

オブジェクト指向と ゲームプログラミング

DirectX Graphics 3D編 - 第9回 マルチテクスチャ

マルチテクスチャブレンド

複数のテクスチャを重ねることができます。このとき、加算、減算、乗算といった演算を用いることができます。テクスチャをさまざまな演算で重ねて合成することにより、表現の幅を大きく広げることができます。

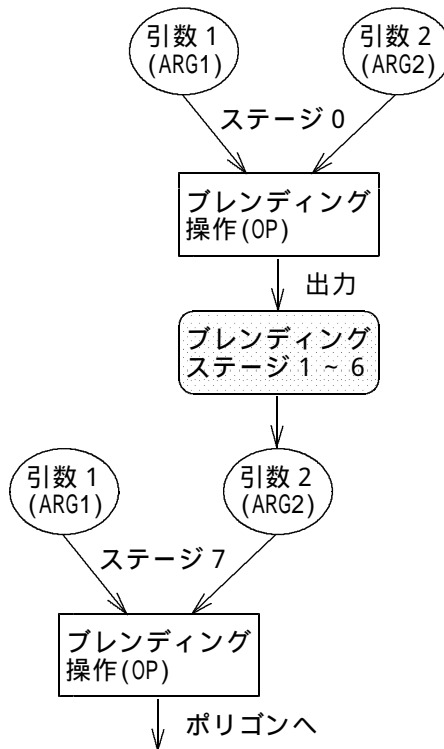
複数のテクスチャを重ねて合成することをマルチテクスチャブレンドといいます。一度のポリゴン描画で複数のテクスチャを重ねて合成することをシングルパスマルチテクスチャブレンドといい、複数回のポリゴン描画でテクスチャを重ねてマルチテクスチャを実現することをマルチパスマルチテクスチャブレンドといいます。

基本的にどの方法でも同じ画像を得ることができますが、その方法でしか表現できない画像もあります。シングルパスは一度の描画で済むため高速ですが、環境により重ねられる枚数に制限があります。マルチパスはテクスチャが適用されたポリゴンを複数重ねるため、その分だけ描画が必要になるため速度は低下しますが、どのような環境でも使用することができます。マルチテクスチャブレンドは、スプライトの描画にも用いることができます。

テクスチャステージ

シングルパスマルチテクスチャでは、ブレンド方法の設定をテクスチャステージで行います。テクスチャステージは、2つの引数(画像)を受け取り、ブレンド操作を行って新たな画像を出力します。最大で0~7ステージの8つのブレンドステージを使用することができます。サポートされるステージ数は環境によって異なるため、IDirect3DDevice9::GetDeviceCapsメソッドで調べる必要があります。

テクスチャステージによる合成は、以下の図で示すように行われます。



引数1の画像が上、引数2の画像が下になります。

テクスチャステージの設定

テクスチャステージによるブレンディング方法の設定は、IDirect3DDevice9::SetTextureStageStateメソッドで行います。使用するステージのブレンディング操作の方法、引数1、引数2の3つを個別に設定します。なお、色(RGB値)とアルファ値の合成方法は別々に設定しなければならないため、1つのステージにつき最大で6つのステートを設定する必要があります。

IDirect3DDevice9::SetTextureStageStateメソッド

- 説明 -

SetTextureStageStateメソッドは、テクスチャステージにステート値を設定します。

- 書式 -

HRESULT SetTextureStageState(DWORD Stage, D3DTEXTURESTAGESTATETYPE Type, DWORD Value);

- パラメータ -

1つ目の引数(Stage)は、設定するテクスチャのステージ識別子です。0～7までの数字を指定します。

2つ目の引数(Type)は、設定するテクスチャステートの指定です。テクスチャブレンディングでは、以下のものを使用します。

D3DTSS_COLOROP	色ブレンディング操作の方法
D3DTSS_COLORARG1	色ブレンディングの引数1
D3DTSS_COLORARG2	色ブレンディングの引数2
D3DTSS_ALPHAROP	アルファブレンディング操作の方法
D3DTSS_ALPHAARG1	アルファブレンディングの引数1
D3DTSS_ALPHAARG2	アルファブレンディングの引数2

