NIGHT VISION

게임프로그래밍
소프트웨어응용학부
2017040060
안장훈

REC

1. 구상 동기

영화 REC 中

게임 OUTLAST 中

다양한 매체에서 활용되는 카메라의 야간투시 효과

2. EFFECTS

캠코더

주요 기능

1. Night vision 야간투시 효과

2. 배터리의 상태에 따른 변화

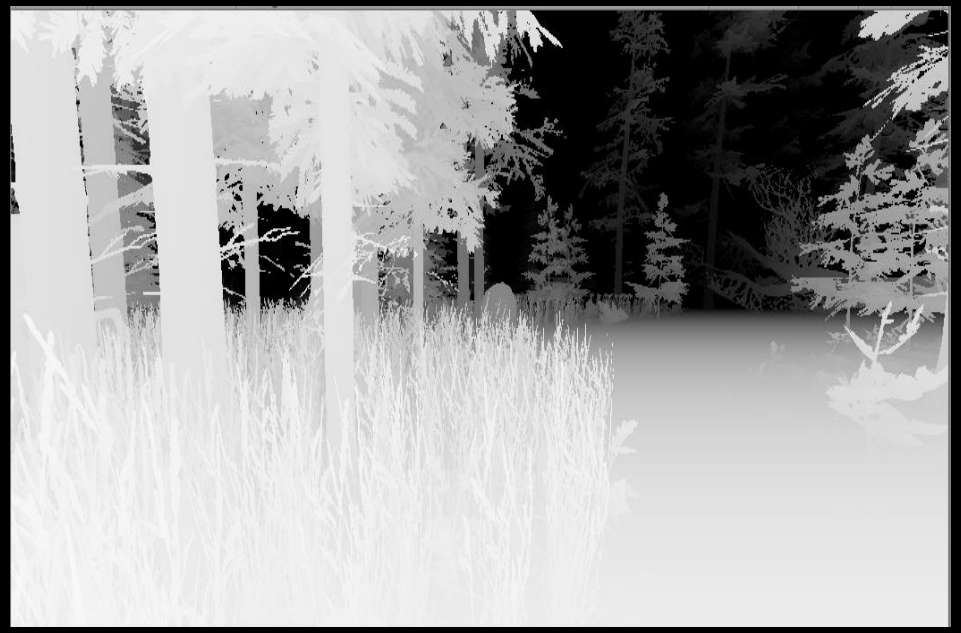
3. 화면의 노이즈 효과

4. Zoom in Zoom out 효과

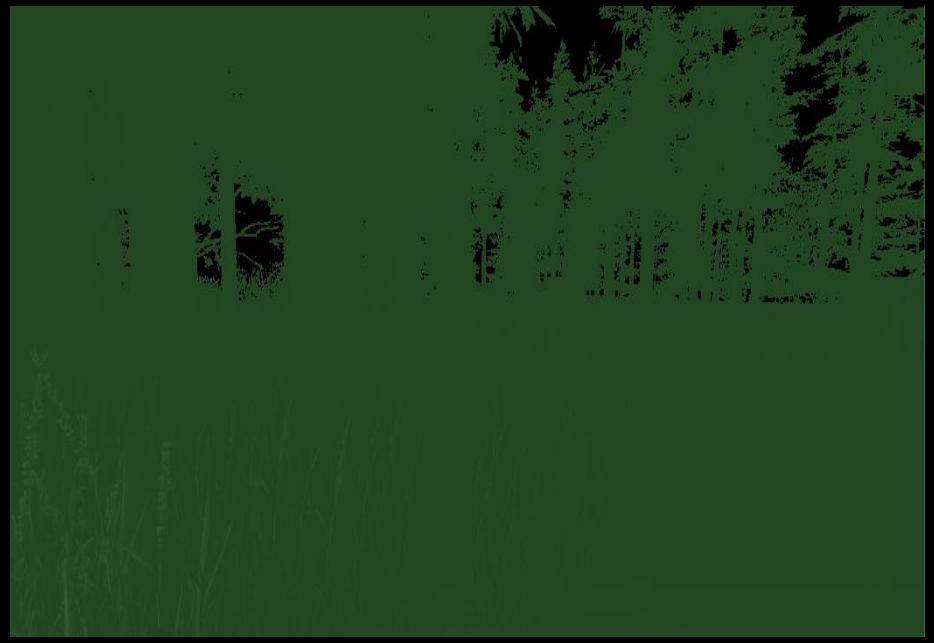
5. Glitch 효과

2. EFFECTS

Night vision 야간투시 효과



Depth view



Green 값 증가 후 합성

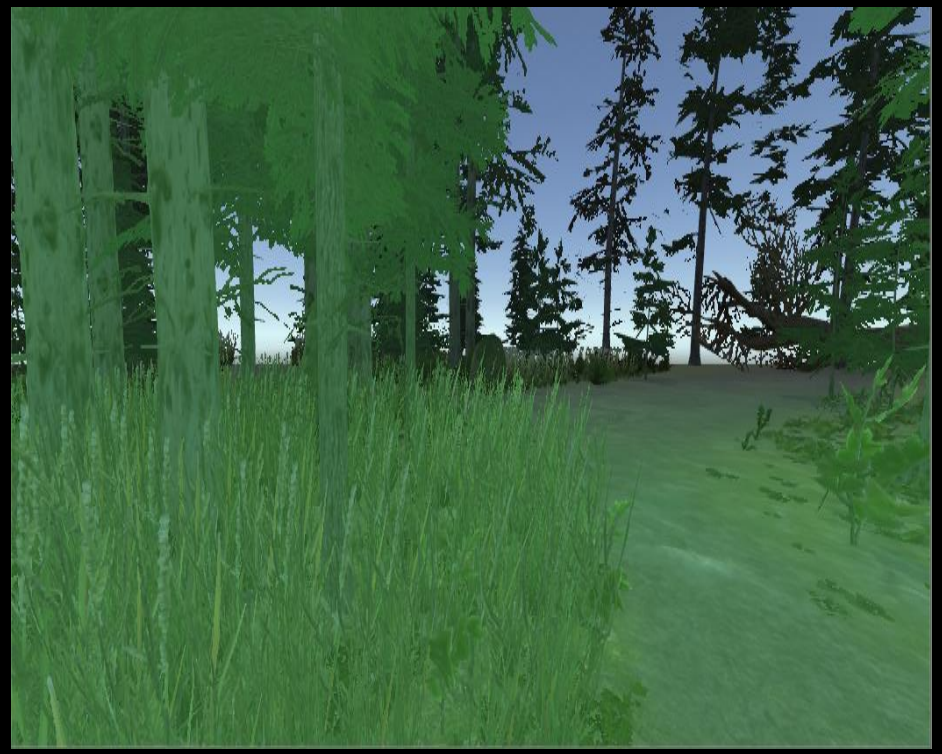
2. EFFECTS

Night vision 야간투시 효과

```
half4 frag(v2f i) : COLOR{
    float depthValue = Linear01Depth(tex2Dproj(_CameraDepthTexture, UNITY_PROJ_COORD(i.uv)).r);
    half4 color = tex2D(_MainTex, i.uv);
    half4 depth = half4((1 - depthValue), (1 - depthValue), (1 - depthValue), 1);

    depth.r *= 0.5;
    depth.g = clamp(depth.g * 1.2, 0,1);
    depth.b *= 0.5;
    depth.rgb = lerp(color.rgb, depth.rgb, depth.g * _NightVision);
    return depth;
}
```

MainTex + Depth



MainTex + Depth

2. EFFECTS

배터리 상태에 따른 변화

```
//Battery 잔량에 따른 화면 flickering
if (_Battery
else if(_Batt
```

```
//배터리
if (battery > 80) mat.SetTexture("_BatteryTex", battery_textures[0]);
else if (battery > 60) mat.SetTexture("_BatteryTex", battery_textures[1]);
else if (battery > 40) mat.SetTexture("_BatteryTex", battery_textures[2]);
else if (battery > 20) mat.SetTexture("_BatteryTex", battery_textures[3]);
else if (battery > 0) mat.SetTexture("_BatteryTex", battery_textures[4]);
mat.SetFloat("_Battery", battery);
mat.SetFloat("_Noise", 5 - (noise_value + 1));
mat.SetFloat("_Scale", scale);

Graphics.Blit(source, destination, mat);
```

```
//Battery 잔량 점멸
if (_Battery < 10) {
    if (battery.a > 0.1)battery.rgb = fixed3(1, 0.2, 0.2);
    battery.a = clamp(abs(sin(_Time * 50)), 0, 1);
}
c = lerp(c, battery, battery.r == 0 ? 0 : battery.a);
```

