

Blending

305890
2009년 봄학기
4/8/2009
박경신

Overview

- Blending
- Alpha Channel

Blending

- Blending Equation
 - 블렌딩의 동작원리와 이용 방법
- Blend Factors
 - Direct3D가 지원하는 다양한 블렌드 모드
- Transparency
 - 기본 도형의 투명도를 제어하기 위해 알파성분을 이용하는 방법
- Creating an Alpha Channel Using DX Tex Tool
- 예제: MtrlAlpha, texAlpha

Blending Equation

- Blending
 - 현재 계산되고 있는 픽셀 값 (즉, 원본 픽셀)에 앞서 쓰여진 픽셀값 (즉, 목적 픽셀)과 결합하는 방식
- Blending Rule
 1. Blending을 이용하지 않는 물체를 먼저 그린다
 2. Blending을 이용하는 물체들을 카메라와의 거리에 따라 정렬한다. (viewing space에서 물체들을 z 성분을 통해 정렬하면 된다.)
 3. Blending을 이용하는 물체들을 뒤에서 앞으로의 순서로 그려나간다.

Blending Equation

□ 두 pixel값을 Blending하는 Equation

$outputPixel = srcPixel * srcBlendFactor + dstPixel * dstBlendFactor$

- 여기서 각 변수는 (r, g, b, a) 벡터이고, *는 성분 단위의 곱임
- outputPixel은 Blending한 결과 pixel
- srcPixel은 현재 pixel
- dstPixel은 back buffer에서의 pixel
- srcBlendFactor와 dstBlendFactor는 [0,1] 범위 내의 값

Blending Equation

□ Blending활성화

Device->setRenderState(D3DRS_ALPHABLENDENABLE, true);

- Default는 비활성화
- 계산량을 위해서 꼭 필요한 경우에서만 활성화하여 사용하고 끝나면 다시 비활성화를 시킬 것.
- 필요한 삼각형들을 한꺼번에 처리하여 하나의 프레임 내에서 여러 번 켜고 끄는 일을 피할 것.

Blend Factors

□ Blend factor의 여러 조합으로 원하는 효과를 만들어 냄

Device->SetRenderState(D3DRS_SRCBLEND, Source);

Device->SetRenderState(D3DRS_DESTBLEND, Destination);

D3DBLEND_ZERO	(0, 0, 0, 0)
D3DBLEND_ONE	(1, 1, 1, 1)
D3DBLEND_SRCCOLOR	(r _s , g _s , b _s , a _s)
D3DBLEND_INVSRCCOLOR	(1-r _s , 1-g _s , 1-b _s , 1-a _s)
D3DBLEND_SRCALPHA	(a _s , a _s , a _s , a _s) srcFactor의 default
D3DBLEND_INVSRCALPHA	(1-a _s , 1-a _s , 1-a _s , 1-a _s) dstFactor의 default
D3DBLEND_DESTALPHA	(a _d , a _d , a _d , a _d)
D3DBLEND_INVDESTALPHA	(1-a _d , 1-a _d , 1-a _d , 1-a _d)
D3DBLEND_DESTCOLOR	(r _d , g _d , b _d , a _d)
D3DBLEND_INVDESTCOLOR	(1-r _d , 1-g _d , 1-b _d , 1-a _d)
D3DBLEND_SRCALPHASAT	(f, f, f, 1), f = min(a _s , 1-a _d)
D3DBLEND_BOTHINVSRCALPHA	srcFactor = (1-a _s , 1-a _s , 1-a _s , 1-a _s) dstFactor = (a _s , a _s , a _s , a _s)

NOTE:D3DRS_SRCBLEND에서만 이용가능

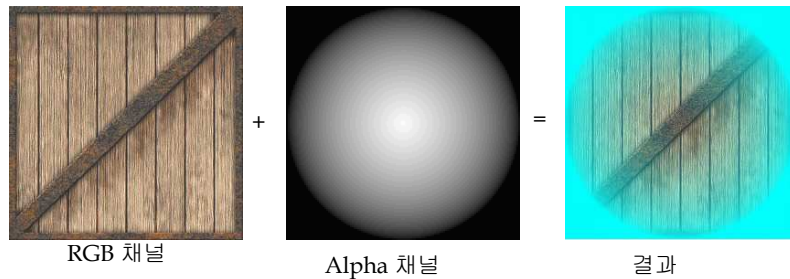
Transparency

□ 투명 (Transparency)

- Pixel의 알파성분 (alpha component)으로 투명도 단계를 지정함.
- 예를 들어, 각 픽셀마다 8비트씩 알파 성분 할당 [0, 255]:
불투명도: alpha 0 - 0%는 완전투명
alpha 128 - 50%는 반투명
alpha 255 - 100%는 완전불투명에 해당
- alpha성분으로 투명도를 지정하려면 srcBlendFactor, dstBlendFactor를 D3DBLEND_SRCALPHA, D3DBLEND_INVSRCALPHA로 지정해야 함 (default로 되어 있음)

Alpha Channel

- 알파채널 (Alpha channel)
 - Alpha 성분: (1) shading시 계산하여 얻거나, (2) texture에서의 alpha channel로부터 얻음.
 - Alpha channel: texture mapping시 alpha channel도 mapping되며 이것이 face에서의 pixel의 alpha성분이 됨.