3つ目の引数(Value)は、設定するステート値です。2つ目の引数により値の意味が変わります。テクスチャブレンディングでは、以下のものを使用します。

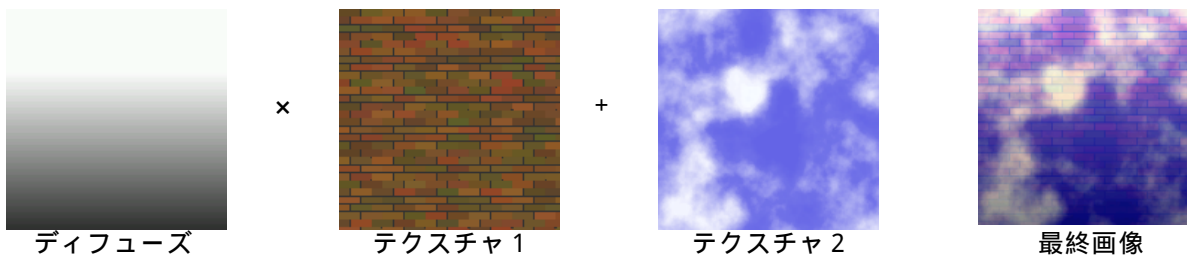
- 2つ目の引数がブレンディング操作(D3DTSS_COLOROPまたはD3DTSS_ALPHAROP)の場合
 - D3DTOP_DISABLE このテクスチャステージおよびこれ以降のステージからの出力をすべて無効にします。テクスチャマッピングを無効にするには、ステージ0にこの引数を設定します。色ブレンディングが有効な場合、アルファブレンディングをD3DTOP_DISABLEに設定すると、不定の動作が生じます
 - D3DTOP_SELECTARG1 引数1の色またはアルファ値を変更せずにそのまま出力します
 - D3DTOP_SELECTARG2 引数2の色またはアルファ値を変更せずにそのまま出力します
 - D3DTOP_MODULATE 引数1と引数2を乗算合成して出力します
 - D3DTOP_MODULATE2X 引数1と引数2を乗算合成し、2倍にしたものを出力します
 - D3DTOP_MODULATE4X 引数1と引数2を乗算合成し、4倍にしたものを出力します
 - D3DTOP_ADD 引数1と引数2を加算合成して出力します
 - D3DTOP_ADDSIGNED 引数1と引数2を加算合成し、0.5を引いて画素の値を-0.5から0.5の範囲にしたものを出力します
 - D3DTOP_ADDSIGNED2X 引数1と引数2を加算合成し、0.5を引いた結果を2倍にしたものを出力します
 - D3DTOP_SUBTRACT 引数1と引数2を減算合成して出力します
 - D3DTOP_ADDSMOOTH 引数1と引数2を加算合成したものと、引数1と引数2を乗算合成したものを減算合成して出力します
 - D3DTOP_BLENDDIFFUSEALPHA マテリアルのディフューズ色に設定されているアルファ値を使ってブレンディングを行います
 - D3DTOP_BLENDTEXTUREALPHA テクスチャのアルファ値を使ってブレンディングを行います
 - D3DTOP_BLENDCURRENTALPHA 直前のステージで使われたアルファ値を使ってブレンディングを行います
- 2つ目の引数が引数(D3DTSS_COLORARG1, 2またはD3DTSS_ALPHAARG1, 2)の場合
 - D3DTA_CURRENT 直前のステージで出力された画像を使用します。ステージ0の場合はディフューズ色が使用されます
 - D3DTA_DIFFUSE ディフューズ色を使用します
 - D3DTA_TEXTURE テクスチャを使用します。ただし、引数1にしか指定できません
 - D3DTA_SPECULAR スペキュラー色を使用します

