**MFC DEPTH Reference Software Manual**

**TABLE OF CONTENTS[[1]](#footnote-1)**

Page #

Contents

[MFC DEPTH Reference Software Manual i](#_Toc382923617)

[TABLE OF CONTENTS i](#_Toc382923618)

[Page # i](#_Toc382923619)

[1. General Information 1-1](#_Toc382923620)

[1. General Information 1-1](#_Toc382923621)

[1.1 System Overview 1-1](#_Toc382923622)

[1.2 Acronyms and Abbreviations 1-1](#_Toc382923623)

[2. Installation and Compilation 2-1](#_Toc382923624)

[2. Installation and Compilation. 2-1](#_Toc382923625)

[2.1 Windows using MS Visual Studio .NET 2-1](#_Toc382923626)

[2.2 UNIX and Windows using gcc (GNU Compiler Collection) 2-1](#_Toc382923627)

[3. Using The MFC DEPTH Encoder Module 3-1](#_Toc382923628)

[3. Using The MFC DEPTH Encoder Module 3-1](#_Toc382923629)

[3.1 Encoder Syntax 3-1](#_Toc382923630)

[3.2 Encoder Output 3-1](#_Toc382923631)

[3.3 Encoder Limitations 3-5](#_Toc382923632)

[4. MFC Depth Encoder Parameters 4-1](#_Toc382923633)

[4. MFC Depth encoder Parameters 4-1](#_Toc382923634)

[4.1 MFC texture coding parameters (main configuration file) 4-1](#_Toc382923635)

[4.1.1 ProcessInput 4-1](#_Toc382923636)

[4.1.2 ProfileIDC 4-1](#_Toc382923637)

[4.1.3 Rpu\_Filter\_Enabled\_Flag 4-1](#_Toc382923638)

[4.1.4 MFC\_Format\_Idc 4-1](#_Toc382923639)

[4.1.5 DefaultGridPosition 4-2](#_Toc382923640)

[4.1.6 View0\_Grid\_Position\_x 4-2](#_Toc382923641)

[4.1.7 View0\_Grid\_Position\_y 4-2](#_Toc382923642)

[4.1.8 View1\_Grid\_Position\_x 4-2](#_Toc382923643)

[4.1.9 View1\_Grid\_Position\_y 4-2](#_Toc382923644)

[4.1.10 Mux3DBaseFilter 4-2](#_Toc382923645)

[4.1.11 Mux3DEnhFilter 4-2](#_Toc382923646)

[4.1.12 EnableDbgYUVFiles 4-3](#_Toc382923647)

[4.1.13 FPASEIPresentFlag 4-3](#_Toc382923648)

[4.1.14 3DVCoding 4-3](#_Toc382923649)

[4.1.15 3DVCodingOrder 4-3](#_Toc382923650)

[4.2 MVC depth coding parameters (main configuration file) 4-3](#_Toc382923651)

[4.3 MVC coding parameters (View 1/Enhancement layer configuration file) 4-3](#_Toc382923652)

[5. MFC DEPTH Hardcoded Encoder Parameters 5-1](#_Toc382923653)

[5. MFC DEPTH Hard coded Encoder Parameters 5-1](#_Toc382923654)

[5.1 defines.h 5-1](#_Toc382923655)

[6. Using The MFC DEPTH Decoder Module 6-1](#_Toc382923656)

[6. Using The MFC DEPTH Decoder Module 6-1](#_Toc382923657)

[6.1 Decoder Syntax 6-1](#_Toc382923658)

[6.2 Decoder Configuration File Format 6-1](#_Toc382923659)

[6.2.1 ExportViews 6-1](#_Toc382923660)

[6.2.2 DeMuxMode 6-1](#_Toc382923661)

[6.2.3 EnableDbgYUVFiles 6-2](#_Toc382923662)

[6.2.4 OutputRecYUVFile 6-2](#_Toc382923663)

[6.2.5 OutputFileForTexture 6-2](#_Toc382923664)

[6.2.6 OutputFileForDepth 6-2](#_Toc382923665)

