Hardware Design, Structure, and Networking
Structure Design
Different Designs for Different Goals

v1.0: Rander, Narayanan, Kanade 1995

v2.0: Vedula, Saito, Kanade 1998

v3.0: Matthews, Baker, Kanade 2002
Structure Design
Different Designs for Different Goals

Kyoto U. 2005

Kyoto U. 2008
ICCV2009 demo
Dependencies
Different Designs for Different Goals

Input / Requirements

- Cable length
- Image resolution
- Area size
- FoV

Output / Conclusions

- Interface
- Sensor
- Framerate
- Shutter
- DoF
- F-number
- Camera
- Lighting
- Lens
Structure Design
Different Designs for Different Goals

Remote studio

Goal: Complete Light-field Capture

Communication
Cameras for Complete LF Capture

Dense low-res cameras or sparse high-res cameras?

Dense cameras & Low sampling rate

Sparse cameras & High sampling rate

for visibility

for detailed geometry/texture
Cameras for Complete LF Capture

Dense low-res cameras or sparse high-res cameras?

Dense low-res cameras / simple task ⇔ Sparse high-res cameras / complex task

++ Visibility

++ Resolution
Truncated Pentagonal Hexecontahedron

6 Pentagons, 40 Hexagons, 10 Trimmed Panels

Face-transitive solid (Catalan solid)
Optimized Placement of Cameras
Minimizing Differences in Camera Angles

Camera $i$
Edge $ij$
Camera $j$
Dome Center
Angle $\theta_{ij}$
The Panoptic Studio
Modularized Design with 20 Panels
The Panoptic Studio
Modularized Design with 20 Panels

- HD Camera
- Kinect
- VGA Camera
Diameter: 5.49 m
Dependencies
Different Designs for Different Goals

Input / Requirements: Area size, Cable length, FoV, Light-field resolution, Framerate, DoF, F-number

Output / Conclusions: Interface, Sensor, Camera, Lighting, Lens
The Panoptic Studio
Modularized Design with 20 Panels
The Panoptic Studio
Modularized Design with 20 Panels

480 VGA Cameras
31 HD Cameras
10 Kinects
Capture System
Heterogeneous distributed system

- Master Node
- Clock System
- Local Node
- Diversity
- Scalability

- 24 VGA Cameras
- 480 VGA Cameras (20 Panels)
- 31 HD Cameras (31 Panels)
- 10 Kinects
- 5 DLP Projectors (5 Panels)
Synchronization
Synchronization
Why do we need this?

Triangulation assumes same time instant
Synchronization

Different standards for machine-vision and production

Note: Camera can be a “generator” in both cases
Timestamp for **machine-vision cameras**

Host-side or camera-side

NTP (host-side time-sync)
Timestamp for **machine-vision** cameras

Host-side or camera-side

PTP
(camera-side time-sync)

NTP
(host-side time-sync)
Synchronization
Design space

- Synchronized shutter (w/ timestamp)
  - Trigger (+LTC or PTP)
  - Genlock (+LTC)
  - PTP + synchronous free run

- Async with timestamp
  - Clapping / Flashing
  - PTP
  - Multiple LTC recording

- (None)
  - Clapping / Flashing
Synchronization
Trigger vs Phase-lock (Genlock)

Propagation delay depends on the camera clock and circuit (observable by strobe output)

typ. 10ns for non-isolated IO-pins

typ. 10us for opt-isolated IO-pins

Obj@100km/h travels 0.27mm in 10us
First $n$ pixels are used for meta-data

Camera clock
   → Camera should know the world clock
Image count
   → Useful for frame-drop detection

Manchester-coded binary audio signal

Originally 24/25/30Hz
   → Fast-forwarded for our 100Hz system
Not for sync
   → Frame start/end in time is not accurate
# Synchronization

## Decision principle

<table>
<thead>
<tr>
<th>Synchronized shutter</th>
<th># of cables</th>
<th>Scalability</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigger (+LTC or PTP)</td>
<td>1 (+1)</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>Genlock (+LTC)</td>
<td>1 (+1)</td>
<td>++</td>
<td>++</td>
</tr>
<tr>
<td>PTP + synchronous free run</td>
<td>0</td>
<td>+</td>
<td>++</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Async with timestamp</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Clapping / Flashing</td>
<td>0</td>
<td>+++</td>
<td>?</td>
</tr>
<tr>
<td>PTP</td>
<td>0</td>
<td>+</td>
<td>++</td>
</tr>
<tr>
<td>Multiple LTC recording</td>
<td>1</td>
<td>++</td>
<td>+</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(None)</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Clapping / Flashing</td>
<td>0</td>
<td>+++</td>
<td>?</td>
</tr>
</tbody>
</table>
Capture System

