SSAA11

AMD Developer Relations

Overview

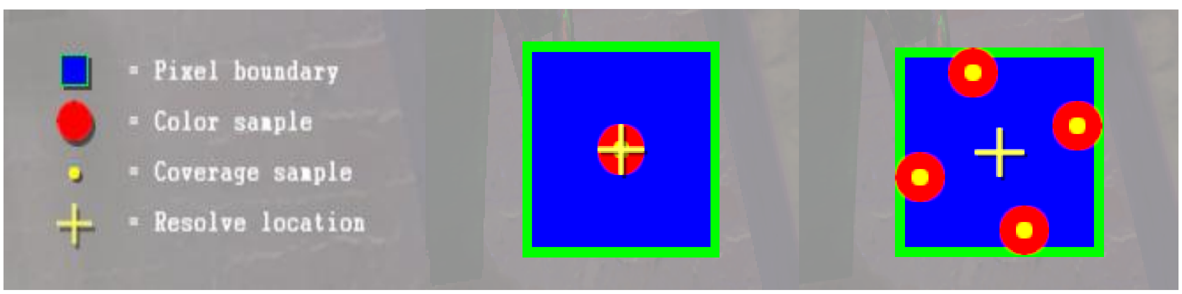
This sample demonstrates some of the ways in which to reduce scene aliasing without using a post process technique. Methods include multisample antialiasing (MSAA), supersample antialiasing (SSAA) and enhanced quality antialiasing (EQAA).

MSAA

By far the most popular technique for reducing aliasing is to use hardware multisampling. This works by rendering to a larger surface, say double width and double height for 4xMSAA, and resolving the result down to the destination buffer.

Modern hardware is optimized to run the color pixel shader once per pixel and, in the case of 4xMSAA, 4 depth values are evaluated. This means there is only a marginal performance cost to switching on MSAA.

Figure 1 shows that the color samples are actually taken in a rotated grid pattern which helps the cases where polygon edges are almost vertical or horizontal.



No AA 4xMSAA

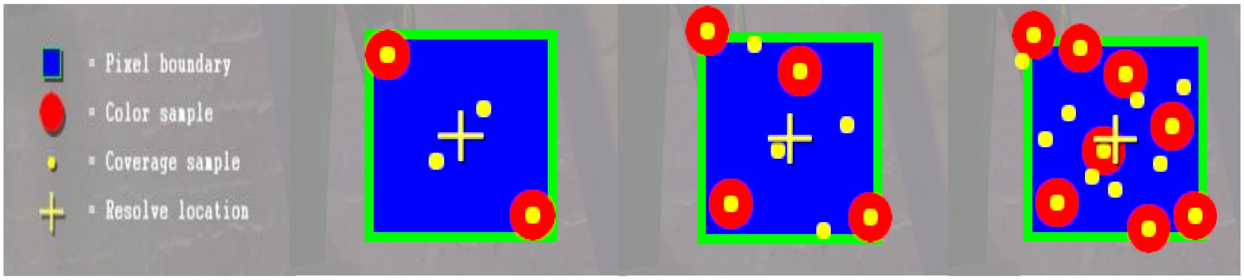
Figure 1.

EQAA

An extension of MSAA is to take more coverage samples than color and depth samples, thus increasing the final image quality with minimal extra performance cost and no extra memory overhead compared to standard MSAA (see Figure 2).

This technique is possible on Northern Islands architecture and above and requires checking both the GPU vendor id and calling `CheckMultisampleQualityLevels` to determine whether this technique is supported.

The original article on EQAA can be found here: http://developer.amd.com/



2f4x EQAA 4f8x EQAA 8f16x EQAA

Figure 2.

SSAA

Supersample antialiasing can be done one of two ways:

1. Manually using a larger non MSAA render target and then manually blitting the result into a destination buffer.

Or,

2. Using an MSAA render target but running the pixel shaders at sample frequency and calling ResolveSubresource to perform the resolve to the destination target.

