SAMPLE

Video Surveillance

1 Introduction

The purpose of this sample is to demonstrate how to build and execute a video surveillance application. The application takes different H.264 elementary streams and composes the different decoded and resized outputs using OpenCL to create a 1920x1080 video frame. The frame is displayed and in parallel encoded using the AMD AMF H.264 video encoder. The application is implemented using native APIs in the AMD Media Framework (AMF).

2 Using the sample

2.1 Location

`$<installDirectory>\samples\amf\videoSurveillance\`

2.2 Contents

Package Contents

Folder:
`$<installDirectory>\samples\amf\videoSurveillance\src\`

File name	Description
`VideoSurveillance.cpp`	Source file for Video Surveillance application

Folder:
`$<installDirectory>\samples\amf\videoSurveillance\build\windows\`

File name	Description
`VideoSurveillanceVs10.sln`	Microsoft Visual Studio 10 solution file
`VideoSurveillanceVs10.vcxproj`	Microsoft Visual Studio 10 project file
`VideoSurveillanceVs10.vcxproj.filters`	Microsoft Visual Studio 10 project filter file
`VideoSurveillanceVs12.sln`	Microsoft Visual Studio 12 project solution file
`VideoSurveillanceVs12.vcxproj`	Microsoft Visual Studio 12 project file
`VideoSurveillanceVs12.vcxproj.filters`	Microsoft Visual Studio 12 project filter file

Folder:
`$<installDirectory>\samples\amf\videoSurveillance\config\`

File name	Description
`exampleConfig.cfg`	Sample configuration file

Folder:
`$<installDirectory>\samples\amf\videoSurveillance\docs\`

File name	Description
`MediaSDK_AMF_videoSurveillance.pdf`	Sample documentation

2.3 Compile

1. Ensure that the following tools and SDKs are present:
   - Microsoft Visual Studio 2010 or 2012
     If Windows Software Development Kit (SDK) is not installed, install it from http://msdn.microsoft.com/en-us/library/windows/desktop/hh852363.aspx.
2. Open one of the following solution files:
   - `$<installDirectory>\samples\amf\videoSurveillance\build\windows\VideoSurveillanceVs12.sln`
   - `$<installDirectory>\samples\amf\videoSurveillance\build\windows\VideoSurveillanceVs10.sln`
3. Build the sample:
   - Open the `VideoSurveillanceVs10.sln` solution file with Microsoft Visual Studio 2010 Professional Edition or the `VideoSurveillanceVs12.sln` solution file with Microsoft Visual Studio 2012 Professional Edition.
   - To build all the solutions, select `Build > Build Solution`.
   - The executable `videoSurveillance.exe` is created in the following folders for 32-bit builds and 64-bit builds:
     `$<installDirectory>\samples\amf\videoSurveillance\bin\x86\`
     `$<installDirectory>\samples\amf\videoSurveillance\bin\x86_64\`
   - Depending on the build (i.e. 32-bit or 64-bit), the custom build step copies the appropriate `.dlls` file from the `$<installDirectory>\dll\amf\` folder into the relevant `\bin\` directory.

3 How to Run

The sample can be executed on an AMD platform that includes the UVD and VCE hardware blocks.

On the command prompt, change to the directory that contains the executable, and execute the following command:

```
videoSurveillance.exe <ConfigFile> <MemoryType>
```

`ConfigFile`: Specify the name of the configuration file
`MemoryType`: Specify the buffer memory type, either `DX9` or `DX11`

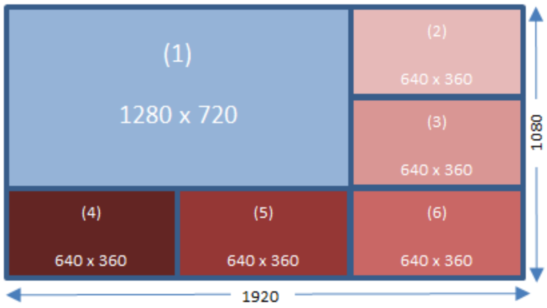
The configuration file contains the names of the input H.264 files. In the implemented sample, the following limits are set:

- Number of instances for decoding and resizing: 6
- If the configuration file contains fewer than 6 H.264 input files, then the first H.264 file will be used as input for the remaining instances.

