Lossless coding for Still Pictures with HEVC

Guillaume Barroux
Fujitsu Laboratories Ltd.
Guillaume.b@jp.fujitsu.com
Objectives of the presentation

- Show the positive points which make this scheme interesting
  - Use case oriented

- Explain the lossless coding which can be done with HEVC
  - Technology oriented
Use Case Part

WHY USING LAYERED LOSSLESS?
Targeted applications

- **Storing format for data on PACS**
  - This solution is to replace heavy and inconvenient pure raw files

- **Lossless monochrome still pictures**
  - CT images etc.

- **Lossless 4:4:4 videos as series of still pictures**
  - 2D rendering of 3D captures are saved in lossless
  - Also targeting endoscopic videos
  - Needs observed by Fujitsu in medical facilities
HEVC for still pictures?

- HEVC is very efficient for still picture coding too

- By industry demand, HEVC even has a dedicated profile
  - https://en.wikipedia.org/wiki/High_Efficiency_Video_Coding#Main_Spill_Picture
Desirability of layered lossless

- Each picture is separated into two parts:
  - A light lossy part
    - Which can still be visually lossless
  - A heavy lossless part
    - Encoding remaining noise for full lossless reconstruction

- The two parts are complementary
- The lossy part can be processed without the other

![Diagram showing complete file with ~70% necessary data for true lossless and ~30% visually lossless base]
Use case 1

Across medical facilities

Physician can

1) Request files from another facility
2) Receive the visually lossless files quickly
3a) Start working on the visually lossless files
3b) Meanwhile, the full lossless versions are downloaded
4) The physician’s facility gets full lossless data for legal purposes
Use case 2

- Within the medical facility

- Physician can
  - 1) Request for lossy files from PACS
  - 2) Receive the visually lossless files quickly on his tablet
  - 3) Work on the visually lossless files
  - The whole transfer requires no big complexity from PACS
    - No encoding/decoding – only re-encapsulating
  - The whole transfer is light for the internal network

![Complete file diagram]

- Approx 70% necessary data for true lossless
- Approx 30% visually lossless base

Copyright 2016 FUJITSU LABORATORIES LTD.
Use case 3

- Within the medical facility

- Facility can delete all lossless complementary data
  - No reading of the stream necessary
  - Simply re-encapsulate the lossy fragments into a new DICOM item
    - Without the lossless complementary parts
  - Reduce memory burden on systems without complexity

Diagram:

- Original PACS data
  - Lossy 0
  - Lossy 1
  - Lossy 2
  - Lossless 0
  - Lossless 1
  - Lossless 2

- PACS data without lossless
  - Lossy 0
  - Lossy 1
  - Lossy 2

- Fragment

- HEVC Data
Availability on market

**Decoders:**
- Software decoding is fast enough for most use cases
- Proposed simplifications allow for cheaper dedicated hardware
- Decoding Lossless complement requires very low complexity:
  - Copy the lossy image
  - Decode extra noise
  - Using DPCM: Difference between values sent instead of raw values

**For encoders**
- Any 4:0:0 / 4:4:4 scalable encoder with the specified constraints
  - It only simplifies their processing
  - No required additional functionality
- Using specific codecs is advantageous but not required
  - Lowers the complexity of implementing and operating

Copyright 2016 FUJITSU LABORATORIES LTD.
Advantages over JPEG2000 (1/2)

Lower complexity when processing images:

<table>
<thead>
<tr>
<th>When sending 1 lossy part:</th>
<th>When sending 1 lossless image:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>JPEG streams</strong></td>
<td><strong>HEVC streams</strong></td>
</tr>
<tr>
<td>1) Find Image in DICOM object</td>
<td>1) Find lossy image in DICOM object</td>
</tr>
<tr>
<td>2) Open JPEG stream</td>
<td>2) Re-encapsulate lossy fragment selected</td>
</tr>
<tr>
<td>3) Find necessary where to truncate stream to get required lossy quality</td>
<td>3) Send new object</td>
</tr>
<tr>
<td>4) Re-encapsulate lossy trunk selected</td>
<td>5) Send new object</td>
</tr>
<tr>
<td>5) Send new object</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>When sending 1 lossless image:</th>
<th>When sending 1 lossless image:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>JPEG streams</strong></td>
<td><strong>HEVC streams</strong></td>
</tr>
<tr>
<td>1) Find Image in DICOM object</td>
<td>1) Find lossy image in DICOM object</td>
</tr>
<tr>
<td>2) Re-encapsulate Image fragment selected</td>
<td>2) Find Lossless image in DICOM object</td>
</tr>
<tr>
<td>3) Send new object</td>
<td>3) Re-encapsulate lossy and lossless fragments selected</td>
</tr>
<tr>
<td></td>
<td>4) Send new object</td>
</tr>
</tbody>
</table>

Note: in HEVC case, finding lossy or lossless fragments means reading the DICOM object header index data
Advantages over JPEG2000 (2/2)

- No patent issue, can be used now

- Similar complexity for decoding

- Can similarly use 1 object for multi frame
  - (with all frames independent from one another)
  - OR

- Can use 1 object for 1 frame

- Big lossy compression improvement
  - 6dB at the same bit-rate

- Small lossless gain
  - 4% extra compression efficiency
Is this needed by users?