Using a manual SSAA technique gives the user control of how to resolve the surface down. Typically this would be just a bilinear read of a block of 2x2 texels written to one destination pixel; however the sample demonstrates that other schemes could be used such as the one labeled “SSAA RG”. This performs four bilinear reads per destination pixel in a rotated grid fashion and reads outside of the 2x2 texel block. The result is generally smoother at the cost of a blurrier final result.

Performing the SSAA at the sample frequency has the advantage that the samples are taken in a rotated grid fashion whereas manually they are in a uniform grid layout. In order to execute pixel shaders at the sample frequency, at least one interpolator must have the “sample” modifier applied to it (see Figure 3).

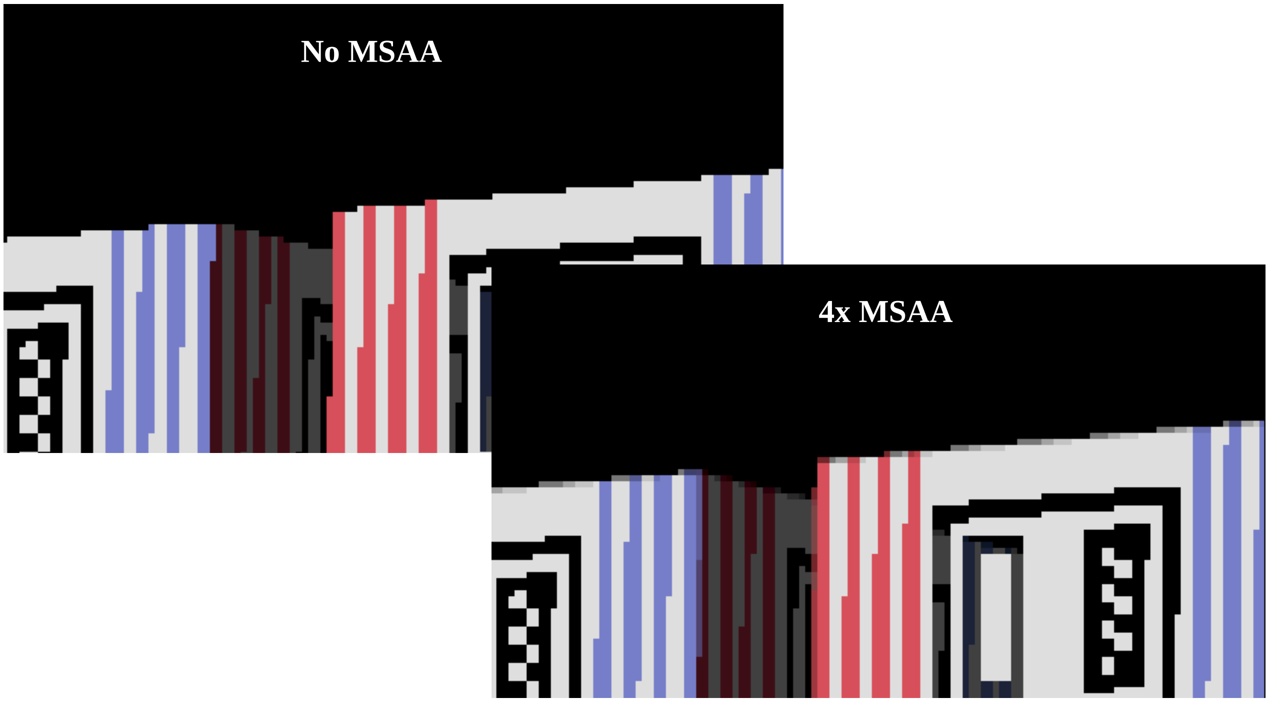
```
#if defined (PER_SAMPLE_FREQUENCY)
struct PS_INPUT
{
    centroid sample float3      normal      : NORMAL;
    centroid sample float3      tangent     : TANGENT;
    sample float2               texcoord    : TEXCOORD0;
    centroid sample float3      worldPos    : TEXCOORD1;
};
#else
struct PS_INPUT
{
    centroid float3 normal      : NORMAL;
    centroid float3 tangent     : TANGENT;
    float2          texcoord    : TEXCOORD0;
    centroid float3 worldPos    : TEXCOORD1;
};
#endif
```

Figure 3.