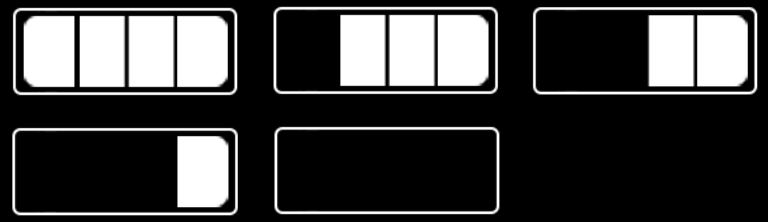
배터리 낮을시 배터리칸 점멸

```
    return frac(sin(dot(uv,float2(12.9898,78.233)))*43758.5453123);
}
```

Random 함수

2. EFFECTS

배터리 상태에 따른 변화



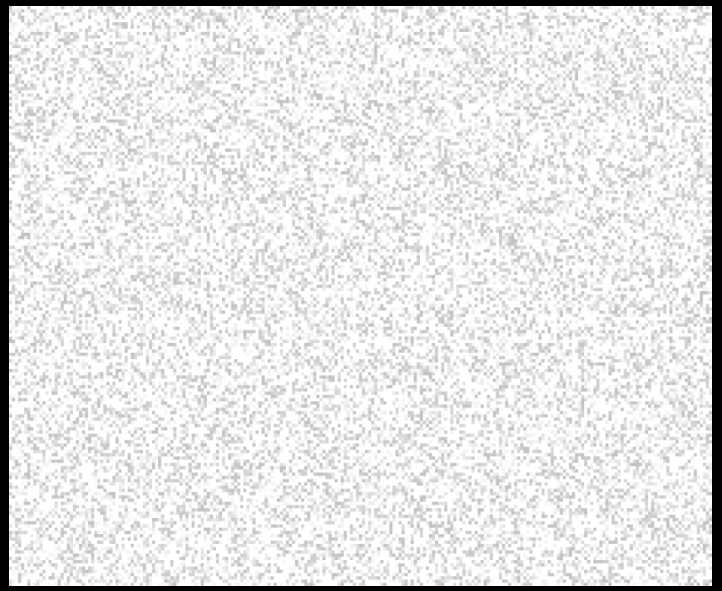
2. EFFECTS

배터리 상태에 따른 변화

2. EFFECTS

화면의 노이즈 효과

실제 카메라의 노이즈



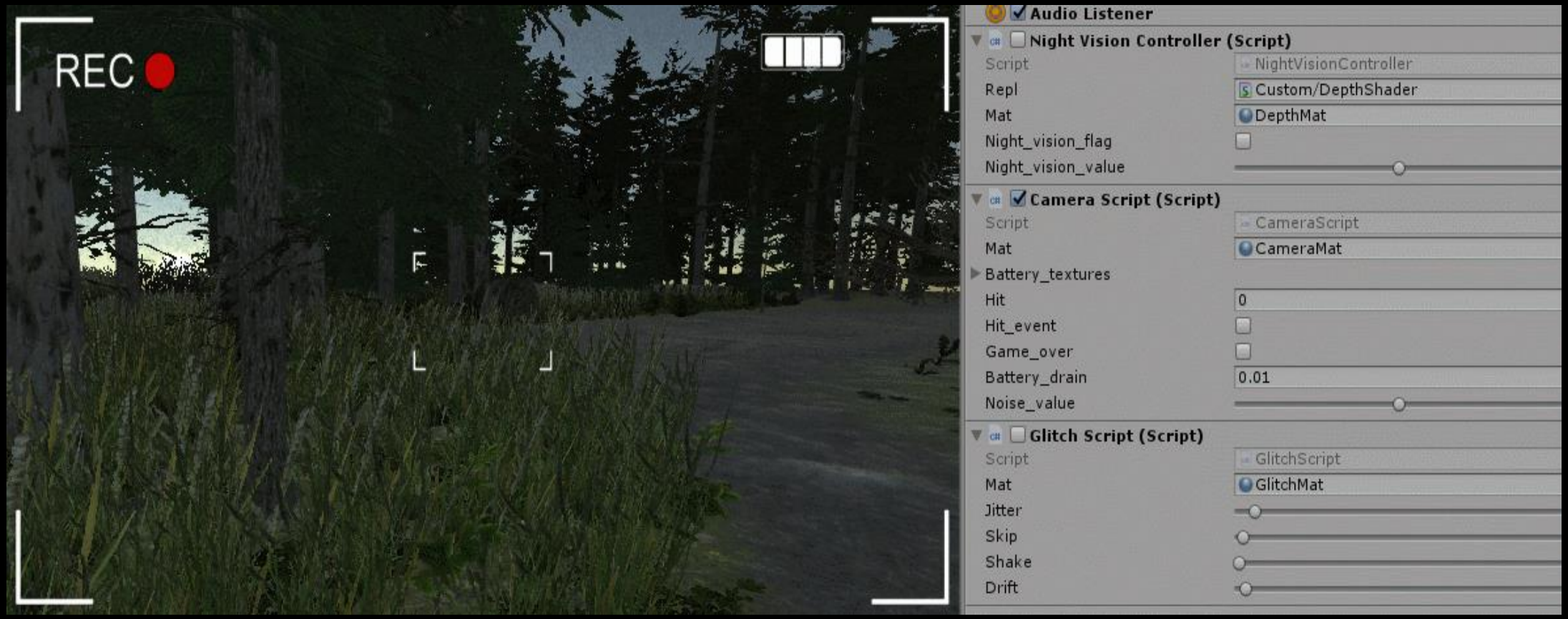
노이즈 텍스처

```
if (_Noise < 4) {
    float2 uv = i.uv;
    uv += i.uv * random(_Time);
    v *= tex2D(_NoiseTex, uv * _Noise);
}
```

