LightDB: A DBMS for Virtual Reality Video

Manos Theodosis, Raahul Acharya, Mia Sturzu
PROBLEM

CURRENT PICTURE

CURRENT VIDEO

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>24</th>
<th>0</th>
<th>24</th>
</tr>
</thead>
<tbody>
<tr>
<td>48</td>
<td>0</td>
<td>48</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>256</td>
<td>0</td>
<td>256</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>48</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>256</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>24</td>
<td>0</td>
<td>24</td>
<td>0</td>
</tr>
<tr>
<td>48</td>
<td>0</td>
<td>48</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>256</td>
<td>0</td>
<td>256</td>
<td>0</td>
</tr>
</tbody>
</table>

T = 0

T = 1

T = 2
VIRTUAL, AUGMENTED, and MIXED REALITY (VAMR)

Spherical

Non-Uniform Sampling

Data and Time Intensive, but high Latency b/c of constant sampling
WHY DO PAST SOLUTIONS NOT WORK?

Facebook Surround 360, Youtube VR, etc. don’t work because *impedance mismatch*.

Treat Data like 2D, so..

View Agnostic

Manual “Plumbing” Required

Incompatible to different 2D layouts
Data Representation 1: Spherical Panoramas

Fixed Location, Any Direction

Image Projection + Stitching

Stereoscopic + Depth

Side

Front

Depth Map

Stream 1

Stream 2

Stream 3
Data Representation 2: Light Fields

Variable Location, Any Direction

Input: Location, Direction

Output: Color, Intensity

Light Slabs + Interpolation

Extrapolate: red
Video Encoding/Streaming

Frame-Storage/Redundant Compression

Tiling

GOP- adaptive streaming
data model

(RasDaMan, SciDB, Oracle Multimedia)

(x, y, t)

VS

(x, y, z, t, θ, φ)

(LightDB)
operators

1. select

2. partition

3. merge

4. translation

\[ + = x \]

\[ y = z \]
operators

5 transform

map(            , f)

\[
\begin{array}{c}
\hat{f}(\bullet) & \hat{f}(\bullet) & \hat{f}(\bullet) & \hat{f}(\bullet)
\end{array}
\]

0.2 0.76

0.2 0.48 0.76

\{P_1, P_2, P_3\}

\{q(P_1), q(P_2), q(P_3)\}

MAP

INTERP

SUBQ
input & creation

- h.264
- decode
- tlf
- encode
- scan
- store
- input
- 0.8 0.3 0.6
- tlf
- index
- creation
architecture

Buffer Cache

Query Processor

Persistent Storage

Catalog

Rule-Based Optimizer

input

queries

physical plans

output

metadata

GOP_1

... 

GOP_m

metadata

metadata

metadata

metadata

TLF_1

TLF_n

TLF_1

TLF_n

input

output
 optimizer

Declarative Query

Logical Plan

Initial Physical Plan

Final Physical Plan

GPU? FPGA? CPU? implementation details

Physical Op → Homomorphic Op
indexing

Spatial Index

R1
R2
R3
R4
R5
R6

GOP Indices

GOP_1
GOP_2
GOP_3
GOP_4
...
GOP_n

Select(t ∈ [a, b])

Tile Index

metadata
other metadata

Tile 1
Tile 1
Tile 1

R1
R2
R3
R4
R5
R6
evaluations

- up to -99%

FFmpeg & OpenCV  LightDB & SciDB

FPGA  vs  CPU

FPS: 60  FPS: <15

LightDB  Others

programmability  performance

hardware acceleration
evaluations

- TLF performance 💚
- Slab performance 💚
- Optimization performance 💚

Operator performance

Index performance
conclusion

- Related work
- Possible next steps