Ultra-High Definition Videos and Their Applications over the Network

assoc. prof. Petr Holub, Ph.D.
CESNET & Masaryk University
hopet@ics.muni.cz

The 7th International Symposium on VICTORIES Project,
2014–10–08
Overview

What is UltraHD and why we need it

Applications showcase: UltraGrid & SAGE & CoUniverse

Future of networked media applications
What Does UltraHD Mean?

- Video beyond High-Definition (HD)
  - there is some historical confusion: 4K vs. 8K video
  - 2160p aka SuperHD/SHD: $3840 \times 2160$ (8 Mpix)
  - 4K in cinema: $4096 \times 2048$, $4096 \times 2160$
  - 8K/4320p: $7680 \times 4320$ (33 Mpix)
  - scalable display systems: 55–100 Mpix or higher
Why Do We Need UHD?

- Limitation: angular resolution of human eye, 1 arcminute for 20/20 (normal) sight
  - optimal viewing angle
    - HD video: 30°
    - 4K video: 55°
    - 8K video: 100°
  - if we had 65" TV, we would need to get as close as
    - HD video: 114" (2.9 m)
    - 4K video: 57" (1.4 m)
    - 8K video: 29" (.7 m)
Why Do We Need UHD?

Human eye has uneven resolution

- low spatial
  - high temporal
- high spatial
  - low temporal
- low spatial
  - high temporal

→ if a viewer is allowed to move his head, we need to increase both spatial and temporal resolution
Why Do We Need UHD?

- Scaling temporal resolution:
  - cinematography: 24 fps, recently 48 fps
  - broadcasting: 25/30/50/60 fps
  - computer systems: 60 fps
  - 8K video: 120 fps

- Higher temporal resolution: 300–10,000 fps
  - beyond the human perception in real-time
  - analysis of various processes: industry, sports, military, …
Why Do We Need UHD?

- Improving color detail
  - 8 b or 10 b per color component in broadcasting
  - up to 16 b for more demanding applications: e.g., pathology
Why Do We Need UHD?

- Invasive cardiology – simultaneous real-time analysis of multiple modalities (X-ray, FFR, OCT, etc.)
Why Do We Need UHD?

- Scientific visualizations – large data analysis
  - geosurvery, pathology: >1 Gpix imagery
  - collaborative data/image sharing
  - remote control of instruments
Why Do We Need UHD?

- Arts & education
  - distributed performances: music, theater
### What Does That Mean for Network?

Uncompressed video bitrates [Gbps]:

<table>
<thead>
<tr>
<th>Resolution</th>
<th>30 fps, 8 b</th>
<th>60 fps, 10 b</th>
<th>120 fps, 16 b</th>
</tr>
</thead>
<tbody>
<tr>
<td>HD – 1080p (1920×1080)</td>
<td>1.5</td>
<td>3.7</td>
<td>12</td>
</tr>
<tr>
<td>4K – 2160p (3840×2160)</td>
<td>6</td>
<td>15</td>
<td>48</td>
</tr>
<tr>
<td>8K – 4320p (7680×4320)</td>
<td>24</td>
<td>60</td>
<td>191</td>
</tr>
</tbody>
</table>
Do We Need Uncompressed Data?

— In most cases – NO
  — because of limits of human eye
  — for archival applications, lossless compression is an option: but provides only limited data reduction ($\approx \ast \frac{2}{3}$)

— Experiments with human sight
  — HD video can be brought from 1.5 Gbps to $\approx 80$ Mbps M-JPEG without user being able to tell the difference in terms of image quality
  — experimentally confirmed in cardiology and cinematography for real-time applications (not archival) using ABX tests$^1$

---

What Does Interactive Mean?

- Specifics of interactive (= real-time) applications: human perception of latency
  - ITU-T G.115: *150 ms* one way latency for phone (audio communication)
  - some applications can tolerate about *200 ms* one-way delay (experiments with remote control of medical robots)
  - some application are much more sensitive
    - music orchestras: *10–40 ms* (chamber–symphonic)
- Interactivity limits amount of processing
  - very limited buffering needed
  - compression often limited to intra-frame or progressive inter-frame schemes
UltraHD Video Wrap-Up

- We need to consider limitations of human perception when optimizing video applications.
- 4K/8K UHD spans wide range of bitrates
  - uncompressed: 6 Gbps – >100 Gbps
  - compressed: starting from 60 Mbps for interactive applications
  - streaming applications can go substantially lower
- End-to-end one-way delay below 150 ms is acceptable for most of the interactive applications
  - specific applications may require 10–40 ms range

