



3D Content Creation Guidelines

Document information

INFO	CONTENT
Title	3D Content Creation Guidelines
Date	29-Mar-2011 10:09 PM
Security	The attached material and the information contained herein are proprietary to Dimenco. Copying, reproduction, adaptation, modification or dissemination in whole or part is not permitted without written permission from Dimenco.
Contact address	Dimenco De Run 4281 5503 LM Veldhoven The Netherlands, E-mail support: support@dimenco.eu E-mail general: info@dimenco.eu Website: www.dimenco.eu

Table of contents

Document information.....	2
1 Introduction.....	2
2 Content creation chain.....	3
2.1 2D + Z image format.....	3
2.2 Concatenate images.....	4
2.3 Encoding to video file.....	4
2.4 Windows Media Video.....	5
2.5 The s3d File Extension.....	6
3 Declipse technology.....	7
3.1 Declipse format.....	8
4 Stretched images.....	9
Appendix: MPEG-2.....	10

1 Introduction

This document holds guidelines for creating content for 2D+Z enabled 3D Displays. This document explains step by step, the required actions to make images and video files, which can be played on a 2D+Z Display. No expensive tooling is required. This documents serves merely as an introduction to the 2D+Z file format and as a first step into domain of 2D+Z. The content will be created with an aspect ratio of 16:9.

2 Content creation chain

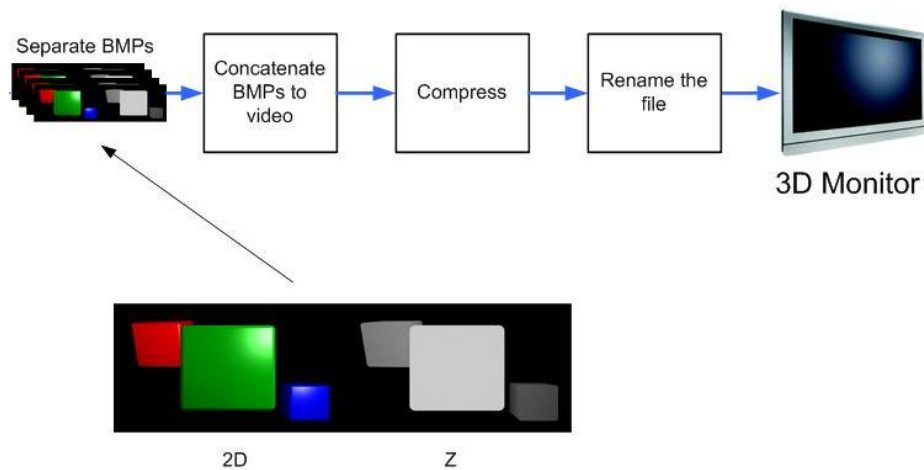


Figure 1: 3D content creation steps

The interface is based on a 2D and Z image. Adjacent to the traditional 2D image a Z image is added, also named depth map. This is an image with the same size as the 2D image. Each pixel of the depth map corresponds to a pixel of the 2D image and indicates the distance of the corresponding 2D pixel to the observer.

Several steps need to be made in order to create content for a 2D+Z Display. 3D Studio Max or any other 3D design software can be used to edit the scene as one is used to do. These tools can generate the 2D + Z images by tweaking them a bit. Finally the images are concatenated to a video file and the video file is compressed. As a final step the files are renamed from .wmv to .s3d.

The Dimenco 3D Player also support the playback of images. These images are RGB (24 bit uncompressed) bmp files and have the extension .b3d.

2.1 2D + Z image format

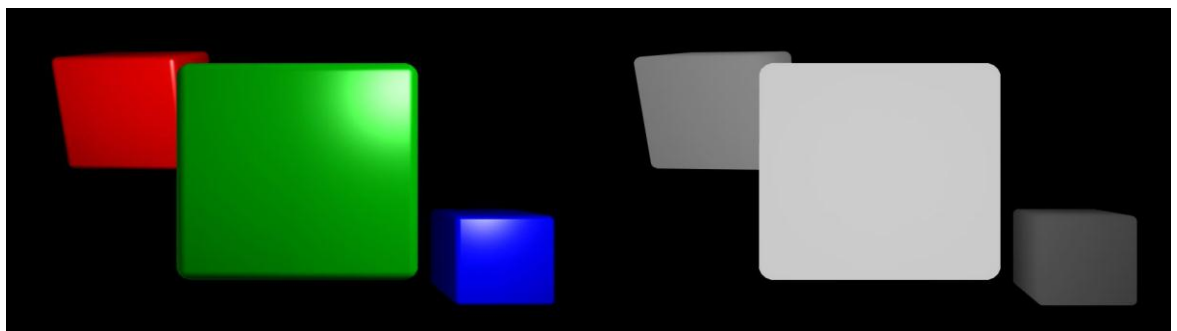


Figure 2: A bitmap, left half 2D, right half Z map.