[6.2.7 WriteUVForTexture 6-2](#_Toc382923666)

[6.2.8 WriteUVForDepth 6-2](#_Toc382923667)

[6.3 Decoder Output 6-2](#_Toc382923668)

[7. MFC DEPTH Hardcoded Decoder Parameters 7-1](#_Toc382923669)

[7. MFC DEPTH Hardcoded Decoder Parameters 7-1](#_Toc382923670)

[7.1 defines.h 7-1](#_Toc382923671)

# General Information

1. General Information

## System Overview

This document contains a detailed description of the usage of the MFC (Multi-resolution Framce Compatible Stereo Coding) Depth software built upon version 18.3 of the H.264/14496-10 MPEG-4 AVC JM reference software. The software also incorporated bugfixes from JM18.6. The document includes information about the encoder and decoder input parameters, syntax, compilation issues, and additional information with regards to the best usage and configuration of this software.

## Acronyms and Abbreviations

* + 1. **AVC**: Advanced Video Coding
    2. **FC**: Frame Compatible
    3. **FCFR**: Frame Compatible Full Resolution
    4. **FR**: Full Resolution
    5. **MFC:** Multi-resolution Frame Compatible Stereo Coding
    6. **MFCD:** Multi-resolution Frame Compatible Stereo Coding Plus Depth
    7. **MVC:** Multi-view Video Coding
    8. **OM**: Orthogonal Muxing
    9. **RPU**: Reference Processing Unit
    10. **SbS**: Side-by-Side
    11. **TaB**: Top-and-Bottom

# Installation and Compilation

2. Installation and Compilation.

## Windows using MS Visual Studio .NET

The software package contains a Visual Studio .NET workspace named “jm\_vc10.sln” for .NET 2010 (v10). The workspace includes the following projects:

*lencod* the H.264/AVC reference encoder

*ldecod* the H.264/AVC reference decoder

*rtpdump* a tool for analyzing contents of RTP packets

*rtp\_loss* a tool for simulating RTP packet losses

*rpu\_sdk* an SDK static library which contains functions related to the MFC. The SDK is split into encoder\_layer, rpu\_layer , decoder\_layer and utilities\_layer. The library can be compiled and generated under directory rpu\_sdk.

Select the desired project and the appropriate compilation mode, i.e. “Debug” or “Release”, and platform, i.e. “Win32” or “x64”. Compilation will create the binaries “lencod\_[mode]\_[platform].exe” or “ldecod\_[mode]\_[platform].exe” in the “bin” directory. “rtpdump.exe” and “rtp\_loss.exe” will be created in the “bin” directory.

For compile time settings and options see section 5.

For faster execution it is recommended to compile and run the software in Release mode and on an x64 platform. For older, 32-bit only capable systems, only the Win32 platform can be used.

Note: The software has been primarily tested on an x64 environment.

## UNIX and Windows using gcc (GNU Compiler Collection)

For compiling the encoder, the decoder, the rpu\_sdk, rtpdump and rtp\_loss:

make

Binaries named “lencod.exe” ,“ldecod.exe”, “rtp\_loss.exe” and “rtpdump.exe” will be created in the “bin” directory. For debug mode binaries one can compile the software using the following syntax:

make DBG=1

The above would generate debug binary files named “lencod.dbg.exe”, “ldecod.dbg.exe”, “rtp\_loss.dbg.exe” and “rtpdump.dbg.exe” in the “bin” directory.

By default, the binary executables will be created according to the architecture of the system, where the compilation is performed, i.e. a 32-bit executable will be created on a 32-bit system, while a 64-bit executable will be created on a 64-bit system. However, the compilation parameter M32, if set to 1, will enforce the generation of a 32-bit executable, even on a 64-bit architecture system. Additional options that can be used during compilation include OPT=N, which can control the compilations optimization level, and STC=1 for static linking of libraries.

For compile time settings and options see section 5.

Note: The software has been primarily tested on in linux with gcc version 4.5.1.