How can we transport it over the network, esp. for interactive applications?
Overview

What is UltraHD and why we need it

Applications showcase: UltraGrid & SAGE & CoUniverse

Future of networked media applications
UltraHD on Commodity HW

- Dedicated hardware solutions are paving the path toward the future...
  - ... but to make the technology widely available, it is neccessary to make it work also on commodity systems
  - dedicated hardware will remain an option only for the most wide-spread technologies for the commodity systems

Mission of our team at CESNET & Masaryk Univesity:

*Explore the limits of commodity hardware for high-resolution image processing and network transmissions.*
Applications Showcase: UltraGrid & SAGE & CoUniverse

- UltraGrid: open-source multi-platform application for low-latency network transmissions of HD and post-HD (4K/8K) video
  - developed by CESNET with contributors from around the world
  - http://www.ultragrid.cz/

- SAGE: scalable distributed display system
  - developed by EVL UIC
  - http://www.sagecommons.org/

- CoUniverse: self-organization for high-bandwidth real-time applications
  - developed by Masaryk Univesity & CESNET
  - http://couniverse.sitola.cz/
UltraGrid Platform

- Technology
  - As high quality and as low latency as possible on commodity hardware
    - commodity video capture cards,
    - commodity GPU cards,
    - 10GE (or better) is a plus but not necessary,
    - Linux, Mac, Windows.
  - A platform for implementing research results, namely
    - compression & image processing,
    - forward error correction,
    - congestion control.
  - End-to-end latency in a local network: 80–150 ms, depending on HW used.
UltraGrid Platform

Interesting milestones

2005: Uncompressed 1080i, multi-point.
2007: Low-latency CPU compression-schemes
    Self-organization
    Optical multicast
2008: 2K/4K
2011: GPU compressions
2012: 8K – Trans-Atlantic multi-point
    ACM Multimedia Award
2013: Comprimato Systems spin-off (GPU JPEG2000)
UltraGrid Platform

- Supported video formats
  - HD, 2K
  - 4K, 8K – tiled or native (single tile)
  - multichannel video (e.g., stereoscopic/3D, tiled)
- Uncompressed vs. compressed video
  - Low-latency compression schemes:
    - GLSL-accelerated DXT1, DXT5-YCoCg
    - CUDA-accelerated JPEG, DXT5-YCoCg
    - CPU-based low-latency H.264 – via external X264 library
    - GPU-accelerated JPEG2000 – available separately via Comprimato Systems company
  - Parallelization is the key! Not only in the networking technologies...
GPU-Accelerated Compression

- Examples of compressed video bitrates for 4Kp30 over IP:
  - H.264-compressed: 60–200 Mbps
  - JPEG-compressed: 150–400 Mbps
  - DXT-compressed: 1 Gbps
  - uncompressed (RGB 8 b): 6 Gbps

SAGE display with various compressions
GPU-Accelerated Compression

- Fine-grained parallelization of JPEG
  - per-row/column DCT/IDCT
  - per pixel RLE and Huffman coding
  - parallel stream compacting
  - parallel decompression using restart intervals

- Performance numbers (including transfer to/from GPU, NVidia 580GTX)$^2$
  - DXT5 GLSL: 349 Mpix/s
  - JPEG CUDA: up to 1.580 Mpix/s (= 38 Gbps)
  
  ... up to 47 fps of 8K UHD on a single GPU (244 W TDP)
  ... and you can parallelize across multiple GPUs
  ... c.f. CPU: 83–167 Mpix/s, FPGAs: 405–750 Mpix/s

- DXT5 CUDA: $\geq$ 1.580 Mpix/s

---

GPU-Accelerated Compression

— Performance of JPEG stages for 2160p video

Figure 5: Distribution of computation time between JPEG phases in dependence on quality and mode settings. Measurements are taken as an average of painting, text, chart, big building in 2160p resolution.

(a) Time performance in milliseconds. There were 30 samples for each measurement and the confidence interval is 95%.

(b) Maximum achieved throughput in Mpix/s.

