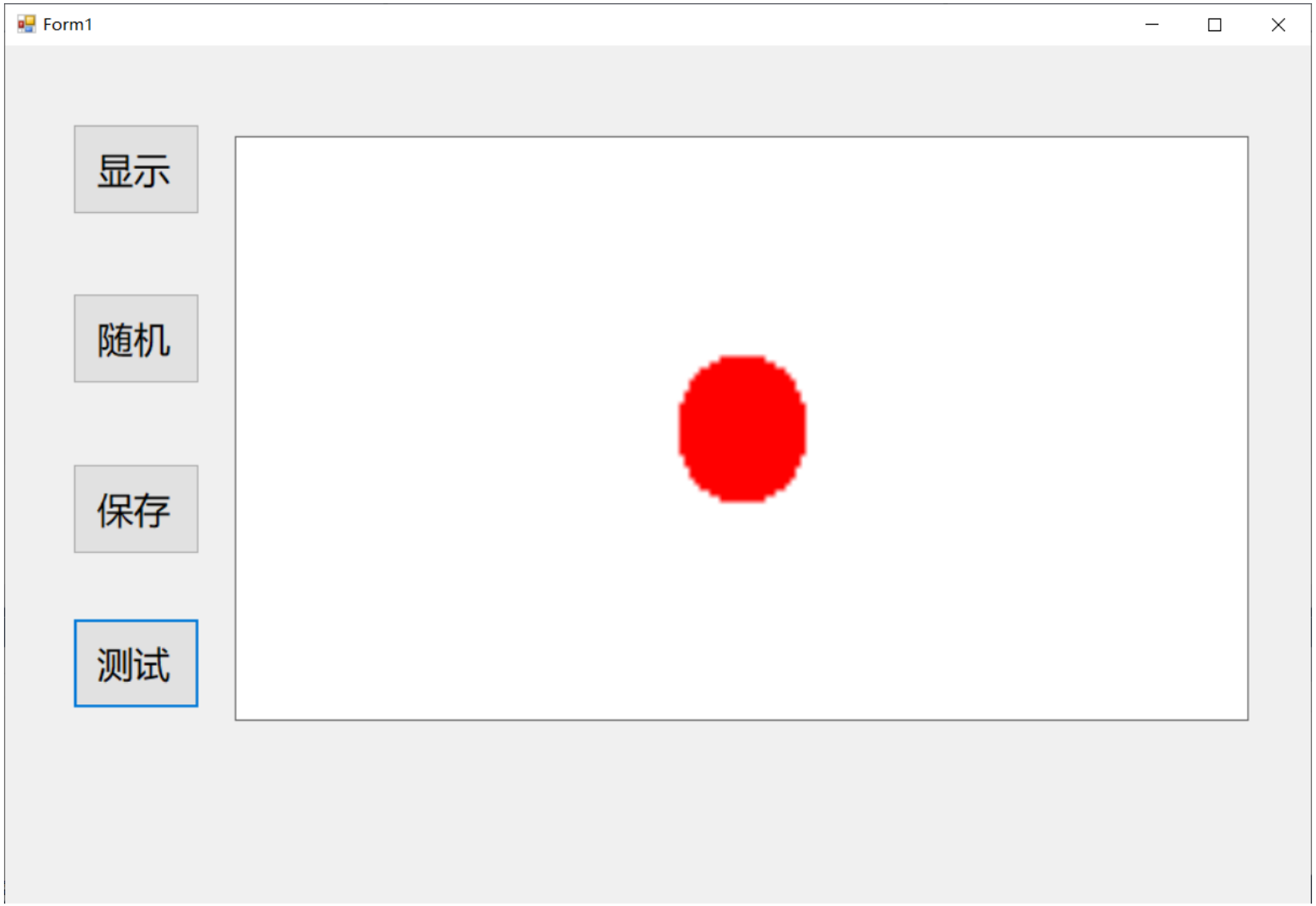
我们的第一个版本

```
//光线类测试
1 个引用
private void btnTest_Click(object sender, EventArgs e)
{
    //观察点位置
    Point3D eye = new Point3D();

    //球体位置
    Sphere sphere = new Sphere(new Point3D(0, 0, -2), 0.5);

    //用于做显示的bmp
    Bitmap bmp = new Bitmap(200,100);
```

