Outline

- Introduction
- Features and Feature Matching
- Geometry of Image Formation
- **Calibration**
- Structure from Motion
- Dense Stereo
- Conclusion
Perspective Camera Calibration

- capture checkerboard images from different points of view
- detect corners in images (2D-3D correspondences)
- utilize set of known, planar 3D points for calibration
Perspective Camera Calibration

- determine intrinsics (shared by all images):
  - focal length
  - principal point
  - aspect ratio
  - distortion parameters

- determine extrinsics (for each image):
  - rotation
  - translation
Perspective Camera Calibration

Initialization:
- \( \mathbf{x} = \mathbf{PX} \)
- Direct Linear Transform (DLT) of projection equation
  \[ [\mathbf{x}] \mathbf{X} \mathbf{P} = 0 \]
- set of 2D-3D correspondences yields linear system of equations
- solve with SVD for entries in \( \mathbf{P} \)
- decompose \( \mathbf{P} = \begin{pmatrix} \mathbf{KR}^T & -\mathbf{KR}^T \mathbf{C} \end{pmatrix} \)

Heikkila and Silven 1997
Perspective Camera Calibration

Optimization:
- non-linear projection equation with distortion:

$$\arg\min_{\forall R_i, C_i, i \in \{1, ..., N\}, f, c_x, c_y, k_1, k_2} \sum_{i=1}^{N} \sum_{j=1}^{M} \left\| \text{proj}(X_j, f, c_x, c_y, k_1, k_2, R_i, C_i) - x_{ij} \right\|_2^2$$

- minimize using Levenberg Marquardt algorithm

Heikkila and Silven 1997
Perspective Camera Calibration - Example

image size: 800x600 px
focal length: 911 px
principal point: (402,316)
rad dist: -0.2393, 0.1820
error init: 0.47
error opt: 0.37
Calibration of Refractive Camera Model

- intrinsic parameters: *Schiller et al. ISPRS 2008*
- input images from different view points
- Initialization: method by *Agrawal et al., CVPR 2012*
- Analysis-by-Synthesis method for optimization by *Jordt-Sedlazeck et al., ECCV 2012*
Initialization

Given
- flat port underwater cameras are axial cameras
- assume camera's intrinsics are known, i.e. focal length, principal point and distortion parameters
- use 1 checkerboard image (with j*k 2D-3D correspondences from checkerboard detector)

Estimate:
- **interface distance and tilt**, unknown camera pose

Geometric Constraints:
- “Plane of Refraction” (uses parallelism of interfaces)
- “Flat Refractive Constraint” (path consistency)
Plane of Refraction (POR) Constraint

- all ray segments and the normal $\tilde{n}$ lie in common plane (second part of Snell’s law) Plane of Refraction
- POR: $(\tilde{n} \times \tilde{X}_a)$
- 3D point transformed into camera coordinate system must lie in this plane as well: $(R^T X - R^T C)^T (\tilde{n} \times \tilde{X}_a) = 0$
- stacking equations derived from POR to linear system of equations and solve
- yields normal, camera rotation, and camera translation perpendicular to axis

Agrawal et al. 2012
Flat Refractive Constraint (FRC)

- FRC is then used to determine layer thickness, i.e. interface distance and glass thickness
- Result is used in non-linear optimization

$R^T X - R^T C$

Agrawal et al. 2012
Analysis-by-Synthesis Optimization

- results from Agrawal's calibration routine yield interface distance for each image
- further optimization enforces shared glass port parameterization for all images
- Analysis-by-Synthesis optimization is independent of errors in corner detection
Analysis-by-Synthesis Calibration

\[
(X_{cc}^{cw}, \tilde{X}_{cc}^{cw}) = \text{Ray}(x, d, \tilde{n})
\]

\[
(X_{wc}^{cw}, \tilde{X}_{wc}^{cw}) = (RX_{cc}^{cc} + C, R\tilde{X}_{cc}^{cc})
\]

\[
X = X_{cw}^{wc} + \kappa \tilde{X}_{wc}^{wc}
\]

pixel color on checkerboard

error across all pixels and images

non-linear optimization with CMA-ES
## Calibration Results on Real Images

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>monocular: left camera</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$d_1$ in mm</td>
<td>7.88</td>
<td>10.60</td>
<td>51.95</td>
<td>50.53</td>
<td>86.25</td>
<td>95.54</td>
<td>149.97</td>
</tr>
<tr>
<td>$\theta_1$ in °</td>
<td>0.34</td>
<td>0.25</td>
<td>0.29</td>
<td>8.06</td>
<td>29.29</td>
<td>0.12</td>
<td>0.12</td>
</tr>
<tr>
<td><strong>monocular: right camera</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$d_2$ in mm</td>
<td>8.07</td>
<td></td>
<td>44.47</td>
<td>47.30</td>
<td>89.67</td>
<td>99.60</td>
<td>150.0</td>
</tr>
<tr>
<td>$\theta_2$ in °</td>
<td>0.29</td>
<td>0.23</td>
<td>7.97</td>
<td>29.16</td>
<td>0.29</td>
<td>0.39</td>
<td></td>
</tr>
<tr>
<td><strong>rig: left camera</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$d_1$ in mm</td>
<td>19.38</td>
<td></td>
<td>48.07</td>
<td>45.30</td>
<td>79.96</td>
<td>100.39</td>
<td>147.13</td>
</tr>
<tr>
<td>$\theta_1$ in °</td>
<td>0.27</td>
<td>0.11</td>
<td>7.99</td>
<td>28.40</td>
<td>0.22</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td><strong>rig: right camera</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$d_2$ in mm</td>
<td>0.72</td>
<td></td>
<td>58.82</td>
<td>28.34</td>
<td>102.81</td>
<td>113.16</td>
<td>160.47</td>
</tr>
<tr>
<td>$\theta_2$ in °</td>
<td>0.28</td>
<td>0.42</td>
<td>7.79</td>
<td>27.79</td>
<td>0.89</td>
<td>0.09</td>
<td></td>
</tr>
</tbody>
</table>
References

Z. Zhang. Flexible Camera Calibration by Viewing a Plane from Unknown Orientations. ICCV99

J. Heikkilä and O. Silven. A Four-step Camera Calibration Procedure with Implicit Image Correction. CVPR97


References


Further Reading


Software

Bouguet's Matlab Toolbox:
http://www.vision.caltech.edu/bouguetj/calib_doc/index.html#ref

Opencv software: opencv.org

Kiel University (MIP):
http://www.mip.informatik.uni-kiel.de/tiki-index.php?page=Calibration
Wrap Up

• perspective calibration based on checkerboard images yields:
  • intrinsics (focal length, principal point, radial distortion)

• extension on underwater images for refractive camera model:
  • housing parameters (interface distance, interface tilt)

• analysis-by-synthesis approach is independent of errors in corner detection