# Using The MFC DEPTH Encoder Module

3. Using The MFC DEPTH Encoder Module

This section provides a detailed description of the MFC related parameters in the H.264/MPEG-4 AVC JM encoder.

Note: Current encoder requires both the texture views and depth views being progressive and having the same resolution.

## Encoder Syntax

lencod.exe [-h] [-d defenc\_texture.cfg] [-depd defenc\_depth.cfg]

|  |  |
| --- | --- |
| ***Options:*** | |
| *-h* | Prints parameter usage. |
| *-d* | Use <encoder\_mfc\_texture.cfg> as the default file for texture encoder parameter initialization. |
| *-depd* | Use <encoder\_mfc\_depth.cfg> as the default file for texture encoder parameter initialization. |

See section 4 for a description of all parameters.

Supported video file formats:

The software only supports MFC Depth High Profile. Therefore, it has only been tested using planar 8-bit 4:2:0 raw YUV video data, with all frames packed in a single file.

Examples of usage:

lencod.exe -d encoder\_mfc\_texture.cfg**[[2]](#footnote-2)** -depd encoder\_mfc\_depth.cfg**2**

## Encoder Output

When running the encoder, the encoder will display on screen rate/distortion statistics for every frame coded. Cumulative results will also be presented. The output information generated may look as follows depending on the setting of the Verbose input parameter:

Setting Default Parameters...

Parsing Configfile encoder\_mfc\_texture.cfg.....................................

..........................................................................

...............................................................................

........................................................

Parsing Second View Configfile encoder\_mfc\_texture\_view1.cfg...................

....................................

MFC HIGH PROFILE :DefaultGridPosition is set. Default values are being set for

he view offsets.

Warning: Hierarchical coding or Referenced B slices used.

Make sure that you have allocated enough references

in reference buffer to achieve best performance.

Setting Default Parameters...

Parsing Configfile encoder\_mfc\_depth.cfg.......................................

.......................................................

...............................................................................

.........................................

Parsing Second View Configfile encoder\_mfc\_depth\_view1.cfg.....................

..................................

Warning: Hierarchical coding or Referenced B slices used.

Make sure that you have allocated enough references

in reference buffer to achieve best performance.

----------------------------------------- MFC-Depth 18.3 (FRExt) --------------

----------------

Texture Input YUV file 1 : K:\data\mfc\_depth\LowRes\Dancer\_c\_5\_480x272

yuv

Texture Input YUV file 2 : K:\data\mfc\_depth\LowRes\Dancer\_c\_9\_480x272

yuv

Depth Input YUV file 1 : K:\data\mfc\_depth\LowRes\Dancer\_d\_5\_480x272

yuv

Depth Input YUV file 2 : K:\data\mfc\_depth\LowRes\Dancer\_d\_9\_480x272

yuv

Output H.264 bitstream : test.264

Texture Output YUV file 1 : texture\_rec.yuv

Texture Output YUV file 1 : texture\_rec2.yuv

Depth Output YUV file 1 : depth\_rec.yuv

Depth Output YUV file 1 : depth\_rec2.yuv

YUV Format : Texture (YUV 4:2:0) Depth (YUV 4:2:0)

Frames to be encoded : 2

Freq. for encoded bitstream : 24.00

Transform8x8Mode : Texture( 1 ) Depth( 1 )

ME Metric for Refinement Level 0 : Texture( SAD ) Depth( SSE )

ME Metric for Refinement Level 1 : Texture( SAD ) Depth( SSE )

ME Metric for Refinement Level 2 : Texture( SAD ) Depth( SSE )

Mode Decision Metric : Texture( SAD ) Depth( SSE )

Texture Image format : 480x272 (480x272)

Depth Image format : 480x272 (480x272)

Search range : 96

Total number of references : 4

References for P slices : Texture( 4 ) Depth( 3 )

References for B slices (L0, L1) : Texture( 3, 1 ) Depth( 2, 1 )

Sequence type : Hierarchy Texture( QP: I 22, P 22, B 22 )