2.4 Windows Media Video

For owners of Windows XP operating system, the encoder can be downloaded from Microsoft free of charge from:

<http://www.microsoft.com/windows/windowsmedia/9series/encoder/default.aspx>

The batch file below assumes that the WMV encoder is installed in its default path. After this encoder is installed you can encode the content using

```
encodewmv source.avi destination.s3d
```

```
@echo off

set WMVPATH="C:\Program Files\Windows Media Components\Encoder\"

set WMENCODER=%WMVPATH%"wmcmd.vbs"

rem Standard WMV video only profile

set VIDEO_QVBR_100=%WMVPATH%"Settings\d6_qvbr_100_video.prx"

cscript.exe %WMENCODER% -input %1 -output %2 -loadprofile %VIDEO_QVBR_100%
```

Figure 3 encodewmv.bat

Please see the help file of the Windows Media Encoder for the other command line options for the encoder. In figure 3 the encodewmv.bat will encode the content with the highest quality possible (ensure that the bit rate is not too high for your application). Other quality settings are possible to e.g. constraint the bit rate as seen in figure 4 which can result in artifacts.

```
@echo off

set WMVPATH="C:\Program Files\Windows Media Components\Encoder\"

set WMENCODER=%WMVPATH%"wmcmd.vbs"

rem Everything below this comment should be on a single line
cscript.exe %WMENCODER% -input %1 -output %2 -a_codec WMA9PRO
-v_mode 4 -v_bitrate 8000000 -v_peakbitrate 10000000 -v_peakbuffer
80000
```

Figure 4 encodewmv_with_peak_of_10Mbit.bat

The batch file shown in Figure 3 does not support audio. In Figure 4 a batch file is shown which does support an AVI input file that contains audio.

2.5 The s3d File Extension

The .s3d file created in the previous chapter is available after WMV encoding. This file can be played on the 2D+Z display. If the content contains a depth map it can be played using the Dimenco 3D Player giving a 3D impression. The depth impression is generated using images from multiple angles. These images are generated by hardware in the 2D+Z Display.

Files using the extension .s3d are played with the Dimenco 3D Player. This extension makes sure that the Dimenco 3D Player sets the 2D+Z Display in 3D mode with the correct settings.

3 Declipse technology

Using the 2D+Z image format a quite astonishing 3D experience can be created. But on locations with large depth differences an undesired effect can be visible in the outer views of the viewing cone. On these locations you should be able to see around an object, but because only one picture is available the information on what should be rendered is not available.

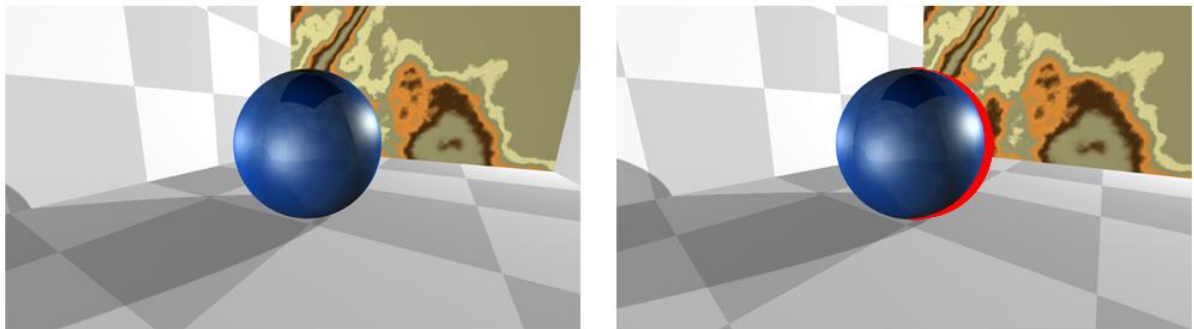


Figure 5 Left and right two views on the same scene. Red denotes missing information.

