Augmented Visualization with Natural Feature Tracking

Gábor Sörös  ETH Zurich
Peter Rautek  Vienna University of Technology
Hartmut Seichter  Graz University of Technology
Eduard Gröller  Vienna University of Technology
The Magic Lens Metaphor

[Leonardo Da Vinci, 16th century]
The Magic Lens Metaphor (2)

[Kalkofen et al. (2007)]

[Bichlmeier et al. (2007)]
The Magic Lens Metaphor (3)

[Sanneblad et al. - Ubiquitous Graphics (2005)]

[Reitmayr et al. - Augmented Maps (2005)]

[Spindler et al. - PaperLens (2009)]
Interaction with Displays

Can we do that **without any markers**?

[Ballagas et al. - Point & Shoot (2005)]

[Boring et al. - Shoot & Copy (2007)]

[Hardy et al. - Touch & Interact (2008)]

[Boring et al. - TouchProjector (2010)]
AUGMENTED VISUALIZATION
Idea

- Assumptions
  - Visualizations are shown on monitors or large projections
  - The visualized scene is rendered/virtual
  - Group of people with different interests

- Goals
  - 6DoF input
  - personal overlay information for each participant
  - widely available & cheap HW
  - avoid any special markers

conventional visualization

augmented visualization
Idea

- **Input**
  - touch screen
  - buttons
  - accelerometer
  - camera tracking

  ![Image of a head and a phone](image)

  *Pose estimation of each participant*

- **Output**
  - ”magic lens” display
  - vibration
  - etc.

  ![Image of a person holding a phone](image)

  *Personal overlays for each participant*
SYSTEM OVERVIEW

Rendering Component

Tracking Component

Mobile Component
System Overview

Mobile component

- grab camera images
- stream images to tracker

Tracker component

- (re)create feature database
- feature extraction
- matching
- pose estimation
- send pose

Rendering component

- render reference image
- send reference image to tracker
- render personal view
- stream personal overlay to mobile device

present personal view

send user input
Rendering Component

- VolumeShop volume rendering framework
- Multiple viewports (common view and personalized views)
- XML-based Plugins
- Simple remote control extension
SYSTEM OVERVIEW

**Rendering Component**  
**Tracking Component**  
**Mobile Component**
Tracking Component

A) Detection & Initialization:
1. feature extraction
   - FAST [Rosten & Drummond 2006]
2. feature description
   - modified SURF (at multiple scales) [Bay et al. 2008] [Wagner et al. 2009]
3. feature matching
   - brute force, Euclidean distance (database not too large)
4. outlier detection & pose estimation
   - geometric constraints & RANSAC

B) Patch Tracking:
   - prediction of affinely warped patches in subsequent images
   - up to extreme tilts and lighting changes
SYSTEM OVERVIEW

Rendering Component

Tracking Component

Mobile Component
Mobile Component

- HTC Desire (Google Nexus One)
  - Hardware H.263 and JPEG encoder
  - WiFi connection
  - Camera

- H.263 video streaming
  - Android Media API + localhost loopback
  - RTP/RTSP
  - Too much delay

- JPEG frame streaming
  - Android Camera API
  - UDP
  - GStreamer node in the tracker
  - Low delay, good quality

- User Interface
  - Touch screen
  - Pointer
  - Overlay images
RESULTS
Results – Video
## Results – Frame rates

<table>
<thead>
<tr>
<th>Camera resolution</th>
<th>JPEG quality</th>
<th>Overlay resolution</th>
<th>Average FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>320x240 80%</td>
<td>128x128</td>
<td>19.2</td>
<td></td>
</tr>
<tr>
<td>320x240 80%</td>
<td>256x256</td>
<td>15.2</td>
<td></td>
</tr>
<tr>
<td>320x240 80%</td>
<td>320x240</td>
<td>14.7</td>
<td></td>
</tr>
<tr>
<td>320x240 80%</td>
<td>800x480</td>
<td>9.5</td>
<td></td>
</tr>
<tr>
<td>640x480 80%</td>
<td>256x256</td>
<td>8.8</td>
<td></td>
</tr>
<tr>
<td>640x480 30%</td>
<td>256x256</td>
<td>9.2</td>
<td></td>
</tr>
<tr>
<td>320x240 30%</td>
<td>256x256</td>
<td>16.8</td>
<td></td>
</tr>
<tr>
<td>176x144 80%</td>
<td>256x256</td>
<td>19.1 (degraded)</td>
<td></td>
</tr>
<tr>
<td>176x144 30%</td>
<td>256x256</td>
<td>failed</td>
<td></td>
</tr>
<tr>
<td>320x240 80%</td>
<td>128x128</td>
<td>14.2 (2 users)</td>
<td></td>
</tr>
<tr>
<td>320x240 80%</td>
<td>128x128</td>
<td>11.4 (3 users)</td>
<td></td>
</tr>
</tbody>
</table>
Results – Content

- Robust tracking when more than 100 keypoints
- This assumption usually holds in scientific visualization
- Distance range is limited
Conclusion

- So far...
  - Interactive frame rates
  - No special hardware required
  - Easy to implement and deploy
  - Simple GUI

- The next steps...
  - Porting the tracker onto the phones
  - Automatic camera calibration
  - Extension to multiple targets (multiple displays)
  - New applications
Thank you for your attention!

谢谢