Introduction
We have done testing for our SDK to demonstrate current performance results on NVIDIA server, desktop and mobile GPUs for frequently used algorithms, image resolutions 2K/4K and bit depths.

Hardware and Software requirements (OS, PC, GPU, etc.)
• OS: Windows-7/8/10 (64-bit), Linux (64-bit), Linux4Tegra (Ubuntu 18.04)
• NVIDIA GPU: Server (Tesla, Quadro), Desktop (GeForce GTX/GT), Laptop (GeForce GT)
• NVIDIA Jetson Nano, TX2, AGX Xavier
• CUDA-10.0 (64-bit)

Benchmarks for Fastvideo Image & Video Processing SDK (ms)

<table>
<thead>
<tr>
<th></th>
<th>GeForce 1080</th>
<th>Quadro P6000</th>
<th>Jetson Nano / TX2 / Xavier</th>
</tr>
</thead>
<tbody>
<tr>
<td>JPEG Encoder</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2K gray (8-bit, q=90%)</td>
<td>0.216</td>
<td>0.11</td>
<td>2.9 / 1.2 / 0.42</td>
</tr>
<tr>
<td>2K (24-bit, q=90%, 4:2:0)</td>
<td>0.36</td>
<td>0.17</td>
<td>4.3 / 1.7 / 0.62</td>
</tr>
<tr>
<td>2K (24-bit, q=90%, 4:4:4)</td>
<td>0.40</td>
<td>0.21</td>
<td>6.8 / 2.6 / 0.75</td>
</tr>
<tr>
<td>4K gray (8-bit, q=90%)</td>
<td>0.55</td>
<td>0.35</td>
<td>11.3 / 4.2 / 1.23</td>
</tr>
<tr>
<td>4K (24-bit, q=90%, 4:2:0)</td>
<td>0.78</td>
<td>0.51</td>
<td>17.1 / 6.3 / 1.83</td>
</tr>
<tr>
<td>4K (24-bit, q=90%, 4:4:4)</td>
<td>1.12</td>
<td>0.74</td>
<td>27.3 / 9.9 / 2.67</td>
</tr>
<tr>
<td>4K gray (12-bit, q=90%)</td>
<td>0.83</td>
<td>0.54</td>
<td>19.3 / 7.2 / 2.07</td>
</tr>
<tr>
<td>4K (36-bit, q=90%, 4:2:0)</td>
<td>1.22</td>
<td>0.82</td>
<td>26.7 / 11.0 / 3.1</td>
</tr>
<tr>
<td>4K (36-bit, q=90%, 4:4:4)</td>
<td>1.90</td>
<td>1.32</td>
<td>50.5 / 18.4 / 5.0</td>
</tr>
</tbody>
</table>

| JPEG Decoder (restart interval 1) |              |              |                           |
| 2K gray (8-bit, q = 90%)          | 0.55         | 0.40         | 5.4 / 2.9 / 0.86          |
| 2K (24-bit, q = 90%, 4:2:0)      | 1.36         | 1.20         | 9.1 / 4.9 / 1.35          |
| 2K (24-bit, q = 90%, 4:4:4)      | 1.02         | 0.73         | 11.1 / 4.9 / 1.37         |
| 4K gray (8-bit, q = 90%)         | 1.52         | 1.28         | 20.9 / 8.7 / 2.13         |
| 4K (24-bit, q = 90%, 4:2:0)      | 2.61         | 2.12         | 39.8 / 15.8 / 4.0         |
| 4K (24-bit, q = 90%, 4:4:4)      | 2.78         | 2.28         | 45.3 / 17.3 / 4.15        |
Time and performance measurements for Fastvideo SDK modules on NVIDIA GPUs for grayscale and color images don't take into account host I/O latency (image loading to RAM from HDD/SSD and saving back). We have presented timings for computations on GPU only. As soon as any image processing pipeline consists of series of such algorithms (SDK modules), it's a reasonable approach to measure only computation time (kernel time), assuming that initial and final images reside in GPU memory.
Fastvideo SDK modules for CUDA image processing (ver. 0.14.2.0)

- Image acquisition (from HDD/SSD/RAM, camera, grabber, byte array in CPU/GPU memory, OpenGL texture, PBO, etc.)
- Unpacking module for specific formats of RAW data
- Dark Frame Subtraction
- Shading Correction
- White balance (R, G1, G2, B)
- Exposure correction with LUT (Raw Curves and Levels for each channel)
- Raw Denoiser
- Debayer: HQLI (8/16-bit), DFPD (8/16-bit), MG (16-bit)
- Denoiser for processed data
- Color Correction with matrix profile
- Color Transforms: RGB-YCbCr-RGB, RGB-HSV-RGB, etc.
- Composite or per-color 8/10/12/14/16-bit 1D LUTs for tone mapping in RGB or HSV
- Rec.601 and Rec.709 (FR and LR) support
- Gamma transform (composite or per-color component)
- 3D LUTs for RGB or HSV
- DCP profile support
- Histograms for RAW and processed data
- RGB Parade (waveform monitor)
- Rotation (90/180/270 degrees), flip/flop, arbitrary angle
- Resize (downsampling and upsampling)
- Remap (shift, resize, rotate, perspective, undistortion, projections, etc.)
- LCP support
- Unsharp Mask
- Realtime output via OpenGL
- JPEG Codec (Baseline JPEG only: 8/24-bit, Huffman encoding/decoding)
- MJPEG Codec
- 12-bit JPEG Encoder
- Raw Bayer Codec (lossy compression and decompression)
- JPEG2000 Codec (8-16 bits per channel, tile support)
- Stream-per-thread option to overlap copy and computations
- Other: trace, multiplexer, time measurements, HD-SDI support, multiple camera

Demo software for Image & Video Processing on CUDA

One can download demo applications for JPEG Codec, JPEG2000 Codec, Debayer and Resizer from https://www.fastcompression.com/download/download.htm. These are high performance applications with command-line interface to evaluate both performance and quality of GPU-based Image & Video Processing SDK. Other sample applications could be found in the SDK.

The latest version of Fast CinemaDNG Processor is available from https://www.fastcinemadng.com

The latest release of Fastvideo SDK is available upon request.

Fastvideo Roadmap 2019

- Bad Pixel Correction Tool
- New GPU Memory Manager
- New version of CUDA JPEG Codec with better performance and higher SSIM
- CUDA JPEG2000 Decoder optimization, MXF Player on GPU
- CUDA software integration with FFmpeg
- Server version of Fast CinemaDNG Processor software