Figure 5 shows the location in red where the information is missing. As no information is available the rendering algorithm in the 2D+Z display masks this by reproducing the contents of this area based on the surrounding information. To enable a crisper image and better quality the Declipse technology can be used. With Declipse extra information is added enabling a better 3D experience, the image above would then be visible as in Figure 6.

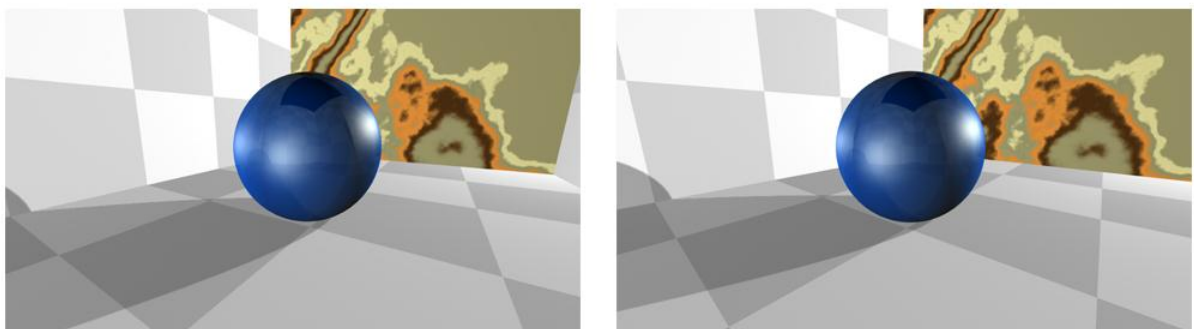


Figure 6 Left and right the same different views with Declipse technology

3.1 Declipse format

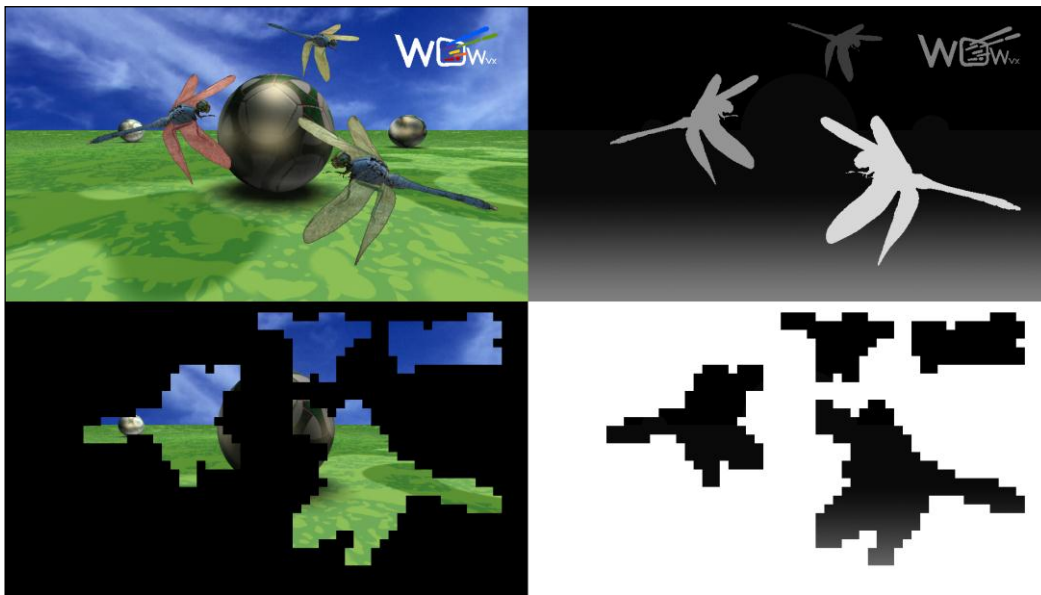


Figure 7 Bitmap showing on top complete 2D image (left) and depth (right). On the bottom background 2D image (left) and depth (right).

The Declipse format is an optional extension on the existing 2D+Z format described in section 2.1. Information on the background is added enabling the rendering algorithm filling in the occluded areas created by the foreground object. Figure 7 shows the four quadrants that should be supplied to the screen. The quadrants specify the following information:

Description	16:9
The total image has a resolution of	1920x1080
The top left: a 2D RGB picture with a resolution of	960x540
The top right: a depth map in grey scale picture with the resolution (This depth map belonging to the top left 2D image)	960x540
The bottom left: 2D RGB picture of the background with resolution. (On the places where the 2D information is equal to the quadrant above (top left) the 2D information is set to black to avoid encoding redundant data. To obtain optimal compression the information stops at the boundaries of (16x16) macro blocks.)	960x540
The bottom right: the depth map on the areas occluded by the foreground objects. This depth map relates to the bottom left 2D image (On the places where the disparity is equal to the quadrant above (top right) the disparity is set to maximal (white) to avoid encoding redundant data. To obtain optimal compression the information stops at the boundaries of (16x16) macro blocks.)	960x540

4 Stretched images

While working in 3D design software, some design rules must be kept in mind. Section 4 describes how to deal with image stretching done by the display.