Set Transparency

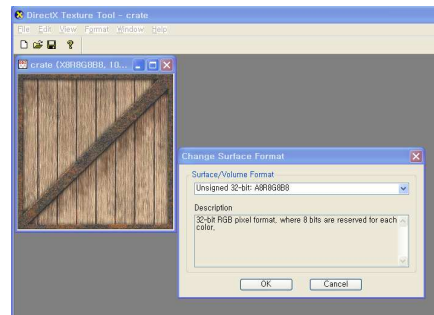
- 현재 texture가 alpha channel을 가지고 있으면 alpha channel에서 가져오고, 없으면 vertex color에서 가져옴.
- 어떤 소스를 이용할 것인지를 지정하는 예:


```
// shading과정에서 diffuse color로부터 alpha를 얻도록 지정함
Device->SetTextureStageState(0, D3DTSS_ALPHAARG1, D3DTA_DIFFUSE);
Device->SetTextureStageState(0, D3DTSS_ALPHAOP, D3DTOP_SELECTARG1);

// alpha channel에서 alpha를 얻도록 지정함
Device->SetTextureStageState(0, D3DTSS_ALPHAARG1, D3DTA_TEXTURE);
Device->SetTextureStageState(0, D3DTSS_ALPHAOP, D3DTOP_SELECTARG1);
```

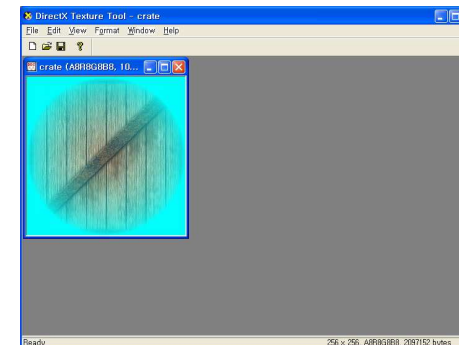
Creating Alpha Channel Using DX Tex Tool

- Alpha channel을 포함하는 DDS파일 제작법
 - DirectX9 SDK Texture Tool을 실행
 - Program->Microsoft DirectX 9.0 SDK Update (February 2005)->DirectX Utilities->DirectX Texture Tool 실행
 - 이미지에 alpha channel을 추가하기
 - File->Open에서 crate.jpg (원래 24-bit RGB)
 - Format->Change Surface Format을 32-bit A8R8G8B8로 변경



Creating Alpha Channel Using DX Tex Tool

- Alpha channel을 포함하는 DDS파일 제작법
 - 이미지의 alpha channel에 데이터를 넣기
 - 8 bit grey-scale image를 준비 (alphachannel.bmp)
 - File->Open Onto Alpha Channel Of This Texture 에서 alphachannel.bmp (8-bit grey-scale)를 읽어들임.
 - File->Save As에서 createwalpa.dds로 저장



Example: MtrlAlpha

- 예제 MtrlAlpha에서는 알파블렌딩을 사용하여 Semi-transparent teapot을 그린다. 'A'와 'S'-key로 알파투명도를 조절한다.



Example: MtrlAlpha

```
#include "d3dUtility.h"
IDirect3DDevice9* Device = 0;
const int Width = 640;
const int Height = 480;
ID3DXMesh* Teapot = 0;
D3DMATERIAL9 TeapotMtrl;
IDirect3DVertexBuffer9* BgQuad = 0;
IDirect3DTexture9* BgTex = 0;
D3DMATERIAL9 BgMtrl;
bool Setup() {
    TeapotMtrl = d3d::RED_MTRL;
    TeapotMtrl.Diffuse.a = 0.5f;           // set alpha to 50% opacity
    BgMtrl = d3d::WHITE_MTRL;
    // Create teapot
    D3DXCreateTextureFromFile(Device, "crate.jpg", &BgTex);
    // Set Texture Filter States.
    Device->SetSamplerState(0, D3DSAMP_MAGFILTER, D3DTEXF_LINEAR);
    Device->SetSamplerState(0, D3DSAMP_MINFILTER, D3DTEXF_LINEAR);
    Device->SetSamplerState(0, D3DSAMP_MIPFILTER, D3DTEXF_POINT);
}
```

Example: MtrlAlpha

```
.....
// set alpha blending
Device->SetTextureStageState(0, D3DTSS_ALPHAAARG1, D3DTA_DIFFUSE);
Device->SetTextureStageState(0, D3DTSS_ALPHAOP,
D3DTOP_SELECTARG1);
// set blending factors
Device->SetRenderState(D3DRS_SRCBLEND, D3DBLEND_SRCALPHA);
Device->SetRenderState(D3DRS_DESTBLEND, D3DBLEND_INVSRCALPHA);
...
// set camera
...
// set projection matrix
...
return true;
}
```