Depth( QP: I 26, P 26, B 26 )

Entropy coding method : CABAC

Texture Profile/Level IDC : (134,51)

Depth Profile/Level IDC : (135,51)

Texture RD-optimized mode decision: used

Depth RD-optimized mode decision : used

Output File Format : H.264/AVC Annex B Byte Stream Format

-------------------------------------------------------------------------------

----

Frame View TId Bit/pic QP SnrY SnrU SnrV Time(ms) MET(ms) Fr

/Fld Anchor/Non-Anchor Ref

-------------------------------------------------------------------------------

----

00000(NVB) 360

00000(IDR) 0 171704 22 42.357 45.838 46.026 1311 0 FRM

3

00000(NVB) 240

00000(IDR) 0 7888 26 49.027 93.268 93.268 687 0 FRM

3

00000( P ) 1 2712 26 49.363 93.268 93.268 777 29 FRM

2

00000( P ) 1 50200 24 40.122 46.543 47.326 1099 60 FRM

2

00001( P ) 0 31296 22 41.447 45.704 45.960 1203 40 FRM

2

00001( P ) 0 1360 26 49.364 93.268 93.268 688 22 FRM

2

00001( P ) 1 2112 26 49.546 93.268 93.268 734 51 FRM

2

00001( P ) 1 18792 24 40.172 46.390 47.254 1023 90 FRM

2

-------------------------------------------------------------------------------

-------------------------------------- Texture Average data all frames -------

----------------------------

Total encoding time for the seq. : 4.637 sec (0.86 fps)

Total ME time for sequence : 0.191 sec

Y { PSNR (dB), cSNR (dB), MSE } : { 41.024, 40.926, 5.25332 }

U { PSNR (dB), cSNR (dB), MSE } : { 46.119, 46.104, 1.59462 }

V { PSNR (dB), cSNR (dB), MSE } : { 46.641, 46.593, 1.42487 }

Base\_Y { PSNR (dB), cSNR (dB), MSE } : { 41.902, 41.878, 4.21982 }

Base\_U { PSNR (dB), cSNR (dB), MSE } : { 45.771, 45.771, 1.72197 }

Base\_V { PSNR (dB), cSNR (dB), MSE } : { 45.993, 45.993, 1.63612 }

Enh\_Y { PSNR (dB), cSNR (dB), MSE } : { 40.147, 40.146, 6.28681 }

Enh\_U { PSNR (dB), cSNR (dB), MSE } : { 46.466, 46.466, 1.46726 }

Enh\_V { PSNR (dB), cSNR (dB), MSE } : { 47.290, 47.290, 1.21362 }

Total bits : 272352 (I 171704, P 100288, NVB 360)

Base layer Total-bits : 203192 (I 171704, P 31296, NVB 192)

Enh layer Total-bits : 69160 (I 0, P 68992, NVB 168)

Bit rate (kbit/s) @ 24.00 Hz : 3268.22

Base layer BR (kbit/s) @ 24.00 Hz: 2438.30

Enh layer BR (kbit/s) @ 24.00 Hz : 829.92

Bits to avoid Startcode Emulation : 19

Bits for parameter sets : 360

Bits for filler data : 0

-------------------------------------------------------------------------------

-------------------------------------- Depth Average data all frames ---------

--------------------------

Total encoding time for the seq. : 2.888 sec (1.39 fps)

Total ME time for sequence : 0.103 sec

Y { PSNR (dB), cSNR (dB), MSE } : { 49.325, 49.321, 0.76033 }

U { PSNR (dB), cSNR (dB), MSE } : { 93.268, 93.268, 0.00000 }

V { PSNR (dB), cSNR (dB), MSE } : { 93.268, 93.268, 0.00000 }

View0\_Y { PSNR (dB), cSNR (dB), MSE } : { 49.195, 49.192, 0.78322 }

View0\_U { PSNR (dB), cSNR (dB), MSE } : { 93.268, 93.268, 0.00000 }

View0\_V { PSNR (dB), cSNR (dB), MSE } : { 93.268, 93.268, 0.00000 }

View1\_Y { PSNR (dB), cSNR (dB), MSE } : { 49.454, 49.453, 0.73745 }

View1\_U { PSNR (dB), cSNR (dB), MSE } : { 93.268, 93.268, 0.00000 }

View1\_V { PSNR (dB), cSNR (dB), MSE } : { 93.268, 93.268, 0.00000 }

Total bits : 14312 (I 7888, P 6184, NVB 240)

