Real-Time Color Space Conversion for High Resolution Video

Klaus Gaedke, Jörn Jachalsky
Technicolor Research & Innovation, Germany
klaus.gaedke@technicolor.com

Color Space Conversion

Color space conversion or color correction is a widely used technique to adapt the color characteristics of video material to the display technology employed (e.g. CRT, LCD, projection) or to create a certain artistic look. As color correction often is an interactive task and colorists need a direct response, state-of-the-art real-time color correction systems for video are so far based on expensive dedicated hardware.

Look-Up-Tables for Color Space Conversion

The use of Look-Up-Tables (LUTs) is a fairly simple, but very powerful approach for the implementation of color space conversion. Depending on the requirements for the conversion, either 1D- or 3D-LUTs have to be deployed. Using 1D-LUTs, each component of an input pixel addresses one table. This results in a simple implementation with low memory requirements.

In order to reduce the memory requirements, a 3D-Reduced-Resolution-Look-Up-Table (3D-RRLUT) is combined with a unit interpolating the output values [1].

This approach reduces the memory requirements significantly. Even for 3D-RRLUTs with 12 bit resolution and 129 entries for each color component, only 9.7 MB of memory are required.

Tetrahedral Interpolation

Tetrahedral Interpolation [1] is an efficient algorithm to interpolate the output values of a 3D-RRLUT.

This cuboid is decomposed into 6 tetrahedrons T1 to T6. Depending on the position \( f_1, f_2, f_3 \) of the input pixel within the cuboid, the corresponding tetrahedron is chosen and the related interpolation is calculated.

Performance

The computation of the Tetrahedral Interpolation requires 250 arithmetic and logical operations for each pixel. The real-time processing requirements for common video formats range from 2.6 GOPS to 52.4 GOPS.

The table below shows the performance of the Tetrahedral Interpolation on the 3 different parallel computing devices. Processing and transfer times are given per frame.

Implementation

The Tetrahedral Interpolation was implemented on 3 different parallel computing devices (IBM Cell B.E., NVIDIA Geoforce GTS250, and NVIDIA Tesla C2050).

Parallelization was done on pixel level. To avoid the significant performance loss caused by branches, always all possible interpolations are calculated. The final result is computed with 3 fast select instructions.

OpenCL 1.0 was used as programming language for the GPUs. The Cell B.E. was programmed in C incl. Intrinsics.

Conclusion

- Tetrahedral Interpolation performs very well on GPUs
  - High performance gain compared to CPUs
  - High utilization of computational resources on GPU
- Single low-cost GTS250 keeps pace with Cell B.E. and is sufficient for real-time color correction of 1080p60
- Single high-end Tesla C2050 supports real-time color correction up to the 2048p25 format

GPUs are ready to replace expensive dedicated hardware for real-time color space conversion