Evaluation of pairwise calibration techniques for range cameras and their ability to detect a misalignment

Antoine Lejeune
<antoine.lejeune@ulg.ac.be>
David Grogna
Marc Vandroogenbroeck
Jacques Verly

Montefiore Institute, University of Liège

2014 International Conference on 3D Imaging (IC3D 2014)
9 December 2014
Multicamera systems

Pair-wise calibration is a building block of multicamera systems. They provide
- better coverage of large volume;
- multiple point of view of the scene; and
- can increase precision and robustness.

Examples of application: immersive virtual environment, gait analysis of humans, ...
Pairwise calibration

- A multicamera system has to be calibrated, i.e. we need to estimate the relative position between couples of camera.
- For color images, we estimate the intrinsic parameters of each camera and the fundamental matrix between pairs of camera.
- In 3D, we wish to find the rigid body transformation \((R, t)\) that brings points of one camera to the reference coordinate frame of the second camera

\[
P^{(2)} = RP^{(1)} + t
\]

where \(R\) is a rotation, \(t\) a translation vector and \(P^{(i)}\) denotes the coordinates of the 3D point \(P\) as seen from camera \(i\).
Range cameras

- Directly measure a geometric information
- Different technologies:
  - Structured light: Microsoft Kinect (version 1)
  - Time-of-flight: PMD CamCube 2.0, Microsoft Kinect (version 2)
- Nonlinear noise that can vary across the pixels of the image
  - no data at all for some parts of the image
  - there are models of the noise (depending on the technology of the camera)
- There can be problems when naively combining several range cameras with overlapping field of views

Microsoft Kinect (version 1)  PMD CamCube 2.0
Classical technique

- Using a two-sided chessboard to perform a color calibration
- Minimize the reprojection error
- OpenCV implementation
Plane pattern

Depth-based calibration using a plane:

- Plane segmentation can be done in the RGB space or in the depth space
- Point correspondences between the camera are established using the center of the plane
- Rigid body transformation estimated by the least-square minimization of

\[
\sum_i w_i \left\| P^{(2)} - R P^{(1)} - t \right\|^2.
\]
Movement based calibration

- Pairwise calibration using the movement in the scene
- No crafted calibration object
- Permit to detect a misalignment and recalibrate the system when one occurs
- Processing pipeline:
Background subtraction

- Simple background model learned over the first $N_{BG}$ frames:
  \[ B_Z(p) = \max_j (Z_j(p)), \quad j < N_{BG}. \]

- Foreground segmentation based on the estimated noise $\sigma(p)$ at each pixel:
  \[
  F(p) = \begin{cases} 
  \text{true} & \text{if } Z(p) \text{ is valid and } |B_Z(p) - Z(p)| > \lambda \sigma(p) \\
  \text{false} & \text{otherwise}
  \end{cases}
  \]
  with $\sigma_{kinect}(p) = (Z(p))^2$ and $\sigma_{tof}(p) = (A(p))^{-1}$.

- Connected component analysis to filter out small components.
Modeling

- We model the segmented objects $O_i$ as gaussian blobs with a center of mass $\mu_i$ and a covariance matrix $\Sigma_i$.
  - Center of mass computed using only the border pixels

\[ C_1^1 \mu_i^{(1)} \quad \mu_i^{(2)} \quad C_2^2 \]

\[ C_1^1 \mu_i^{(1)} \quad \mu_i^{(2)} \quad C_2^2 \]
Tracking and matching

- Tracking within a single camera is based on the center of mass $\mu_i$ and the covariance $\Sigma_i$.
- Matching between cameras is performed by only using the covariance $\Sigma_i$.
- We use the Kullback-Leibler divergence as a similarity measure.
Misalignment detection

- A “ground truth” pairwise calibration is previously obtained: \((R_{GT}, t_{GT})\)
- We estimate regularly the current transformation using the method based on movement: \((R_t, t_t)\)
- A misalignment is detected when

\[
t_{err} = \| t_{GT} - t_t \| > \tau
\]

or

\[
R_{err} = \left\| \log \left( R_{GT}^T R_t \right) \right\|_F > \theta,
\]

i.e. when the translational error or the angular error are above some threshold.
Evaluation

- Groundtruth \((R_{GT}, t_{GT})\) is computed using the chessboard-based method.

- Evaluation metrics
  - Translational error:
    \[
    t_{err} = \| t_{GT} - t \| 
    \]
  - Angular error:
    \[
    R_{err} = \| \log (R_{GT}^T R) \|_F \in [0; 90^\circ] 
    \]

- 4 spatial configurations and 2 sets of cameras (Kinect-Kinect and Kinect-CamCube) tested

![Diagram of spatial configurations](image.png)
### Comparison (Kinect-Kinect)

![Diagram showing configurations and distances]

<table>
<thead>
<tr>
<th>Configuration</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Translation error (in meter)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chessboard (GT)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Plane</td>
<td>0.094</td>
<td>0.057</td>
<td>0.047</td>
<td>0.155</td>
</tr>
<tr>
<td>Movement</td>
<td>0.076</td>
<td>0.069</td>
<td>0.128</td>
<td>0.189</td>
</tr>
<tr>
<td><strong>Angular error (in degree)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chessboard (GT)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Plane</td>
<td>2.59</td>
<td>1.34</td>
<td>0.45</td>
<td>5.53</td>
</tr>
<tr>
<td>Movement</td>
<td>2.49</td>
<td>1.37</td>
<td>1.96</td>
<td>3.58</td>
</tr>
</tbody>
</table>
## Comparison (Kinect-Camcube)

![Diagram showing configurations (a), (b), (c), (d) with distances 1.6m, 0.28m, 6m, 2m](image)

<table>
<thead>
<tr>
<th>Configuration</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
<th>(d)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Translation error (in meter)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chessboard (GT)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Plane</td>
<td>0.176</td>
<td>0.042</td>
<td>0.044</td>
<td>0.064</td>
</tr>
<tr>
<td>Movement</td>
<td>0.10</td>
<td>0.223</td>
<td>0.169</td>
<td>0.339</td>
</tr>
<tr>
<td><strong>Angular error (in degree)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chessboard (GT)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Plane</td>
<td>4.42</td>
<td>4.94</td>
<td>0.54</td>
<td>1.61</td>
</tr>
<tr>
<td>Movement</td>
<td>3.46</td>
<td>2.88</td>
<td>1.95</td>
<td>11.28</td>
</tr>
</tbody>
</table>
Conclusion

- Techniques based on range values don’t reach the same level of precision as state of the art pairwise calibration technique for color images.
- However,
  - they can provide a good approximation in some cases
  - they are easier to set-up
  - movement based calibration permits to detect a misalignment and can offer a temporary calibration when it happens
Thank you for listening,

any questions?