View 0 Total-bits : 9320 (I 7888, P 1360, NVB 72)

View 1 Total-bits : 4992 (I 0, P 4824, NVB 168)

Bit rate (kbit/s) @ 24.00 Hz : 171.74

View 0 BR (kbit/s) @ 24.00 Hz : 111.84

View 1 BR (kbit/s) @ 24.00 Hz : 59.90

Bits to avoid Startcode Emulation : 34

Bits for parameter sets : 240

Bits for filler data : 0

-------------------------------------------------------------------------------

Total Bit Rate (kbit/s) @ 24.00 Hz : 3439.97

-------------------------------------------------------------------------------

-----------------------

Exit JM 18 (FRExt) encoder ver 18.3

-------------------------------------------------------------------------------

Note: The above PSNR number is corresponding to the input of codec. For MFC texture view, Base layer is a frame compatible format, Enhancement layer is another frame compatible format. To compute the PSNR of the full resolution left texture view and right texture view, a reconstruction step at decoder is required. The full resolution reconstruction of the left and right texture views is not supported at the encoder. For Depth view, PSNR is corresonding to the left and right depth view.

## Encoder Limitations

At this point, the encoder is characterized by certain limitations which may impact its usage. In particular, some items that have been identified as being problematic or not properly supported in the JM18.3 software for MVC setting include:

* The encoder requires both the texture views and depth views being progressive and having the same resolution.
* The encoder may not perform all level/profile checks as specified in Annex A of the standard which may result in incompatible/non-conforming bitstreams. Some of the conformance issues can be fixed by changing the encoder configuration. For example, the issue of mv limits can be avoided by disabling certain modes.
* Picture Level Rate Distortion (RD) Optimization does not currently fully support interlace coding modes and may also require memory optimizations to improve performance.

# MFC Depth Encoder Parameters

4. MFC Depth encoder Parameters

## MFC texture coding parameters (main configuration file)

The MFC coding parameters are primarily based on the MVC coding parameters that are supported in JM18.3. In this manual, only any newly introduced parameters as well as modified/extended parameters and parameters that have been restricted in terms of their usage are listed. For other parameters and their usage, please refer to the JM reference software manual **[[3]](#footnote-3)**.

### ProcessInput

*Class*: Numeric (Integer)

*Description*: Perform optional preprocessing on the input sequence. For MFC, the value is set to 14.

### ProfileIDC

*Class*: Numeric (Integer)

*Description*: Value of the **profile\_idc** syntax element. For MFC, the value is set to 134.

|  |  |
| --- | --- |
| ***Options:*** | |
| *44* | CAVLC 4:4:4 Intra |
| *66* | Baseline |
| *77* | Main |
| *88* | Extended |
| *100* | High |
| *110* | High 10 or High 10 Intra |
| *122* | High 4:2:2Predictive or High 4:2:2 Intra |
| *244* | High 4:4:4 or High 4:4:4 Intra |
| *118* | Multiview High |
| *128* | Stereo High |
| *134* | MFC High Profile |
| *138* | Multiview Depth High Profile |
| *135* | MFC Depth High Profile |

### Rpu\_Filter\_Enabled\_Flag

*Class*: Numeric (Integer)

*Description*: Specifies whether the MFC RPU filter is enabled or disabled. Default value is 1.

|  |  |
| --- | --- |
| ***Options:*** | |
| *0* | RPU Filter Disabled. The output of the RPU is set to a constant value of 128. |
| *1* | RPU Filter Enabled. |

### MFC\_Format\_Idc

*Class*: Numeric (Integer)

