Real-Time Network Streaming of Dynamic 3D Content with In-Frame and Inter-Frame Compression

P. Gasparello, G. Marino, F. Bannò, F. Tecchia, M. Bergamasco
Introduction

• The main idea is to *capture* an OpenGL command sequence in real-time and to *transmit* it towards one or more remote node to be *replicated*.

• The proposal is the exploitation of both In-Frame and Inter-Frame redundancies in order to reduce the amount of data to be sent, and allow the system to work even when the bandwidth is limited.
OpenGL commands are captured on the Master Node by means of a custom OpenGL driver. Those calls are then compressed, packed, and sent to the one or more slave nodes.
Compressing Real-time 3D Stream

OpenGL calls are intercepted on the fly (at driver level), and compression/decompression occurs in real-time

[Diagram showing the process of compressing real-time 3D stream with blocks labeled as Generic OpenGL calls, Packetizer, Previous Data Frame, Diff, LZO, Internet, Unpacketizer, Previous Data Frame, Diff, LZO, Geometry Definitions, Geometry Compr., Geometry Decompr.]
3D Data structures are inferred by analysing the overall OpenGL command stream:

- Immediate mode
- Display lists
- Vertex arrays
- Vertex buffer objects
Geometry Compression

**1° Stage**
- Manifoldize
- Connectivity
- Attributes
- Quantization

**2° Stage**
- VBE
- Connectivity Symbols
- Attributes Symbols
- Vertex Predict

**3° Stage**
- SC
- SC
- SC
- SC
- SC
- SC

**Uniform Quantization of Coordinates**
Geometry Compression

1° Stage
- Manifoldize
  - Connectivity
  - Attributes

2° Stage
- Quantization
- VBE
  - Connectivity Symbols
  - Attributes Symbols

3° Stage
- SC
- SC
- SC
- SC
- Arith Coder
- Output Buffer

Single Rate
Valence Based Encoding

(ADD 6)
Geometry Compression

1° Stage
- Manifoldize
  - Connectivity
  - Attributes

2° Stage
- Quantization
  - VBE
    - Connectivity Symbols
    - Attributes Symbols
  - Vertex Prediction

3° Stage
- SC
- Arith Coder
- Output Buffer

Vertex Prediction by Parallelogram Rule

Predicted vertex
Prediction error
Geometry Compression

1° Stage
- Model
  - Manifoldize
    - Connectivity
    - Attributes

2° Stage
- VBE
  - Connectivity Symbols
  - Attributes Symbols
  - Quantization
    - Vertex Predict

3° Stage
- SC
  - Arith Coder
    - Output Buffer

Entropy Compression by a Context Based Arithmetic Encoding

Multiple Contexts for a better statistical modeling
Exploiting Frame-to-Frame Coherence

Inter-frame redundancy is exploited by using *VCDIFF* (RFC3284)
Mesh Compression Results

- Compression Ratio up to 3.5% of the original size (saving up to 96.5%)
- Throughput of compression is on average from 100K up to 250K vertex per second
- Test Machine:
  - Intel i5 M520 2.4GHz

<table>
<thead>
<tr>
<th>Model</th>
<th># Vert.</th>
<th>Time [s]</th>
<th>Thr. [vps]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raptor</td>
<td>1M</td>
<td>4.132</td>
<td>242K</td>
</tr>
<tr>
<td>Dragon</td>
<td>656K</td>
<td>5.851</td>
<td>112K</td>
</tr>
<tr>
<td>Eros</td>
<td>477K</td>
<td>3.940</td>
<td>121K</td>
</tr>
</tbody>
</table>
Streaming Measurements

Motorbike

Spheres

Rollercoaster
### Streaming Measurements [Bytes]

<table>
<thead>
<tr>
<th>Model</th>
<th>Frame 0</th>
<th>Frame 1</th>
<th>Frame 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spheres</td>
<td>33,375</td>
<td>4,602</td>
<td>4,602</td>
</tr>
<tr>
<td>Motorbike</td>
<td>6,147,768</td>
<td>1,705</td>
<td>1,705</td>
</tr>
<tr>
<td>Rollercoaster</td>
<td>18,359,992</td>
<td>18,410</td>
<td>18,410</td>
</tr>
</tbody>
</table>

