GPU-Based DWT Acceleration for JPEG2000

Jiří Matela (matela@ics.muni.cz)
Masaryk University

Memics, Znojmo, 2009–11–14
The Advanced Network Technologies Laboratory at FI

- Conducting research in the field of collaborative environments
- Transfer collaborative environments into the digital world
- Exploit the visualization and communication capabilities
- Visual information and visual communication are very important part
Example of Visualization and Visual Communication in Collaborative Environments

High Performance Computing class taught live from LSU
Example of Visualization and Visual Communication in Collaborative Environments

Ocean currents visualization and HD communication
Example of Visualization and Visual Communication in Collaborative Environments

Hi-Res pathological image and HD video communication
High Resolution Visual Communication is Bandwidth Demanding

- Uncompressed HD video $\sim 1.5$ Gbps
- 4K animation $\sim 6.4$ Gbps
- Network connections may not have enough of bandwidth
- Several visualization might be shared between participants at the same time
- Need for compression
JPEG2000 Compression Standard

- Successor to the popular JPEG
- Superior compression performance
- Lossy and lossless compression
- Error resilience
JPEG2000 Compression Standard

- Multiple resolution representation
JPEG2000 Compression Standard

- Progressive transmission by pixel and resolution accuracy
Discrete Wavelet Transform (DWT)

- The key prerequisite of JPEG2000 compression process
- Digital signal processing technique allowing to study a digital signal in different resolutions
- Most of advanced features of JPEG2000 rely on DWT
- By application of 2D DWT, the source image is decomposed into four subbands (denoted LL, HL, LH, HH)

Result of application of 2D DWT
• Capable of running thousands of threads in parallel
• Threads are grouped into so called **thread blocks**
• Threads within a block cooperates among themselves by sharing data through a **shared memory**
• Common work flow is to copy data from RAM to GPU **global memory** then fetch data from global memory into thread block’s shared memory
• Global memory access pattern is very important performance consideration
• Adjacent threads access adjacent locations in global memory
• So that access is **coalesced**
• Also adjacent threads should execute same instructions – SIMD or SIMT model
Accelerating DWT using CUDA

- Proposed implementation accelerates lifting scheme of DWT
- Source image is partitioned into square-shaped pieces
- Each thread block (rectangular in shape) process one such piece of image data
- Twice as much data in shared memory then threads in block
DWT Computation

• Within a block data are transformed by repeated application of low-pass and high-pass filters

\[
\begin{align*}
    d_{i}^{1} &= d_{i}^{0} - \frac{1}{2}(s_{i}^{0} + s_{i+1}^{0}) \\
    s_{i}^{1} &= s_{i}^{0} + \frac{1}{4}(d_{i-1}^{1} + d_{i}^{1})
\end{align*}
\]

• Initially filters are applied to each row
• Resulting into coefficients of low-pass subband at even positions and coefficients of high-pass subband at odd positions
DWT Computation

- Filters are applied to each column
- Resulting into foursomes of coefficients of four subbands of 2D DWT (LL, HL, LH, HH)
- The final step is to move data from shared memory back to the global memory.
- Particular subbands however needs to be stored separately in global memory.
- It is important comply with global memory coalesced access.
- Even lines are stored first and first half of threads store LL coefficients and the second half stores HL coefficients.
Performance evaluation

- Performance of proposed GPU accelerated DWT compared to DWT in JasPer
- JasPer is referential implementation of JPEG2000
- HD frame processed in 0.81 ms
- About 68 times faster

<table>
<thead>
<tr>
<th>Implementation</th>
<th>512×512</th>
<th>1920×1080</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>JasPer</td>
<td>6ms</td>
<td>55ms</td>
<td>N/A×</td>
</tr>
<tr>
<td>CUDA DWT</td>
<td>0.12ms</td>
<td>0.81ms</td>
<td>67.9×</td>
</tr>
</tbody>
</table>
Conclusion

• Real-time compression of high definition multimedia data is necessary in collaborative environments
• JPEG2000 offers advanced features demanded by various collaborative applications
• JPEG2000 is computationally very demanding
• The proposed GPU accelerated implementation of DWT is very promising
• Considering obtained results we believe there is a room to accelerate the other components of the JPEG2000 algorithm and therefore allow for more detailed and more expressive visualisation shared in collaborative environments
Thank you for your attention!

Q?/A!

matela@ics.muni.cz