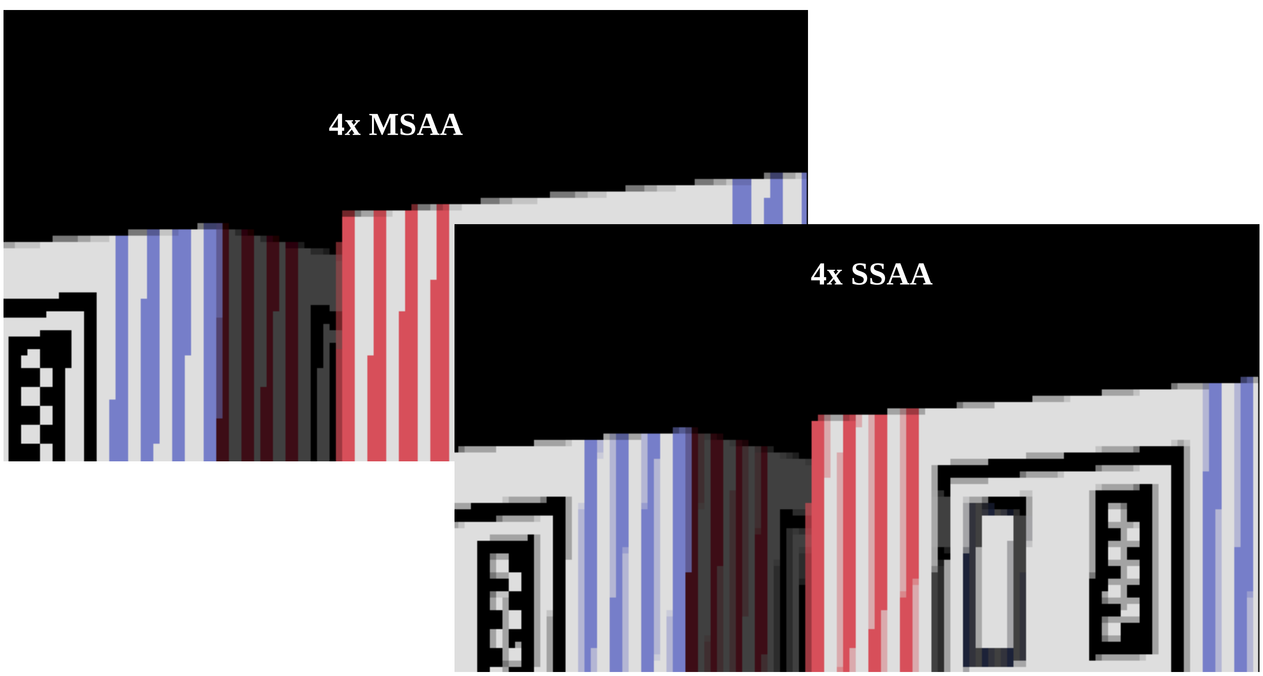
Mip LOD Biasing

Pixel derivatives are calculated by the hardware on 2x2 pixel quads. These derivatives are then used to calculate the mip map levels for each filtered texture read. When using per-sample supersampling the derivatives are still only calculated per pixel rather than per sample. This means that by default textures will appear more blurred compared to standard supersampling. To compensate for this mip LOD biasing needs to be applied. For 4xSSAA this should be -1 and at 8xSSAA this should be -1.5.