Heterogeneous distributed system

- **Master Node**
- **Clock System**
- **Local Node**
- **Diversity**
- **Scalability**

**Capture System Components**

- 24 VGA Cameras
  - 480 VGA Cameras (20 Panels)
- 31 HD Cameras (31 Panels)
- 10 Kinects
- 5 DLP Projectors (5 Panels)
VGA system

480 sensors as a *single* camera

24 Image Sensors

Capture Node 01

PCI-Express cable (+ extender)

Capture Node 02

Controller
(Clock Generator)

Capture Node 20

Synchronized shutter
(all 480 sensors)

Timestamped
(frame-embedded)

Clock, Sync, Timestamp

Host bus interface (1x PCIe)

- EBU N14, SMPTE RP154
- SMPTE-12M
Synchronization

Take-home messages

• Shutter-level hardware sync is best if you can afford it.
  • Additional cables per camera
  • Signal generator / distributer

• PTP + synchronous free run can be an option for small camera system
  • All cameras should be in a single LAN

From FLIR/Pointgrey and ViewPLUS
Capture System
Capture System

Why do we need PC system?

Storage

Computation

Visualization / Broadcasting
Capture System
Design space

Distributed

Decentralized

Centralized
Capture System
Design principle

Distributed
- Scalability
- Diversity
++ Scalability
++ Modularity
++ Single-view processing
-- Multi-view processing

Decentralized
- Scalability
- Diversity
+ Scalability
+ Modularity
++ Stereo processing
- Multi-view processing

Centralized
- Scalability
- Diversity
-- Scalability
-- Modularity
-- Single-view processing
++ Multi-view processing
Capture System
Heterogeneous distributed system

- **Local Node**
  - 24 VGA Cameras
  - 480 VGA Cameras (20 Panels)

- **Master Node**

- **Clock System**

- **Local Node**

- **Kinect Master Node**
  - 10 Kinetics

- **31 HD Cameras**
  - (31 Panels)

- **5 DLP Projectors**
  - (5 Panels)

- **Scalability**
- **Diversity**
PC System
Take-home messages

- Distributed architecture is required for large-scale system
  - Modular design = easier maintenance

- Centralized architecture can be better for smaller system
  - Simpler real-time multi-view processing
    - No need to exchange images among nodes
Redesign from scratch?
Dependencies
Different Designs for Different Goals

- Light-field resolution
- Cable length
- Laser resolution
- Image resolution
- Framerate
- Interface
- Sensor
- Shutter
- Lighting
- Lens
- Area size
- FoV
- DoF
- F-number

Realtime-processing
Visualization
Interaction / feedback
Broadcasting

Input / Requirements  Output / Conclusions
Redesign from scratch?
Diversity / scalability / realtime-ness

Camera node
- 4-camera nodes
  - 4-bus HA PCIe x4
  - USB3 or GigE

GPGPU node
- 4-GPGPU nodes
  - 4-bus HA PCIe x4
  - PCIe x8

Visualization node
- 4-Visualization nodes
  - 4-bus HA PCIe x4
  - PCIe x16

Storage node
- 4-Storage nodes
  - SSD PCIe x8

Trigger generator

HW-based Shared Memory (40Gbps per point-to-point)
Redesign from scratch?
Diversity / scalability / realtime-ness

Components

Quad-bus Host Adapter

Optical Interconnect

AVALDATA APX-7142

ViewPLUS SyncUSB3

Trigger generator / distributer

Camera node

4-bus HA PCIe x4
4-bus HA PCIe x4
4-bus HA PCIe x4
4-bus HA PCIe x4
Opt IF PCIe x8

HW-based Shared Memory (40Gbps per point-to-point)

USB3 or GigE

Trigger generator
Discussions
Limitations, future work

Large-scale, Outdoor
- Cabling?
- Pan/tilt robot?
- Lens control?

Underwater
- Cabling?
- Illumination?

Microscale
- DoF?
- Multi-view?
Discussions
Limitations, future work

3D surface capture via unknown refractive media
Discussions

Limitations, future work

3D surface capture via \textit{unknown} refractive media
Panoptic Studio
Heterogeneous distributed system

24 VGA Cameras
480 VGA Cameras (20 Panels)

31 HD Cameras (31 Panels)

10 Kinects

5 DLP Projectors (5 Panels)

Master Node
Clock System
Local Node
Kinect Master Node

Scalability
Diversity

Thank you!