The following example shows a sample configuration file:

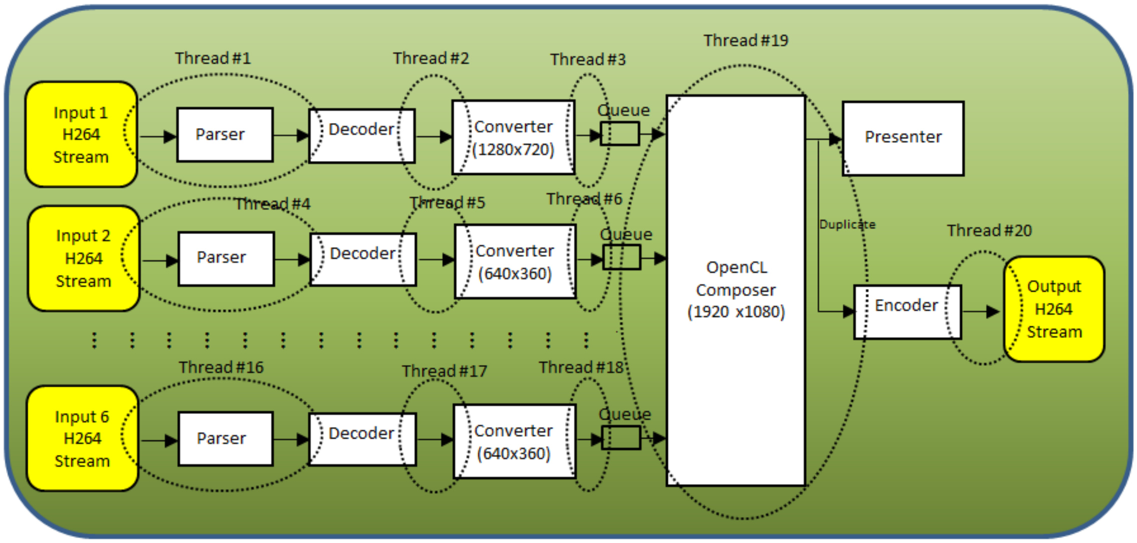
- `input_01.h264 0`: 1st H.264 decoded, resized (1280x720) and fed to the OpenCL composer
- `input_02.h264 1`: 2nd H.264 decoded, resized (640x360) and fed to the OpenCL composer
- ...
- `input_06.h264 1`: 6th H.264 decoded, resized (640x360) and fed to the OpenCL composer

The OpenCL composer mixes the resized decoded outputs specified above and creates 1920x1080 frames, as shown below.



4 Implementation Details

The sample implements the following topology for video surveillance application:



The data in the batch transcode pipeline flows through the following processing elements:

- Parser: H.264 Elementary stream data is first read by the parser which finds the SPS & PPS, finds NALU unit, and populates data structures which are fed to the H.264 Decoder.
- Decoder: HW Accelerated (UVD) H.264 Video Decoder. Decodes the input content to generate NV12 frames.
- Converter: 1st instance decoded output is resized to 1280x720 and remaining all instances decoded outputs are resized to 640x360 resolution before giving it to the OpenCL composer.
- OpenCL Composer: It takes all the resized outputs from the queues and composes to 1920x1080 frames, which are fed to Presenter, and same data is duplicated, fed to encoder.
- Encoder: HW Accelerated (VCE) H.264 Video Encoding. Encodes the duplicated 1920x1080 content from the OpenCL composer to generate compressed H.264 Elementary stream.

The sample prints the following performance parameters per session:

- Latency in ms
- Average decode + blend (OpenCL) + encode time in ms / frame
- Average time in ms to write one transcoded frame into file

5 Supported formats

The following file formats are supported:

- Video Decoders supported: H.264
- Video Encoders supported: H.264

- Output file format: H.264 Compressed Elementary Stream

6 Known Limitations

The sample is currently supported on the following platforms:

- Windows 7 (DirectX 9)
- Windows 8.1 (DirectX 9 and DirectX 11.1)

Contact

Advanced Micro Devices, Inc.
One AMD Place
P.O. Box 3453
Sunnyvale, CA, 94088-3453
Phone: +1.408.749.4000

For AMD Accelerated Parallel Processing:

URL: **developer.amd.com/appsdk**
Developing: **developer.amd.com/**
Forum: **developer.amd.com/openclforum**

The contents of this document are provided in connection with Advanced Micro Devices, Inc. ("AMD") products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. The information contained herein may be of a preliminary or advance nature and is subject to change without notice. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property rights is granted by this publication. Except as set forth in AMD's Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD's products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD's product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

Copyright and Trademarks