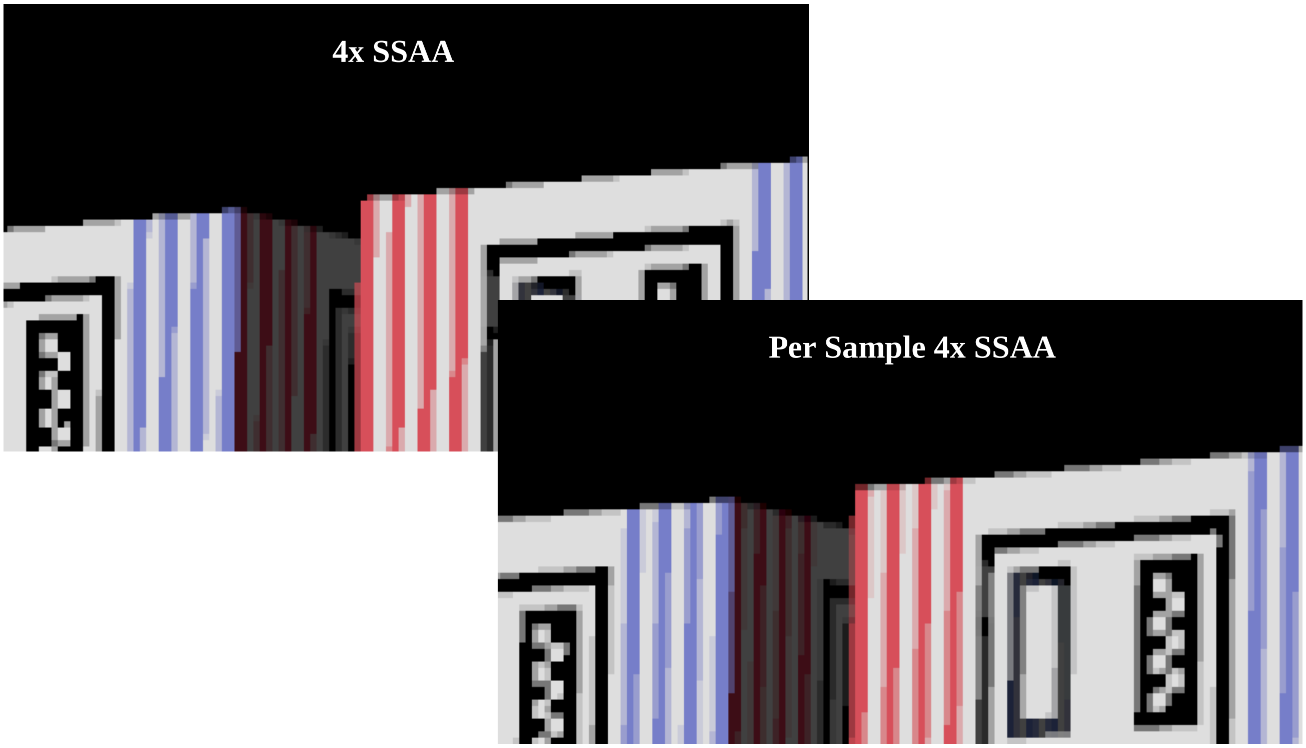
Image Quality



No MSAA versus 4xMSAA. Note the polygon edge antialiasing but no improvement in alpha tested edges or internal triangle regions.



4xMSAA versus 4xSSAA. Note worse edge AA but demonstrates alpha tested and shader AA.



4xSSAA versus per-sample 4xSSAA. Note better edge and shader AA.

Performance

Technique	Scene Cost (ms)	Resolve Cost (ms)	Total Frame Time (fps)
None	1.3	0.15	529
4x MSAA	1.39	0.35	463
4x SSAA	4.26	0.4	192
4x SSAA Sample Freq.	4.34	0.79	177

Performance measured on SSAA11 demo “typical scene” using Radeon HD7970 GHz Edition at 2560x1600 with RGBA8 render target.

Advanced Micro Devices
One AMD Place
P.O. Box 3453
Sunnyvale, CA 94088-3453

www.amd.com
http://ati.amd.com/developer