Table 1: DXT compression performance. 4320p results are not available for ATI cards because of texture size limitation (4096×4096) of the drivers.
Forward Error Correction

- LDGM
  - CPU (vectorized using SSE) can be used up to \( \approx 600 \text{ Mbps} \) flows because of CPU↔GPU transmissions overhead
  - CPU performance is insufficient to go beyond 1 Gbps, even when vector parallelism is applied
  - massively parallel GPU implementation is required for 1 Gbps and above

\[ \implies \text{packet loss up to 10% can be mitigated with reasonable overhead} \]
SAGE

- Developed by Electronic Visualization Lab @ UIC
- Rendering platform & network middleware allowing interconnection of theoretically unlimited number of computers into a single rendering cluster
- Fully parallel architecture on tiled display
  - allows parallel rendering of visualization applications, arbitrary translation and overlap of windows, a few other transforms (e.g., scaling, rotation)
  - supports 100 Mpix per display wall or even more
- Around 100 installations around the world
SAGE: How Does It Work?

- SAGE workspace is controlled by a Free Space Manager (FSManager)
- FSManager knows window coordinates for all applications, thus knowing on which screens the window gets rendered
- FSManager informs producers of graphics data, how the image should be split and where it should be sent to
SAGE: How Does It Work?

Tiled Display

SAGE Receiver

SAGE Receiver

SAGE Receiver

SAGE Receiver

FreeSpace Manager

UI client

UI client

SAIL

SAIL

SAIL

App1

App2

App3

Pixel Stream

SAGE Messages

SAIL: Sage Application Interface Library
SAGE and UltraGrid

- UltraGrid can render through libSAIL
  - single node and two node modes (bitrates for 4K)

- audio uses SAGE
- measured end-to-end latency: 270 ms
SAGE and UltraGrid
CoUniverse

— Motivation

— multipoint collaborative environments comprise a large number of components: producers, receivers, distributors (application-level multicast – ALM)
  ⇒ manual orchestration is cumbersome
  — need to react dynamically to changing network conditions

— bitrates comparable to capacities of network links
  — 1080p30 HD video over IP:
    H.264: 20–60 Mbps, M-JPEG: 60–150 Mbps,
    uncompressed: 1.5 Gbps,
  — 4K is 2–4× more compared to HD,
  — 8K is 2–4× more compared to 4K.

⇒

Self-organization is needed.
- Optimization of ALM = \( \mathcal{NP} \)-complete problem.
- Shortest-path/greedy routing may not even provide a solution for bitrates comparable to the capacity of network links.
- Application-level multicast allows for per-client data transformations.
- We need to optimize for:
  1. minimization of latency (alternatively equalization)
  2. maximization of subjective quality (user perception)
- We would like to integrate with the advanced networks services where available (e.g., on-demand circuits/NSI, SDN)
CoUniverse

- State of the CoUniverse
  - prototype implementation at https://couniverse.sitola.cz/
  - builds a self-organizing P2P network using JXTA
  - implements orchestration of UltraGrid
  - solves the $\mathcal{NP}$-complete flow scheduling problem using constraint programming or ant-colony optimization techniques (switchable)
  - supports integration with NSLv2 (collaboration with AIST)
Overview

What is UltraHD and why we need it

Applications showcase: UltraGrid & SAGE & CoUniverse

Future of networked media applications
Future of Networked Media Applications

- Resolution may grow for specific applications
  - 8Kp120 will be probably sufficient for generic 2D
  - large-scale visualizations and collaborative environments may exceed this
- Complex real-time processing, e.g.,
  - data (re)compression,
  - reconstruction of 3D models from 2D data,
  - anonymization of data for medical applications.
- Capture & transmission of 3D scenes (holography)
- Interaction with the media
  - e.g., touch-based vs. touch-less interaction, haptic feedback
Future of Networked Media Applications

- Better integration of real-time applications with the networks
  - custom routing and multicasting schemes based on SDN (or network programmability in general),
  - complex data processing on network elements – failed dream of active networks?
- Improvement of delivery schemes for streaming applications (out of scope of this talk)
  - caching strategies, routing optimization, …
  - scalability is needed for massive delivery.
Future of Networked Media Applications

- Efficient adaptation to changing network conditions
  - adaptive (e.g., layered) compression schemes,
  - ongoing experiments with congestion control interaction for real-time applications.

- Adaptation of network for applications needs
  - temporary allocation of network resources (BoD services, etc.),
  - use of programmability for optimization of network structure.
Selected Relevant Papers


Thank you for your attention!

Q&A