- Let us talk about what we know: Japan

- Medical facilities have to keep lossless images for X years
  - Then they want to painlessly transform them into lossy archives
  - DICOM can address this problem with the proposed scheme

- Remote facilities need to be connected to each others
  - Sending lossless files as a bulk and on the fly is not an option
  - Their internal network is often not strong
  - DICOM can address this problem with the proposed scheme

- The needs exists → there is value in supporting the syntax

- This scheme comes from medical facilities requirements
  - This is not a lab pet project we want to see taking off
Technical Part

LOSSLESS CODING WITH HEVC
Lossless in HEVC

- HEVC is most famous for its lossy compression
  - Its “main” frequency transform (DCT) is lossy

- HEVC has DPCM for lossless coding
  - Less efficient than frequency transforms

- How to get both efficiency + lossless?
  - Use both DCT and DPCM
Layered coding in HEVC

- Encode a picture in lossy mode
  - Good compression ratio with DCT

- Encode the remaining noise in the enhancement layer
  - Guaranteed Lossless results

What does a scalable stream look like?

- Different layers can be sent together or separately
- Adapt shape of stream to defined needs
  - In our case, separating layers is more interesting
Layered Lossless Compression

Key point: Divide images into two parts
- 1 light layer with good quality but lossy data
- 1 heavier layer with information necessary for lossless reconstruction

The ratio quality/weight of lossy base is adjustable
- Encoder decides
- Optimal compression when lossy is at around 30% of bitrate

Necessary data for true lossless ~70%
Visually lossless base ~30%
Complete file
Scalability: What can you fear?

- HEVC’s scalability can have up to 63 layers
  - Some that can be decoded without others
  - Some that need other streams to be decoded

- Diverse types of scalability
  - Temporal scalability
    • Higher frequency when decoding more layers
  - Quality scalability
    • Higher quality when decoding more layers
  - Color Gamut scalability
    • Changing color gamut depending on the decoded layer
  - Spatial scalability
    • Change the frame size depending on the layer decoded

Usually, decoders need to be ready for all these cases.
Example of a possible scalable stream:

- **Enhancement Layer**
  - B pictures
  - Make reference to 2 other pictures

- **Independent non-base layer**
  - I pictures
  - Do not require other pictures
  - B pictures
  - Make reference to 2 other pictures
  - P pictures
  - Make reference to 1 other picture

- **Base Layer**
  - I pictures
  - Do not require other pictures
  - B pictures
  - Make reference to 2 other pictures
  - P pictures
  - Make reference to 1 other picture

Picture number
Fujitsu’s proposed scalability

- Only 2 layers
- Only quality scalability
- First layer only has I pictures
  - Each picture is independently coded
- Second layer only has P pictures
  - Only need to encode remaining noise (by “DPCM”)
  - Second layer carries the information for lossless reconstruction
- Decoder will **never** have to handle more complexity than that:

```
Enhancement Layer

P  P  P  P  P  Lossless

Base Layer

I  I  I  I  I  Lossy
```
Where is the motion compression?

- Each frame is encoded independently from others
- No motion compression
- Every frame is essentially a **still picture**
What is the expected complexity?

- Fujitsu estimates performances can be:
  - **Software** decoding of 4:0:0 12 bits 510*510 frames at 30fps
  - **Software** encoding of 4:0:0 12 bits 510*510 frames at 4fps
  - Deployable on existing platforms without extra-hardware
FORWARD
Can we help?

Do you need additional details? Explanations?

- Please reach Guillaume Barroux at guillaume.b@jp.fujitsu.com
- Also available for discussions over phone or Skype calls

We are also available to help with:

- Showing Fujitsu encoded tests
- Help you set your own test using the official HEVC reference software
  
  • [https://hevc.hhi.fraunhofer.de/shvc](https://hevc.hhi.fraunhofer.de/shvc)

Copyright 2016 FUJITSU LABORATORIES LTD.
shaping tomorrow with you