- 戻り値 -

関数が成功するとD3D_OK、それ以外はエラーコードを返します。

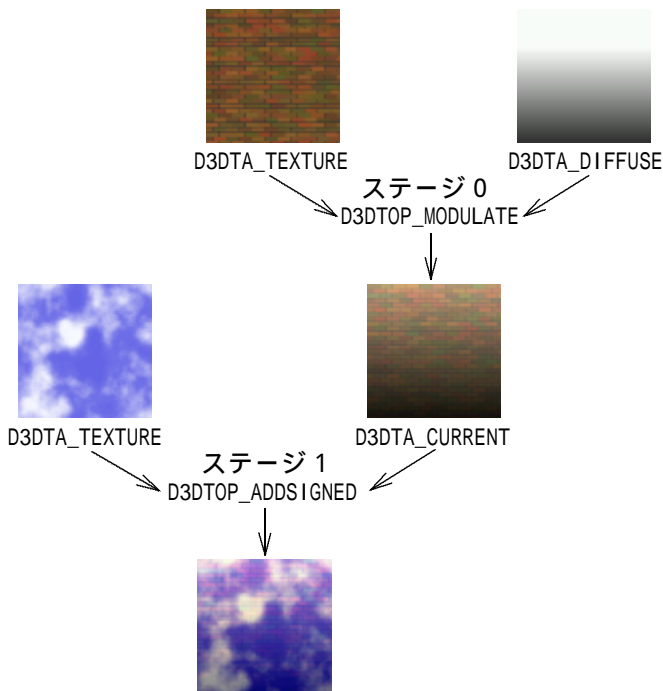
デフォルトでは、ステージ0のみ使用するようになっており、色のブレンディング方法は乗算(D3DTPMODULATE)、引数1はテクスチャ(D3DTA_TEXTURE)、引数2は直前のステージの画像(D3DTA_CURRENT)で、アルファ値の合成方法は引数1(テクスチャ)のものをそのまま出力するようになっています。

2枚のテクスチャを重ねて合成する場合、テクスチャステージを以下のように設定します。

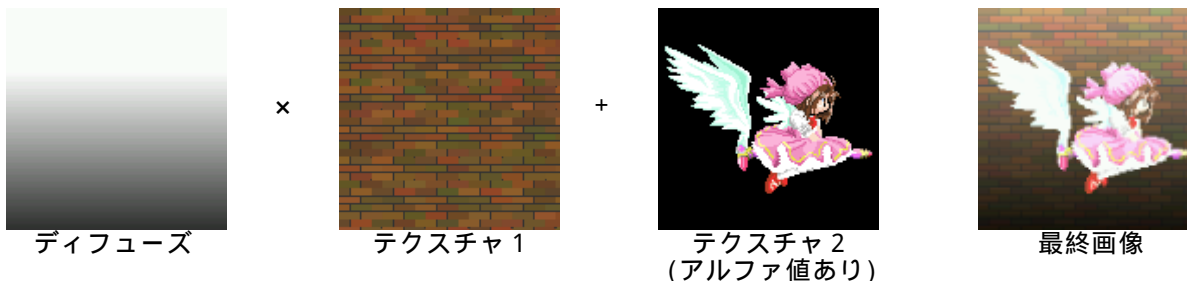


```
// テクスチャ設定(pD3DDevは初期化済みのDirect3DDevice9オブジェクト)
// ステージ0
pD3DDev->SetTextureStageState(0, D3DTSS_COLOROP, D3DTOP_MODULATE); // 色ブレンディング操作(乗算合成)
pD3DDev->SetTextureStageState(0, D3DTSS_COLORARG1, D3DTA_TEXTURE); // 色引数1(テクスチャ)
pD3DDev->SetTextureStageState(0, D3DTSS_COLORARG2, D3DTA_DIFFUSE); // 色引数2(ディフューズ)

// ステージ1
pD3DDev->SetTextureStageState(1, D3DTSS_COLOROP, D3DTOP_ADDSIGNED); // 色ブレンディング操作(加算合成)
pD3DDev->SetTextureStageState(1, D3DTSS_COLORARG1, D3DTA_TEXTURE); // 色引数1(テクスチャ)
pD3DDev->SetTextureStageState(1, D3DTSS_COLORARG2, D3DTA_CURRENT); // 色引数2(前ステージの画像)
```



また、以下のようにアルファ値を考慮して合成することもできます。



```
// テクスチャ設定
// ステージ 0
pD3DDev->SetTextureStageState(0, D3DTSS_COLOROP, D3DTOP_MODULATE); // 乗算合成
pD3DDev->SetTextureStageState(0, D3DTSS_COLORARG1, D3DTA_TEXTURE); // テクスチャ
pD3DDev->SetTextureStageState(0, D3DTSS_COLORARG2, D3DTA_DIFFUSE); // ディフューズ

// ステージ 1
pD3DDev->SetTextureStageState(1, D3DTSS_COLOROP, D3DTOP_BLENDTEXTUREALPHA); // アルファ合成
pD3DDev->SetTextureStageState(1, D3DTSS_COLORARG1, D3DTA_TEXTURE); // テクスチャ
pD3DDev->SetTextureStageState(1, D3DTSS_COLORARG2, D3DTA_CURRENT); // 前ステージの画像
```