Real-time 3D Reconstruction

Tom Landry
Senior advisor, Vision and Imaging Team, CRIM
Anti-agenda… this presentation is not...

A math course, thank God!

A programming course, but that would be fun

A true state-of-the art, because it’s a very active field

A pure realtime GPU-centric presentation, sorry
Agenda

Quick overview of image-based reconstruction
Common types 3D sensors
A few examples of popular commercial solutions
Tips for easier 3D scanning
Common post-processing tools
Structure from Motion (SFM)

Produces sparse point cloud and camera poses

Requires:
- Good texture
- Similar illumination
- Visual overlap with viewpoint variety
From SFM to Multi-View Stereo

Start with SfM output

Estimate depth map and normal for every pixel

Fuse depth maps in 3D and get dense point cloud

Construct 3D mesh with Poisson reconstruction

Mostly offline, batch processing

Source: Universität Zurich
### Images to 3D model pipelines

<table>
<thead>
<tr>
<th></th>
<th>Point cloud</th>
<th>Dense cloud</th>
<th>Surface</th>
<th>Textured</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bundler</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMVS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>COLMAP</td>
<td></td>
<td></td>
<td>CUDA only</td>
<td>CUDA only</td>
</tr>
<tr>
<td>Meshlab</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MVE</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MVS-texturing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>openMVG</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>openMVS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Theia</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VisualFSM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Source: blog.mardy.it
What about GPU?

- Image processing (filtering, warping, feature extraction)
- Multiple-view geometry (correspondence search, cost functions)
- Optimization (regularization, gradient computation)

Did you say manual? Post-processing? Hey, it’s not real time!
Simultaneous Location and Mapping

Unlike marker-based or fixed-rigged

Tracking set of points through successive frames, then triangulate 3D position

Optimise best camera-point configuration to minimize reprojection error

Often uses Bundle Adjustment

\[ \chi^2(T_i) = \sum_{z_j \in Z_i} (z_j - \hat{z}(T_i, x_j))^2 \]

Source: Google Tango
From Monocular SLAM to Dense tracking

Generating and tracking sparse features

Creation of dense 3D surface texture-mapped model

Model built by depth maps from multi-view reconstruction

Source: Newcombe, DTAM
What to make of this?

- Mix of techniques, pipelines and algorithms based on visual information
- Commercial toolboxes hides part of the complexity away
- Still a good idea to know what’s underneath the hood
- Offline processes from 2010 might become 2018 realtime ones

What if we could have a shortcut to get that 3D info?

- Of course, shortcut will inevitably become a maze
3D sensors
**Laser scanner.**
Source: Wenzel metrology

**Structured light.**
Source: Depthbiomechanics.co.uk

**Time of flight.**
Source: Basler
High-tech sensor hooked to a high-end PC
Easy to use, fully real-time, customizable
Needs good conditions for high framerates
Two processors: iPad and PC
Real-time WiFi RGB-D stream
Cloud-based model?

Originates from RGBD Demo
Traces of Open Source

Source: Skanect
Normal scan is aligned to model
Statistics and heuristics normalize scan
Measurements of landmarks
Obtention of articulated 3D model

First do a “raw” scan in realtime
Send to cloud for high-res processing
Get back enhanced scan

Probably photogrammetry fusion
Markers

Color not always used during tracking
Texture is always helpful
Add “noisy” or structured trackers

Not AR-like tracker use case
Automatic alignment of scan
Easy identification of landmarks
Enhances tracking stability
During acquisition, special care about:

- Initial conditions, if you’re in for repetitive scans
- Software parameters (scan volume, voxel size, texture resolution)
- Motion in the subject (minimize) and sensor (fluid and predictable)
- Light flares, reflexion, absorption, transparency
- Presence of 3D features or textures
Company based in France, with activities in Japan and Canada

Develops AR/VR/MR toolboxes for retail, architecture, real estate

CRIM is helping WosomTech to characterize sensor localization and measurement errors
- Combine meshes
- Align points (ICP)
- Smooth
- Clean
- Acquire
- Simplify
- Filter
- Resample
- Compare
- Reconstruct, etc.

Source: CRIM

Source: CloudCompare

Source: PCL
Source: CRIM, Stefanka Lingerie

Source: Paraview
What about GPU?

Better performance means better framerate

Higher framerate gives:

- More observations integrated into scan, precision
- Less time to capture subject, less motion of subject blurring scan
- Better tracking because less disparity between frames, fluidity

Better graphic card means manipulation of larger, more precise 3D models, with better framerate e.g. 3D acceleration