Matching of interest point groups with pairwise spatial constraints

Ee Sin, Ng and Nick Kingsbury
email: esn21@cam.ac.uk and ngk10@cam.ac.uk

UNIVERSITY OF CAMBRIDGE
Signal Processing Group
Dept. of Engineering
Scope

Introduction

Background

Proposed algorithm
  Pairwise spatial constraints
  Matching algorithm

Experimental results

Conclusions
Introduction

Why are accurate correspondences important?

- Accurate correspondences are required in various computer vision tasks (e.g. detection, classification)
- Performance of these algorithms degrades under various conditions (e.g. occlusion, viewpoint change)
- We focus on the use of interest points (e.g. DoG) and descriptors (e.g. SIFT) here to establish correspondences

1. Lowe, *Distinctive image features from scale-invariant keypoints*, IJCV 2004
Matching of features using appearance alone is insufficient
We consider the use of spatial information as well
Spatial information used are in the form of pairwise spatial constraints between features
The aim is to use spatial information to produce robust correspondences
Spatial information has been used previously for matching:

- Graph matching \(^2\)
- Optimisation with geometric models \(^3\)
- Spatial pyramids \(^4\)

The proposed algorithm has similarities to techniques based on graph matching and optimisation with geometric models

---


\(^3\) D. Lowe, *Distinctive image features from scale-invariant keypoints*, IJCV 2004

Pairwise relationships between a pair of interest points

- $\varphi_v$ – Orientation of feature at $v$
- $\varphi_u$ – Orientation of feature at $u$
- $\delta_{u,v}$ – Length of line vector
- $\theta_{u,v}$ – Orientation of line vector

**Figure:** Pairwise spatial relationships used for a pair of interest points $u, v$
Pairwise relationships

- Consider interest points $u, v$ in an image $X$:
  \[
  \hat{x} = \delta_{u,v} \exp(j\theta_{u,v})
  \]  

- We define 2 sets of pairwise relationships between $u, v$:
  \[
  A_1(u, v) = \begin{pmatrix}
  \phi_u - \theta_{u,v} \\
  \phi_v - \theta_{u,v}
  \end{pmatrix}
  \]  
  \[
  A_2(u, v) = \begin{pmatrix}
  f_u \\
  f_v
  \end{pmatrix}
  \]

- where $\phi_u$ and $\phi_v$ are feature orientations, $f_u, f_v$ are feature descriptors
Pairwise relationships between a pair of interest points

\( \phi_v - \theta_{u,v} \) and \( \phi_u - \theta_{u,v} \) are spatial relationships collected for a pair of interest points in an image.

They should remain fairly consistent for a pair of matching interest points in another image.

**Figure:** Pairwise spatial relationships used for a pair of interest points \( u, v \)
Pairwise relationships

- Likewise, we consider $p, q$ in an image $Y$:

$$\hat{y} = \delta_{p,q} \exp(j\theta_{p,q})$$ (4)

- We collect the pairwise relationships $A_1(p, q)$ and $A_2(p, q)$ between $p, q$ as defined previously.

- The log-ratio of line vectors $(\ln \frac{\hat{x}}{\hat{y}})$ defines a pairwise relationship between interest point pairs $u, v$ and $p, q$.
Proposed algorithm
Pairwise spatial constraints

Pairwise relationships

\[ \kappa + j\rho = \ln \left( \frac{\delta_{u,v} \exp(j\theta_{u,v})}{\delta_{p,q} \exp(j\theta_{p,q})} \right) \]

\[ = \ln \frac{\delta_{u,v}}{\delta_{p,q}} + j(\theta_{u,v} - \theta_{p,q}) \] (5)

- \( \rho \) - difference in orientation of vectors
- \( \kappa \) - log-ratio of vector lengths
- Scale change (\( \kappa \)) and rotation (\( \rho \)) of interest point pairs

Ng & Kingsbury (University of Cambridge)  Matching with pairwise spatial constraints  ICIP 2010 10 / 33
Matching a pair of interest points \( \{u, v\} \) to \( \{p, q\} \)

\[
\begin{align*}
\phi_v - \theta_{u,v} \quad & \quad \theta_{u,v} \\
\delta_{u,v} \quad & \quad \rho \\
\phi_u - \theta_{u,v} \quad & \quad \theta_{u,v} \\
\phi_q - \theta_{p,q} \quad & \quad \theta_{p,q} \\
\delta_{p,q} \quad & \quad \rho \\
\phi_p - \theta_{p,q} \quad & \quad \theta_{p,q}
\end{align*}
\]

**Figure:** Matching a pair of interest points \( u, v \) to a second pair \( p, q \).
Proposed algorithm
Pairwise spatial constraints

Pairwise spatial matching

- Define a similarity space $S(\kappa, \rho)$
- $S$ measures consistency of orientation and scale change
- Consider matching $u, p$
- Define orientation consistency of interest point pairs as:

$$\chi_{u,p} = \frac{\cos(\phi_u - \theta_{u,v} - \phi_p + \theta_{p,q}) + 1}{2}$$

(6)

- Note: The cos function has a fairly broad maximum
Pairwise spatial matching

- We define the feature similarity of interest point pairs as:
  \[
  \gamma_{u,p} = \exp\left(-\frac{\|f_u - f_p\|^2}{2\sigma^2}\right)
  \tag{7}
  \]

- The similarity of interest point pairs depends on orientation consistency and feature similarity

- Likewise, we define \(\gamma_{v,q}\) and \(\chi_{v,q}\) for \(v, q\)

- The pairwise similarity can then be expressed as:
  \[
  \psi\{(u,p),(v,q)\} = \frac{\chi_{u,p}\gamma_{u,p} + \chi_{v,q}\gamma_{v,q}}{2}
  \tag{8}
  \]