If an object is positioned at the left edge of the image has a depth that puts it behind the screen, a viewer looking at it from the right would expect to see more pixels to the left of the object where the frame of the display would no longer be concealing the object. Because these pixels are not in the original image, the image is stretched slightly before sending it to rendering to prevent this problem.

Error! Reference source not found. shows how the image is stretched. Stretching and rendering takes place inside the display.

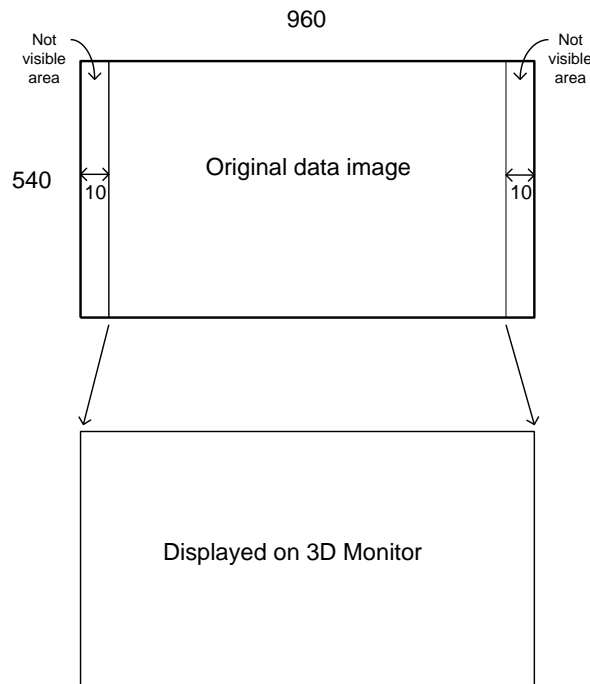


Figure 8: A band at the left and right side of the image data are not displayed on the 3D displays.

The original input data (excluding depth) has a resolution of 960x540. The 10 most left pixels and the 10 most right pixels of each row are not visible on the display, when the depth of the pixels is on the screen. However when the depth is behind the screen, the observer is able to see this information in these bands.

So while creating content, the editor must keep in mind that small bands at the left and right side will mostly not be visible. Especially for text and logo's located closely to left and right borders of the screen the placement must be done carefully.

Appendix: MPEG-2

It is also possible to create content that is compressed using MPEG-2. Be aware that a MPEG-2 decoder is not installed by default. If no MPEG-2 decoder is installed you can advice GPL MPEG-1/2 DirectShow Decoder Filter (<http://sourceforge.net/projects/gplmpegdec>). Installing multiple MPEG-2 decoders can lead to incompatibility problems between video and audio. Using the WMV format (see chapter 2.3) results in a playback supported by the Windows XP operating system.

Install ffmpeg (<http://www.videohelp.com/download/ffmpeggui03c.zip>). The batch files below assume this application is installed in 'C:\tools'. Using these batch files (encode16_9.bat & encode4_3.bat) with the contents as below you can compress the created video file using the command:

```
encode16_9 source.avi destination.s3d
```

```
@echo off
"C:\tools\ffmpeg.exe" -i "%1" -hq -vcodec mpeg2video -b 24000 -bt 16000 -
aspect 32:9 -s 1920x540 -y "%2".vob
```

```
move "%2".vob "%2"
```

Figure 9 encode16_9.bat

Some handy parameters:

Parameter	Description	Choices
-acodec	select audio codec	mp1 mp2 (default) ac3
-ab	bitrate in kb/s	-
-r	framerate	

There are some things to take in consideration:

The current compression takes over the frame-rate of the source. When this is out of spec ffmpeg will give an error. Using the `-r` option the frame rate can be set. The default audio codec is mp2 audio at 64 kb/s. You can set another audio codec using the parameter `-acodec`. Please be aware that the system playing back the created file should have the proper audio codec installed.