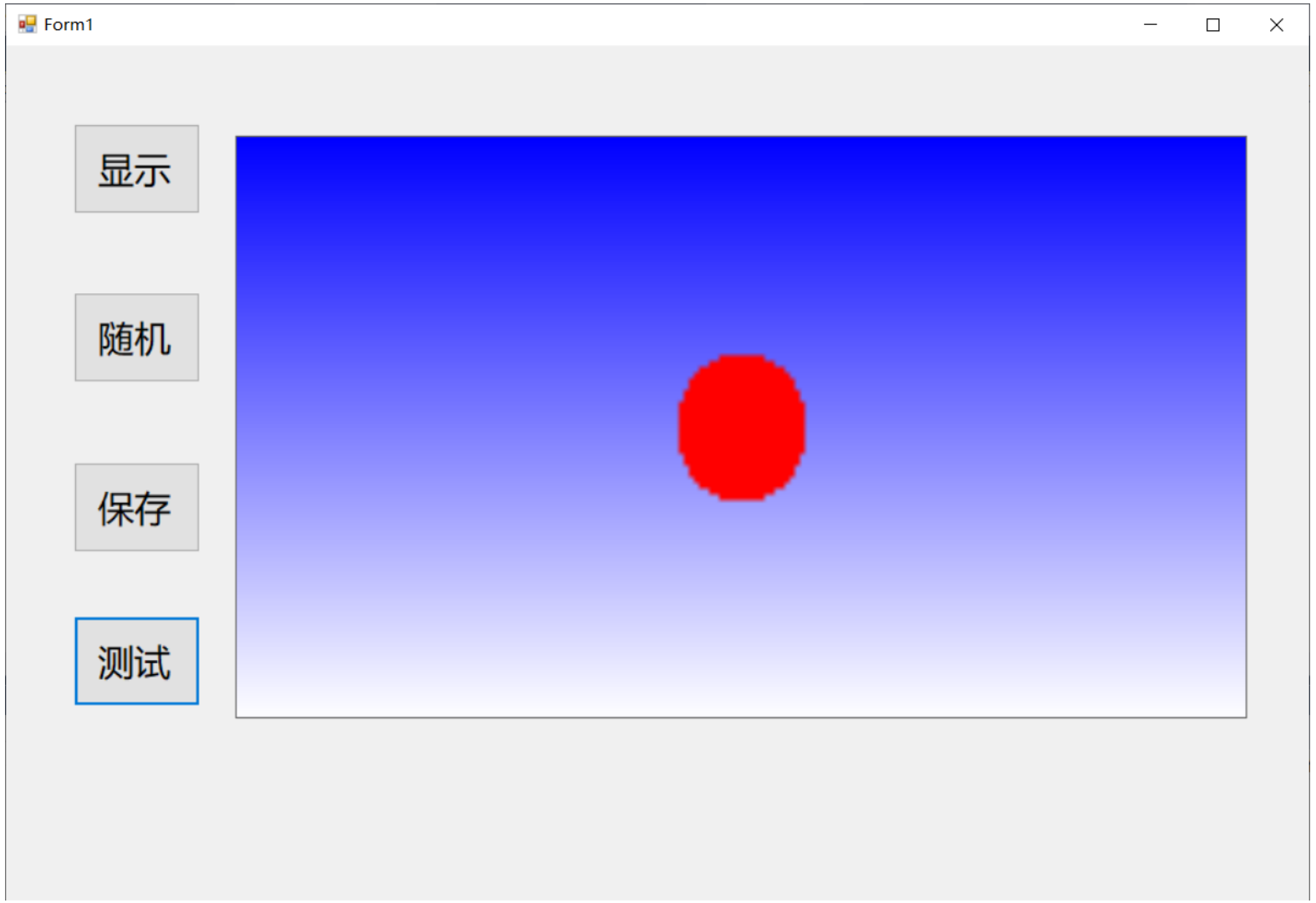
我们的第一个版本

```
1 个引用
private void btnTest_Click(object sender, EventArgs e)
{
    Point3D eye = new Point3D(); //观察点位置
    Sphere sphere = new Sphere(new Point3D(0, 0, -2), 0.5);//球体位置
    Bitmap bmp = new Bitmap(200,100); //用于做显示的bmp

    for (int i = 0; i < 200; i++)
    {
        for (int j = 0; j < 100; j++)
        {
            //成像平面上的每个点的位置
            Point3D p = new Point3D(-2 + 0.02 * i, 1 - 0.02 * j, -1);

            //起始光线的方向
            Vector3D dir = p - eye;
            Ray primaryRay = new Ray(eye,dir);

            //相交测试
            if (primaryRay.isHit(sphere))
            {
                bmp.SetPixel(i, j, Color.Red);
            }
            else
            {
                bmp.SetPixel(i, j, Color.White);
            }
        }
    }

    picRt.BackgroundImage = bmp;
```