#### With only LZO

#### With LZO plus In-Frame Compression

<table>
<thead>
<tr>
<th>Model</th>
<th>Frame 0</th>
<th>Frame 1</th>
<th>Frame 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spheres</td>
<td>10,663</td>
<td>4,602</td>
<td>4,602</td>
</tr>
<tr>
<td>Motorbike</td>
<td>2,093,678</td>
<td>1,705</td>
<td>1,705</td>
</tr>
<tr>
<td>Rollercoaster</td>
<td>7,655,571</td>
<td>18,410</td>
<td>18,410</td>
</tr>
</tbody>
</table>

#### With LZO plus Inter-Frame Compression

<table>
<thead>
<tr>
<th>Model</th>
<th>Frame 0</th>
<th>Frame 1</th>
<th>Frame 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spheres</td>
<td>33,375</td>
<td>3,931</td>
<td>3,931</td>
</tr>
<tr>
<td>Motorbike</td>
<td>6,147,768</td>
<td>208</td>
<td>208</td>
</tr>
<tr>
<td>Rollercoaster</td>
<td>18,359,992</td>
<td>92</td>
<td>92</td>
</tr>
</tbody>
</table>

#### With LZO plus In-Frame plus Inter-Frame Compression

<table>
<thead>
<tr>
<th>Model</th>
<th>Frame 0</th>
<th>Frame 1</th>
<th>Frame 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spheres</td>
<td>10,663</td>
<td>3,931</td>
<td>3,931</td>
</tr>
<tr>
<td>Motorbike</td>
<td>2,093,678</td>
<td>208</td>
<td>208</td>
</tr>
<tr>
<td>Rollercoaster</td>
<td>7,655,571</td>
<td>92</td>
<td>92</td>
</tr>
</tbody>
</table>
## Streaming Measurements [ms]

<table>
<thead>
<tr>
<th>Model</th>
<th>Frame 0</th>
<th>Frame 1</th>
<th>Frame 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spheres</td>
<td>12.6673</td>
<td>4.0159</td>
<td>3.95106</td>
</tr>
<tr>
<td>Motorbike</td>
<td>1,167.56</td>
<td>27.4411</td>
<td>29.8014</td>
</tr>
<tr>
<td>Rollercoaster</td>
<td>20,523.3</td>
<td>41.9717</td>
<td>32.6774</td>
</tr>
</tbody>
</table>

With only **LZO**

<table>
<thead>
<tr>
<th>Model</th>
<th>Frame 0</th>
<th>Frame 1</th>
<th>Frame 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spheres</td>
<td>80.5976</td>
<td>7.16921</td>
<td>4.81178</td>
</tr>
<tr>
<td>Motorbike</td>
<td>9,440.13</td>
<td>17.3465</td>
<td>18.9491</td>
</tr>
<tr>
<td>Rollercoaster</td>
<td>32,382.5</td>
<td>54.3147</td>
<td>28.9925</td>
</tr>
</tbody>
</table>

With **LZO** plus **In-Frame** plus **Inter-Frame** Compression
Sample Applications
Conclusions and Future Work

• We realized a system capable of distributing real-time generated content using on-the-fly compression/decompression of OpenGL command stream.

• We advanced beyond the state-of-the-art by introducing *in-frame geometry compression* of 3D data structures and by exploiting *inter-frame redundancy*.

• Measurements suggest a significant potential whenever the amount of data bandwidth is limited, for instance in Internet distributed interactive applications.

• Future works are focusing on:
  
  – investigating techniques for progressive transmission of frame data containing 3D model descriptions.
  
  – the introduction of a state tracker able to make the system robust in case of data loss. Data packets containing OpenGL state modifications could be sent over TCP connections, while the other ones could be sent with faster UDP datagrams.
Acknowledgments

• This work was funded by the project BEAMING, acknowledging the financial support of the Seventh Framework Programme for Research of the European Commission DG-INFSO-D2, undergrant number 248620.

• Authors wish to thank the Salford CVE group and the other laboratories that are using our system for the feedback and for the help given on improving our software.
thank you!

p.gasparello@sssup.it