랜덤함수를 사용한 불규칙적 노이즈 텍스처 합성

2. EFFECTS

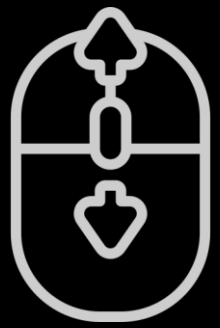
화면의 노이즈 효과



2. EFFECTS

Zoom in Zoom out 효과

```
float scroll = Input.GetAxis("Mouse ScrollWheel") * 5;
if (scale > 0.3 && scroll > 0 && camera_on) scale -= 0.1f;
if (scale < 1 && scroll < 0 && camera_on) scale += 0.1f;
```



setFloat

```
fixed2 center = (0.5, 0.5);
//Zoom in Zoom out 구현
half4 v = tex2D(_MainTex, ((i.uv - center) * _Scale + center));
```

2. EFFECTS

Glitch 효과

```
c1 = tex2D(_MainTex, frac(float2(i.uv.x + jitter + shake, skip)));
c2 = tex2D(_MainTex, frac(float2(i.uv.x + jitter + shake + drift, skip)));
```

GLITCH?

```
skip_dt += Time.deltaTime * skip * 10.0f;
float jitter_thresh = Mathf.Clamp01(1.0f - jitter);
float jitter_displace = 0.002f + Mathf.Pow(jitter, 3) * 0.05f;

mat.SetVector("_Jitter", new Vector2(jitter_displace, jitter_thresh));
mat.SetVector("_Skip", new Vector2(skip, skip_dt));
mat.SetVector("_Drift", new Vector2(drift * 0.04f, Time.time));
mat.SetFloat("_Shake", shake * 0.2f);
Graphics.Blit(source, destination, mat);
```

JITTER

```
//jitter
float jitter = random(float2(i.uv.y, _Time.x)) * 2 - 1;
jitter *= step(_Jitter.y, abs(jitter)) * _Jitter.x;
```

SKIP

```
//skip
float skip = lerp(i.uv.y, frac(i.uv.y + _Skip.y), _Skip.x);
```

SHAKE

```
//shake
float shake = (random(float2(_Time.x, 2)) - 0.5) * _Shake;
```

DRIFT

```
//drift
float drift = clamp(skip + _Drift.y, 0, 1) * _Drift.x;
```

2. EFFECTS

Glitch 효과 JITTER

2. EFFECTS

Glitch 효과 SKIP

2. EFFECTS

Glitch 효과 SHAKE

2. EFFECTS

Glitch 효과 드리프트

2. EFFECTS

CAMERA EFFECTS

3. MINI GAME

간단한 맵

+

귀여운 몬스터

+

NIGHT VISION

= HORROR!

4. 시연 영상

REC

감사합니다!