Example: MtrlAlpha

```
bool Display(float timeDelta)
{
    if (Device) {
        // increase/decrease alpha via keyboard input
        if (::GetAsyncKeyState('A') & 0x8000f)
            TeapotMtrl.Diffuse.a += 0.01f;
        if (::GetAsyncKeyState('S') & 0x8000f)
            TeapotMtrl.Diffuse.a -= 0.01f;
        // force alpha to [0, 1] interval
        if (TeapotMtrl.Diffuse.a > 1.0f)
            TeapotMtrl.Diffuse.a = 1.0f;
        if (TeapotMtrl.Diffuse.a < 0.0f)
            TeapotMtrl.Diffuse.a = 0.0f;
        ....
    }
}
```

Example: MtrlAlpha

```
.....
// render
Device->Clear(0, 0, D3DCLEAR_TARGET | D3DCLEAR_ZBUFFER, 0xffffffff,
1.0f, 0);
Device->BeginScene();
// draw background
D3DXMATRIX W;
D3DXMatrixIdentity(&W);
Device->SetTransform(D3DTS_WORLD, &W);
Device->SetFVF(Vertex::FVF);
Device->SetStreamSource(0, BgQuad, 0, sizeof(Vertex));
Device->SetMaterial(&BgMtrl);
Device->SetTexture(0, BgTex);
Device->DrawPrimitive(D3DPT_TRIANGLELIST, 0, 2);
.....
```

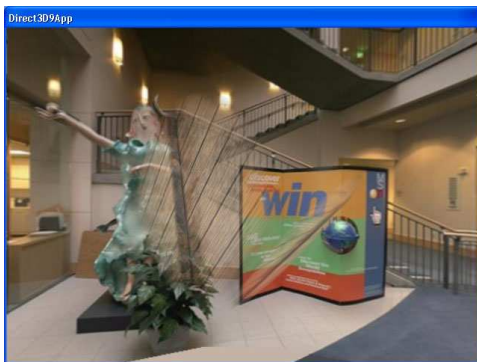
Example: MtrlAlpha

```
.....
// draw teapot
Device->SetRenderState(D3DRS_ALPHABLENDENABLE, true);
D3DXMatrixScaling(&W, 1.5f, 1.5f, 1.5f);
Device->SetTransform(D3DTS_WORLD, &W);
Device->SetMaterial(&TeapotMtrl);
Device->SetTexture(0, 0);
Teapot->DrawSubset(0);
Device->SetRenderState(D3DRS_ALPHABLENDENABLE, false);

Device->EndScene();
Device->Present(0, 0, 0, 0);
}
return true;
}
```

Example: texAlpha

- 예제 texAlpha에서는 알파블렌딩을 사용하여 Semi-transparent cube를 그린다. 이 예제에서는 텍스처 알파 채널에서 알파값을 가져온다.



Example: texAlpha

```
IDirect3DTexture9*   CrateTex = 0;
Cube*                Box = 0;
D3DXMATRIX           CubeWorldMatrix;
IDirect3DVertexBuffer9* BackDropVB = 0;
IDirect3DTexture9*   BackDropTex = 0;

bool Setup()
{
    // create the backdrop quad
    // create the cube
    // load the textures and set filters
    D3DXCreateTextureFromFile(Device, "cratealpha.dds", &CrateTex);
    D3DXCreateTextureFromFile(Device, "lobbyxpos.jpg", &BackDropTex);
    // Set sampler filter
    // Set alpha blending using alpha channel in texture
    Device->SetTextureStageState(0, D3DTSS_ALPHAARG1, D3DTA_TEXTURE);
    Device->SetTextureStageState(0, D3DTSS_ALPHAOP,
    D3DTOP_SELECTARG1);
}
```

Example: texAlpha

```
// set blending factors
// disable lighting
// set camera
// set projection matrix
return true;
}
bool Display(float timeDelta)
{
    if (Device) {
        D3DXMATRIX xRot;
        D3DXMatrixRotationX(&xRot, D3DX_PI * 0.2f);
        static float y = 0.0f;
        D3DXMATRIX yRot;
        D3DXMatrixRotationY(&yRot, y);
        y += timeDelta;
        if (y >= 6.28f) y = 0.0f;
        CubeWorldMatrix = xRot * yRot;
    }
}
```

Example: texAlpha

```
// render
Device->Clear(0, 0, D3DCLEAR_TARGET | D3DCLEAR_ZBUFFER,
0xff0000ff, 1.0f, 0);
Device->BeginScene();
// draw backdrop
D3DXMATRIX I;      D3DXMatrixIdentity(&I);
Device->SetStreamSource(0, BackDropVB, 0, sizeof(Vertex));
Device->SetFVF(FVF_VERTEX);
Device->SetTexture(0, BackDropTex);
Device->SetTransform(D3DTS_WORLD, &I);
Device->DrawPrimitive(D3DPT_TRIANGLELIST, 0, 2);
// draw cube
Device->SetRenderState(D3DRS_ALPHABLENDENABLE, true);
if (Box) Box->draw(&CubeWorldMatrix, 0, CrateTex);
Device->SetRenderState(D3DRS_ALPHABLENDENABLE, false);
Device->EndScene();
Device->Present(0, 0, 0, 0);
}
return true;
}
```