NVIDIA Capture SDK

Sample Description Document

Version 7.1

Contents

Chapter 1

Module Index

1.1 Modules

Here is a list of all modules:

Chapter 2

Module Documentation

2.1 Multihead

This sample demonstrates how to grab the frame buffers from multiple displays using NvFBC. This is done by creating multiple NvFBC objects (one for each adapter), incrementing the NVFBC_TARGET_ADAPTER environment variable before each NvFBC_CreateFunction call, up to the maximum number of adapters in the system. This is accomplished with a helper class in NvFBCLibrary.h (in the Util folder) that initializes the NvFBC dll, sets pointers to the dll functions, and provides methods to set the TargetAdapter and create the NvFBC device.

2.2 NvFBCCudaNvEnc

This sample demonstrates how to use the NvFBCToCuda interface to copy the desktop into a CUDA buffer. From the CUDA buffer, this is then mapped directly to NVENC, where the NVENC hardware video encoder can encode the stream. We recommend that developers use the NVIDIA Video Codec to utilize the NVENC hardware video Encoder. This sample provides a useful wrapper class around the NVENC API, in Encoder.h.

2.3 NvFBCCudaSimple

This sample demonstrates how to use the NvFBCToCuda interface to copy the desktop into a CUDA buffer. It covers loading the NvFBC.dll, loading the NVFBC function pointers, creating an instance of NvFBCCuda, and using NvFBCCuda to copy the frame buffer into a CUDA device pointer.

2.4 NvFBCCursorCapture

This sample demonstrates how to use NVFBCs cursor capture feature to copy the desktop and cursor to system memory, where the cursor and desktop are grabbed using separate threads.

2.5 NvFBCDX9ClassificationMap

This sample demonstrates how to use the NvFBCToDX9 to capture Classification maps, and also to capture the desktop to DX9 video memory surfaces. This sample also generates a downscaled copy of the frame buffer to a bitmap, and saves it to file.

Reader.h: Declares the Reader class, this class is simply a wrapper around the NVENC reader API.

Reader.cpp: Defines the Reader class, wraps up the details when using the NVENC video reader so it don't detract from the NvFBCDX9 example.

NvFBCDX9ClassificationMap.cpp: Demonstrates usage of the NvFBCToDX9 interface with the Image Classification capture API. It demonstrates how to load NvFBC.dll, initialize NvFBC function pointers, create an instance of NvFBCDX9 and using NvFBCDX9 to extract a ClassificationMap from the driver, and also capture a downscaled version of the desktop.

2.6 NvFBCDX9DiffMap

This sample demonstrates how to use the NvFBCToDX9 to capture diff maps, and also to capture the desktop to DX9 video memory surfaces. This sample also generates a downscaled copy of the frame buffer to a bitmap, and saves it to file.

Reader.h: Declares the Reader class, this class is simply a wrapper around the NVENC reader API.

Reader.cpp: Defines the Reader class, wraps up the details when using the NVENC video reader so it don't detract from the NvFBCDX9 example.

NvFBCDX9DiffMap.cpp: Demonstrates usage of the NvFBCToDX9 interface with the DiffMap capture API. It demonstrates how to load NvFBC.dll, initialize NvFBC function pointers, create an instance of NvFBCDX9 and using NvFBCDX9 to extract a diffmap from the driver, and also capture a downscaled version of the desktop.

2.7 NvFBCDX9NvEnc

This sample demonstrates how to use the NvFBCToDX9 to capture to a DirectX 9 surface, and then send it to the NVENC encoder. This sample also demonstrates how to enable NVFBC Image Area Classification feature and pair it seamlessly with NVENC Emphasis Level Map feature for tweaking visual quality of Capture+Encoded video stream This sample provides a useful wrapper class around the NVENC API, in Encoder.h.

From the DX9 Buffer, we use VideoProcessBlt to transferred to NVENC where the H.264 video encoder can encode the stream.

Encoder.h: Declares the Encoder class, this class is simply a wrapper around the NVENC encoder API.

Encoder.cpp: Defines the Encoder class, wraps up the details when using the NVENC video encoder so it don't detract from the NvFBCDX9 example.

NvFBCDX9NvEnc.cpp: Demonstrates usage of the NvFBCToDX9 interface. It demonstrates how to load NVFBC.dll, initialize NvFBC function pointers, create an instance of NVFBCToDX9Vid interface and using NVFBCToDX9Vid APIs to copy the frame buffer into a DX9 device pointer, which is then passed to NVENC encoder for video encoding.

2.8 NvFBCDx9NvEncSharedSurface

The sample demonstrates how to use NVFBC to grab the desktop to a DX9 shared surface, and then send it to the hardware encoder to encode as H.264 or HEVC using a separate DX9 context.

Encoder.h: Declares the Encoder class, this class is simply a wrapper around the NVENC encoder API.

Encoder.cpp: Defines the Encoder class, wraps up the details when using the NVENC video encoder so it don't distract from the NvFBCDX9 example.

NvFBCDX9NvEnc.cpp: Demonstrates usage of the NvFBCToDX9 interface. It demonstrates how to load vFBC.dll, initialize NvFBC function pointers, create an instance of NvFBCDX9 and using NvFBCDX9 to copy the frame buffer into a DX9 device pointer, where it was VideoProcessBlt, where is then passed to NVENC to generate a H.264 video.

2.9 NvFBCEnableAPI

This sample demonstrates how to use NvFBC_Enable API exported by NVFBC library to programmatically enable or disable NVFBC feature. The API needs to be called from an application that is running with administrator privileges.

2.10 NvFBCToSys

Demonstrates the use of NvFBCToSys to copy the desktop to a system memory buffer and save it as a file. This sample demonstrates the use of NvFBCToSys class to record the entire desktop. It covers loading the NvFBC DLL, loading the NvFBC function pointers, creating an instance of NvFBCToSys and using it to copy the full screen frame buffer into system memory.

2.11 NvFBCToSysClassificationMap

Demonstrates the use of NvFBCToSys to copy the desktop to a system memory buffer and save it as a file. This sample demonstrates the use of NvFBCToSys class to record the entire desktop. It covers loading the NvFBC DLL, loading the NvFBC function pointers, creating an instance of NvFBCToSys and using it to copy the full screen frame buffer into system memory. This sample also demonstrates enabling NVFBC Image Area Classification Map feature.

Notice

HDMI

OpenCL

Trademarks

Copyright