*Description*: Specifies the FC base layer format for MFC.

|  |  |
| --- | --- |
| ***Options:*** | |
| *0* | SbS |
| *1* | TaB |

### DefaultGridPosition

*Class*: Numeric (Integer)

*Description*: Specifies whether the default grid position shall be used or not for the FC base layer. The default grid position depends on the value of MFC\_Format\_Idc as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***MFC\_Format\_Idc*** | ***View0\_Grid\_Position\_x*** | ***View0\_Grid\_Position\_y*** | ***View1\_Grid\_Position\_x*** | ***View1\_Grid\_Position\_y*** |
| *0* | 4 | 8 | 12 | 8 |
| *1* | 8 | 4 | 8 | 12 |

### View0\_Grid\_Position\_x

*Class*: Numeric (Integer)

*Description*: Specifies the explicit x component of the ( x, y ) coordinate pair for constituent frame 0 of the FC base layer, required only when DefaultGridPosition is set to 0. The allowed values are 4, 8 and 12.

### View0\_Grid\_Position\_y

*Class*: Numeric (Integer)

*Description*: Specifies the explicit y component of the ( x, y ) coordinate pair for constituent frame 0 of the FC base layer, required only when DefaultGridPosition is set to 0. The allowed values are 4, 8 and 12.

### View1\_Grid\_Position\_x

*Class*: Numeric (Integer)

*Description*: Specifies the explicit x component of the ( x, y ) coordinate pair for constituent frame 1 of the FC base layer, required only when DefaultGridPosition is set to 0. The allowed values are 4, 8 and 12.

### View1\_Grid\_Position\_y

*Class*: Numeric (Integer)

*Description*: Specifies the explicit y component of the ( x, y ) coordinate pair for constituent frame 1 of the FC base layer, required only when DefaultGridPosition is set to 0. The allowed values are 4, 8 and 12.

### Mux3DBaseFilter

*Class*: Numeric (Integer)

*Description*: Specifies the low pass downsampling filter used for the creation of the MFC FC base layer. Default value is 0.

|  |  |
| --- | --- |
| ***Options:*** | |
| *0* | SVC3D (cutoff frequency = 0.40) |
| *1* | MFC\_MUX\_FC\_P44 (cutoff frequence = 0.44) |

### Mux3DEnhFilter

*Class*: Numeric (Integer)

*Description*: Specifies the downsampling filter used for the creation of the MFC enhancement layer. Default value is 1.

|  |  |
| --- | --- |
| ***Options:*** | |
| *0* | SVC3D (cutoff frequency = 0.40) |
| *1* | MFC\_MUX\_FC\_P44 (cutoff frequency = 0.44) |

### EnableDbgYUVFiles

*Class*: Numeric (Integer)

*Description*: Specifies if the generation of debug YUV files should be enabled. Default value is 0. The debug YUV files contain the base layer frame compatible source yuv, the enhancement layer frame compatible source yuv, the base layer frame compatible reconstructed yuv, the rpu processed base layer frame compatible reconstructed yuv. The size of the source yuv files were of the size of the input source’s horizontal and vertical ressolutions.The reconstructed yuv files were padded as to have the horizontal and vertical resolutions as multiple of 16.

### FPASEIPresentFlag

*Class*: Numeric (Integer)

*Description*: Specifies if the Frame Packing Arrangement (FPA) SEI message shall be created and be present in the FC base layer. Default value is 1 for MFC**[[4]](#footnote-4)**. In MFCD case, the value shall be set to 0.

### 3DVCoding

*Class*: Numeric (Integer)

*Description*: Specifies if both texture and depth coding shall be enabled. The value shall be set to 1.

### 3DVCodingOrder

*Class*: Numeric (Text)

*Description*: Specifies coding order of texture view and depth view. Default case is "T0D0D1T1", which means for the same POC, MFC texture BL is coded first, followed by depth view 0, depth view 1 and MFC texture EL.

## MVC depth coding parameters (main configuration file)

Same as MVC case. No additional parameters added.