我们的第一个版本



来点渐变

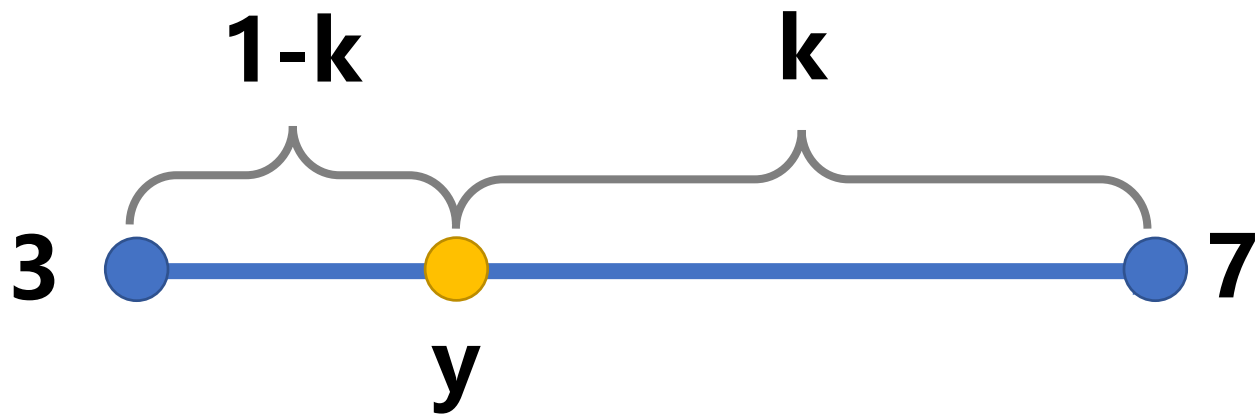


←---蓝色

←---白色

线 性 插 值

$k[0.0, 1.0]$



$$y = (1 - k) * 3 + k * 7;$$

$$color = (1 - k) * blue + k * white;$$

```
1 个引用
public static SColor Blue
{
    get
    {
        return new SColor(0, 0, 1);
    }
}

1 个引用
public static SColor White
{
    get
    {
        return new SColor(1,1,1);
    }
}
```

```
2 个引用
public static SColor operator *(double d, SColor color)
{
    return new SColor(d * color.R, d * color.G, d * color.B);
}

1 个引用
public static SColor operator +(SColor color1, SColor color2)
{
    return new SColor(color1.R + color2.R,
        color1.G + color2.G,
        color1.B + color2.B);
}
```

```
1 个引用
public Color ToRGB255Color()
{
    return Color.FromArgb((int)(R*255), (int)(G*255), (int)(B * 255));
}
```

```
//相交测试
if (primaryRay.isHit(sphere))
{
    bmp.SetPixel(i, j, Color.Red);
}
else
{
    double t = j / 100.0;
    SColor r = (1-t) * SColor.Blue + t * SColor.White;
    bmp.SetPixel(i, j, r.ToRGB255Color());
}
```