Unordered feature tracking made fast and easy

PIERRE MOULON¹,², PASCAL MONASSE¹

¹Université Paris-Est, LIGM (UMR CNRS), Center for Visual Computing, ENPC, ²Mikros Image. pmo@mikrosimage.eu

THE UNION-FIND SOLUTION

• We see the correspondence fusion problem as set unions,
• It can be solve efficiently using the Union-Find [1] algorithm.

Consider a graph $\mathcal{G}$:
- vertices: features $(\text{ImageId, FeatureId}) =$ sets,
- edges: correspondences $(\text{LeftFeature, RightFeature}) =$ sets union,
- ⇒ Tracks are connected components of $\mathcal{G}$.

Our algorithm enumerates edges (pairwise correspondences) and fuses the sets containing the edges’ endpoints (features).

Algorithm 1 Unordered feature tracking
Input: list of pairwise correspondences
Output: tracks
create a singleton for each feature
for each pairwise match do
  join(leftMatch, rightMatch)
end for
return each connected set as a track

THE TRACK COMPUTATION PROBLEM

The problem of feature point tracking consists in linking multi-view correspondences from pairwise matches that share a common point. It consists of a correspondence fusion problem.

Consider a graph $\mathcal{G}$:
- vertices: features $(\text{ImageId, FeatureId}) =$ sets,
- edges: correspondences $(\text{LeftFeature, RightFeature}) =$ sets union,
- ⇒ Tracks are connected components of $\mathcal{G}$.

Our algorithm enumerates edges (pairwise correspondences) and fuses the sets containing the edges’ endpoints (features).

Algorithm 1 Unordered feature tracking
Input: list of pairwise correspondences
Output: tracks
create a singleton for each feature
for each pairwise match do
  join(leftMatch, rightMatch)
end for
return each connected set as a track

AVAILABLE SOLUTIONS

Available algorithms do not solve the problem in an optimal way:

- Bundler [2] requires a start image index
- ETH-V3D [3] requires many sorting operations
- Bundler depends on image pair order
- ETH-V3D heavy memory consumption

Neither approach is able to identify all valid tracks.

COMPLEXITY & PERFORMANCE

Experimental results on small to large datasets (10 to 2,600 images), largest dataset with more than 41 million pairwise matches, yielding a million tracks.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution time ratio</td>
<td>1.57</td>
<td>2.09</td>
<td>1.00</td>
</tr>
<tr>
<td>Track count</td>
<td>78%</td>
<td>90%</td>
<td>100%</td>
</tr>
<tr>
<td>Lines of code (C++)</td>
<td>(\approx 200)</td>
<td>(\approx 150)</td>
<td>(\approx 100)</td>
</tr>
</tbody>
</table>

\(\alpha(\cdot)\) is the inverse Ackermann function, quasi-constant in practice.

REFERENCES