## MVC coding parameters (View 1/Enhancement layer configuration file)

No additional paramteres are added.

# MFC DEPTH Hardcoded Encoder Parameters

5. MFC DEPTH Hard coded Encoder Parameters

For MFC DPETH, several additional preprocessor directives, i.e. macro definitions, have been added that can control the behavior of the software.

## defines.h

|  |  |  |
| --- | --- | --- |
| *MFC\_ENC\_3D\_FCFR* |  | Enable MFC |
| *MFC\_PROCESS\_INPUT* |  | Set to 14 for MFC |
| *MFC\_INTERLACE\_POC\_MANAGEMENT* |  | allow PocMemeoryManagement for interlace in MVC profile |
| *MFC\_JMCODE\_FIX* |  | Fix JM code issue for interlaced coding |
| *DISPLAY\_LAYERLEVEL\_BITUSAGE* |  | Enable bit usage for each layer |
| *MFC\_ENC\_EL\_CONFIG* |  | Separate RDPictureDecision from Base layer |
| *MFC\_PROFILING* |  | Code for encoder run time profiling |
| *MFC\_FPA\_SEI* |  | Enable FPA SEI message |
| *MFC\_DEPTH\_ENC* |  | Enable MFC Depth |

# Using The MFC DEPTH Decoder Module

6. Using The MFC DEPTH Decoder Module

This section provides a detailed description of the MFC Depth related parameters in the H.264/MPEG-4 AVC JM decoder.

Note that for MFC texture views, an additional reconstruction process is required after decoding MFC encoded bitstreams to reconstruct the enhanced resolution output images. To perform only decoding of both the base and enhancement layer video sequences for texture views, without reconstruction, ExportViews in decoder\_mfc.cfg should be set to 0. To perform a full reconstruction of both left and right texture views, ExportViews in decoder\_mfc.cfg should be set to 1.

## Decoder Syntax

ldecod [-s] [-h] [-d] [decoder\_mfc\_depth.cfg]   
 {[-f curenc1.cfg]...[-f curencN.cfg]}   
 [-i bitstream.264] [-o output.yuv] [-r reference.yuv]   
 {[-p DecParam1=DecValue1]...[-p DecParamM=DecValueM]}   
 [-n] Nframes [-mpr] LValue

Examples of usage:

MFC example:

Decoding only: ldecod.exe –d decoder\_mfc\_depth.cfg **-p ExportViews=0**

MFC Complete texture View Reconstruction: ldecod.exe –d decoder\_mfc\_depth.cfg **-p ExportViews=1 -p DemuxMode=1**

## Decoder Configuration File Format

### ExportViews

*Class*: Numeric (Integer)

*Description*: Specifies if the left and right texture views should be reconstructed by applying the recommended reconstruction method specified in the MFC specification. Default value is set to 0 (output base and enhancement layers as is). To be more specific, to perform only decoding of both the base and enhancement layer video sequences, without reconstruction, ExportViews in decoder\_mfc.cfg should be set to 0. To perform a full reconstruction of both left and right texture views, ExportViews in decoder\_mfc.cfg should be set to 1.

### DeMuxMode

*Class*: Numeric (Integer)

*Description*: Specifies whether the texture material are reconstructed at full resolution from only the FC base layer, using a predefined interpolation process, or using the complete MFC process. Default value is set to 1. It only takes effect when ExportViews is set to 1.

|  |  |
| --- | --- |
| ***Options:*** | |
| *0* | FC base layer |
| *1* | MFC |

### EnableDbgYUVFiles

*Class*: Numeric (Integer)

*Description*: Specifies if the generation of debug YUV files should be enabled. Default value is 0. For MFC texture views, The debug YUV files contain the base layer frame compatible reconstructed yuv, the rpu processed base layer frame compatible reconstructed yuv.The reconstructed yuv files were padded as to have horizontal and vertical resolutions as multiple of 16.

### OutputRecYUVFile

*Class*: Numeric (Integer)

*Description*: Specifies if output reconstructed YUV files should be enabled.

