navigation

- Wiki Main
- Forum
- Recent changes
- Random page
- Help

categories

- General
- Aircraft Design
- Scenery Design
- Missions
- SimConnect
- File Formats
- Tools
- Manuals
- Video Tutorials

search

Search

Go Search

tools

DXT compression explained

Applicable
MS Flight
FSXA
FSX
FS2004
FS2002

Since the introduction of the compressed DXT texture types in Flight Simulator, choosing the right type of compression has often lead to discussion. What are the differences between DXT1 and DXT3 for example? The latter one gives me a bigger filesize, so does that not mean better quality of my texture?

This article hopes to explain the differences between the different formats, so that you can make a good choice which compression to use. Also will it explain how the compression itself works, so that you understand the influence it has on the quality of your texture.

Contents

1 DXT1
2 DXT1 with alpha
3 DXT3
4 DXT5
5 Related
5.1 Internal
5.2 External

DXT1

The first compression we will look at is DXT1. This is the most simple compression and also the basis for the other types.

But let's start with a note about compression itself. You have probably noticed that the DXT compressed textures always have the same size. This is because they have a fixed compression ratio. Because the resulting size is fixed, it also means that the quality loss of the compression depends on the original texture

- What links here
- Related changes
- Special pages
- Printable version
- Permanent link
- Page information

you used.

When storing the RGBA information of your texture each pixel requires 32 bit to store the information. For DXT compressions the texture is split into segments of 4x4 pixels which are then compressed. These 16 pixels take 512 bits to store without the compression.

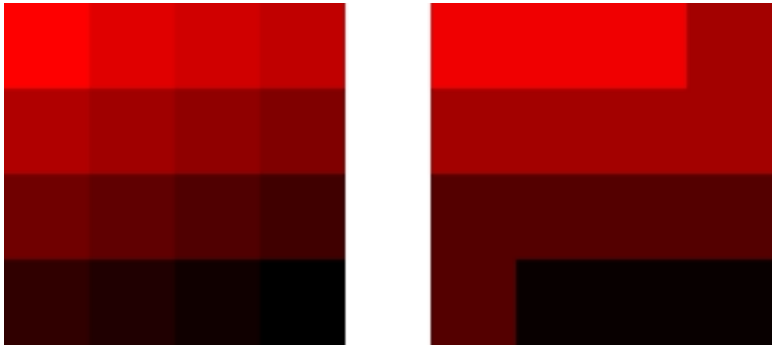
When compressing a 4 colour palette is determined for these 16 pixels. Afterwards each pixel will get an index into this palette, which only requires 2 bit per pixel. For the palette only two colours are stored, the two extremes, and the other two colours are interpolated between these extremes. The colour information is also stored with a compression so that only 16 bits are used per colour. This means that these 16 pixels of the texture only take 64 bits to store (32 for the palette and 32 for the indexing). That is a compression ratio of 1:8.

But let's take a step back, what does this all mean for the colour information of your original texture? First we will take a look at the palette. It uses 16 bit to store the colours, instead of the normal 8 bit per colour. It uses a 565 compression for this, which means that red and blue are stored with 5 bit and green with 6 bit. This thus means that you can use less colours in your texture and if you are using shades of the same colour that are very similar these will be lost in the compression.

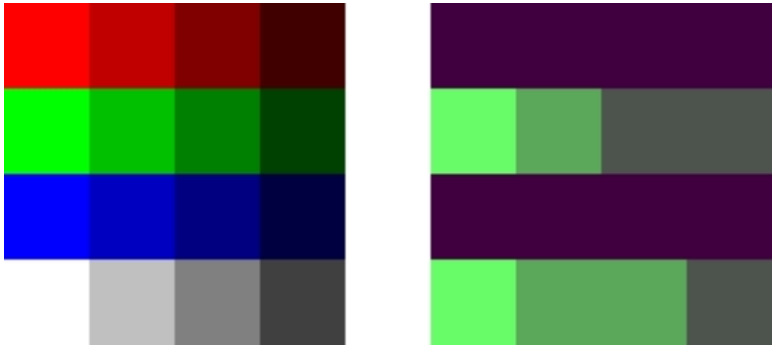
Besides that the palette only stores two colour, let's call them colour1 and colour2. The other two colours that can be indexed are linearly interpolated. So that means that colour3 is 2/3 of colour1 plus 1/3 of colour2 and for colour4 the reverse is true (1/3 if colour1 plus 2/3 of colour2). This means that if the 4 most common colours used in your 16 pixels are not on a one linear line in the colour space, you will loose some colours.

Below are some images to clarify what the compression does to the quality of your texture. To make it more clear a region of 4x4 pixels has been blown up to a bigger size in the images.

This image shows how the compression affects colours. On the left you see 16 shades of red, ranging from pure red to pure black. On the right you see the four resulting colours that the DXT compression choose to represent these 16 pixels.



This image shows what happens when the different colour are not on a linear line in colour space. In this case all extremes (red, green and blue) have been used. It is clear the resulting interpolated colours not match the originals at all. Normally an area of 4x4 pixels would not have so wide variety of colours, but it illustrates that textures with many different colours are affected more.



DXT1 with alpha

The above holds when your texture does not have any transparency, if you want to add a 1 bit alpha channel things change slightly. A 1 bit alpha channel means that the transparency is either on or off. To store this one of the colours of the colour palette will get the meaning completely transparent.

The other three colours are left to store the colour information of your pixels. So for the two colours stored in this case only one has to be interpolated, colour3 is 1/2 colour1 plus 1/2 colour2. This means even less resolution is left for different shades of colours.

DXT3

The colour information in DXT3 is saved similar to the DXT1 format without an alpha channel. So a palette of 4 colours is used to represent 16 pixels of the texture. But the alpha information is stored with 4 bit in the DXT3

format. That means that in total 128 bit are used for the 16 pixels (32 for the palette, 32 for the colour indices and 64 for the alpha information). So this results in a compression ratio of 1:4.

Because the alpha information is stored in 4 bit, the smallest step in the alpha value that can be stored is 16. This means that very smooth transitions in the alpha channel can not be stored in the DXT3 format.

DXT5

The DXT5 formats differs from the DXT3 format in the way the alpha information is saved. The colour information is stored in the same way.

For the alpha information it uses a palette, similar to the way the colour information is also stored. This palette contains a minimum and maximum alpha value. Then 6 other alpha values are interpolated between this minimum and maximum. This thus allows more gradual changes of the alpha value.

A second variant does interpolate only 4 other alpha values between the minimum and maximum, but also adds an alpha value of 0 and 1 (for fully transparent and not transparent). For some textures this might give better results.

Related

Internal

Texture - Related	
Quick links to related topics - Transclusion-Texture-Oneline	
General	Night · Perspective - Correcting (GIMP) · Seasonal textures · Taxilines with Gmax · Reflection and scenery · Bump Maps · Disappearing textures
Transparency	Problem solving · PhotoShop
Performance	Optimising for performance · DrawCallMonitor · Texture - Missing (FSX)
File Formats	Formats (Overview) · **DXT Compression** · Texture.cfg (FSX)

SDK - FSX Modeling SDK (FSX) · Panels and Gauges SDK (FSX) · SimObject Container SDK (FSX)

SDK SDK Toolset (FS9) · SDK Toolset (FS8)

Click the links to access the topic pages, (Default - Opens in same window).

External

MSDN DirectX SDK Compressed Texture Resources

Categories: FSX Acceleration | FSX | FS9 (FS2004) | Texturing | Scenery Design | Aircraft Design