### OutputFileForTexture

*Class*: Text

*Description*: Specifies the output file name for texture views.

### OutputFileForDepth

*Class*: Text

*Description*: Specifies the output file name for depth views.

### WriteUVForTexture

*Class*: Numeric (Integer)

*Description*: Specifies whether write UV components for texture views should be enabled. Default value is set to 1.

### WriteUVForDepth

*Class*: Numeric (Integer)

*Description*: Specifies whether write UV components for depth views should be enabled. Default value is set to 1.

## Decoder Output

When running the decoder, the decoder will display on screen a variety of statistics for every frame that is decoded. The output information generated may look as follows:

Setting Default Parameters...

Parsing Configfile decoder\_mfc\_depth.cfg

....................

Parsing command line string 'ExportViews = 0'.

----------------------------- JM 18.3 (FRExt) -----------------------------

--------------------------------------------------------------------------

Input H.264 bitstream : test.264

Output decoded YUV : texture\_dec.yuv

--------------------------------------------------------------------------

Profile IDC : 100

Image Format : 480x272 (480x272)

Color Format : 4:2:0 (8:8:8)

--------------------------------------------------------------------------

POC must = frame# or field# for SNRs to be correct

--------------------------------------------------------------------------

Frame POC Pic# QP SnrY SnrU SnrV Y:U:V Time(ms)

--------------------------------------------------------------------------

00000(IDR) 0 0 22 0.0000 0.0000 0.0000 4:2:0 13

00000(IDR) 0 0 26 0.0000 0.0000 0.0000 4:2:0 3

00000( P ) 0 0 26 0.0000 0.0000 0.0000 4:2:0 2

00001( P ) 0 0 24 0.0000 0.0000 0.0000 4:2:0 5

00002( P ) 2 1 22 0.0000 0.0000 0.0000 4:2:0 3

00002( P ) 2 1 26 0.0000 0.0000 0.0000 4:2:0 1

00002( P ) 2 1 26 0.0000 0.0000 0.0000 4:2:0 2

00002( P ) 2 1 24 0.0000 0.0000 0.0000 4:2:0 4

Output status file : log.dec

8 frames are decoded.

Note: The decoder does not support the computation of PSNR. The external tools can be used to compute the PSNR after the decoding or the reconstruction of the left and right view.

# MFC DEPTH Hardcoded Decoder Parameters

7. MFC DEPTH Hardcoded Decoder Parameters

For MFC, several additional preprocessor directives, i.e. macro definitions, have been added that can control the behavior of the software.

## defines.h

|  |  |  |
| --- | --- | --- |
| *MFC\_DEC\_3D\_FCFR* | : | Enable MFC decoder |
| *MFC\_DEC\_3D\_FCFR\_STAT* | : | Print inter-layer stats for MFC |
| *IMGTYPE* | : | Hardcoded to 0. |
| *MFC\_JM\_CODE\_CHANGE* | : | Code changes, mostly bug fixes of JM code |
| *MFC\_PROFILING* | : | Code for decoder run time profiling |
| *MFC\_DEPTH\_DEC* |  | Enable MFC depth decoder |

1. Note: this software manual only contains additional information relating to the Multi-resolution Frame Compatible Stereo Coding (MFC) Plus Depth extension of MPEG-4 AVC and its reference software. Please refer to the JM Reference Software Manual for any details relating to the generic use of the MPEG-4 AVC JM reference software (based on version 18.3). [↑](#footnote-ref-1)
2. Note: In JM18.3, the second view configuration is placed in another configuration file, specified by the parameter View1ConfigFile. In our example, the second view configuration file for texture is encoder\_mfc\_texture\_view1.cfg and the second view configuration file for depth is encoder\_mfc\_depth\_view1.cfg. [↑](#footnote-ref-2)
3. Note: The JM Reference software manual can be found in the doc directory of the JM reference software. [↑](#footnote-ref-3)
4. Note: The FPA SEI message is required to be sent for the base layer in MFC. [↑](#footnote-ref-4)