- \(\gamma\) and \(\chi\) measure orientation consistency and feature similarity respectively
Pairwise spatial matching

- The similarity score $\psi\{(u,p),(v,q)\}$ is calculated for pairwise combinations of interest points.
- These scores are collected in $S(\kappa, \rho)$.
- Matches can then be found by searching for peaks in $S$.
- For example, using the maxima of histogram or mean shift mode estimator \(^5\)

---

\(^5\)D. Comaniciu and P. Meer, *Mean shift: A robust approach towards feature space analysis*, PAMI 2002
Proposed algorithm

Pairwise spatial constraints

Figure: An example of the similarity space for matching two images
Proposed algorithm

- Spatial constraints are weak for interest points that are far apart
- Thus, we choose to employ spatial constraints over a local neighbourhood
- Consider adjacent square windows having 50% area overlap in an image
- Windows are chosen to be a certain fraction of image area (typically chosen to be 1/25 of image area)
- We used interest points from the Difference of Gaussians detector, along with SIFT descriptors
Figure: Summary of matching algorithm
Proposed algorithm

Matching algorithm

Consider the interest points in $n_1$ and $m_1$.

The spatial relationships between pairs of interest points are collected.

Form similarity space $S$ between $n_1$ and $m_1$.

Perform search for maxima in $S$.

Store peak score and pairwise matches contributing to the peak.

Figure: Summary of matching algorithm
Consider the next window in \( Y \), and we match \( n_1 \) to \( m_2 \) now.

- Collect spatial relationships and find pairwise matches with corresponding peak score.

This is repeated for all \( m \) in \( Y \).

- To find the window that \( n_1 \) will be matched to, we choose the window \( m \) with highest peak score of all windows in \( Y \).

**Figure:** Summary of matching algorithm
The window in $Y$ with highest peak score will be selected as the accurate match to $n_i$.

- All pairwise matches are found as we iterate through the windows.

**Figure:** Summary of matching algorithm
Reducing the computational cost

- The computational cost is high when we consider all possible pairwise combinations of interest points
- We do not need to consider all possible combinations
- An initial set of matches can be first selected
- Initial matches are selected using the ratio of nearest neighbours threshold
  
  In our tests, we set this threshold to be 0.4, with the unconstrained SIFT initial matches as the baseline
- Spatial relationships are only considered between these initial matches

---

D. Lowe, *Distinctive image features from scale-invariant keypoints*, IJCV 2004
Reducing the computational cost

- In addition, the similarity score $\psi\{(u,p),(v,q)\}$ for each interest point pairs considered must be $> \tau$ to be stored in $S$
- This reduces the number of pairs we consider when collecting $S$
- $\tau$ is typically set to 0.7 here
Experimental results

Evaluation framework

- We compared four algorithms
  - Unconstrained baseline SIFT using ratio of nearest neighbours threshold (*uc-sift*)
  - Spectral matching (*sp-match*) \(^7\)
  - *uc-sift* followed by a Hough transform for fitting the matches to a geometric affine model (*hough-sift*) \(^8\)
  - Proposed algorithm (*pw-match*)

- We adopted an evaluation framework which used epipolar constraints to validate actual correspondences \(^9\)

---

\(^7\) M. Leordeanu and M. Hebert, *A spectral technique for correspondence problems using pairwise constraints*, ICCV 2007

\(^8\) D. Lowe, *Distinctive image features from scale-invariant keypoints*, IJCV 2004

Evaluation framework

- 25 objects were tested.
- Test views of the framework consist of each object being rotated on a turntable at intervals of $5^\circ$, and matched to a ground truth view of the object.
- We repeat the tests 3 times for each object, at ground truth views of $-30^\circ, 0^\circ, 30^\circ$, with viewpoint change of $-45^\circ$ to $45^\circ$ at intervals of $5^\circ$ relative to each ground truth view.
- We compared the average correspondence ratios of the algorithms:

$$\text{correspondence ratio} = \frac{\sum \text{correct matches}}{\sum \text{total matches}}$$

(9)
Experimental results

Figure: Results for viewpoint change
Experimental results

Results

- *pw-match* produced higher correspondence ratios across all viewpoints.
- This is followed by *hough-sift*, which performs better than *sp-match* and *uc-sift*.
- *uc-sift* has the lowest correspondence ratio, since no spatial information is being considered.
The improvement in correspondence ratio for *pw-match* is higher at larger viewpoint changes.

This implies that the use of spatial constraints have produced matches that are more robust to viewpoint change.

The proposed algorithm has approximately 25% higher computational time compared to *uc-sift*.
Results

- We also performed tests on the ZuBud building database
- 15 pairs of building images were selected from the database
- Since the ground truth is unavailable, we labelled the false matches by hand
Experimental results

91 correct matches, 16 false matches
(a) uc-sift

72 correct matches, 3 false matches
(b) pw-match

Figure: Results for ZuBud database
Experimental results

35 correct matches, 13 false matches
(a) uc-sift

16 correct matches, 5 false matches
(b) pw-match

Figure: Results for ZuBud database
Table: Matching results for 15 buildings in ZuBud database

<table>
<thead>
<tr>
<th>Results</th>
<th>uc-sift</th>
<th>sp-match</th>
<th>pw-match</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total matches</td>
<td>2199</td>
<td>2033</td>
<td>1483</td>
</tr>
<tr>
<td>Correct matches</td>
<td>1913</td>
<td>1830</td>
<td>1421</td>
</tr>
<tr>
<td>False matches</td>
<td>286</td>
<td>203</td>
<td>62</td>
</tr>
<tr>
<td>Correspondence ratio</td>
<td>0.870</td>
<td>0.900</td>
<td>0.958</td>
</tr>
</tbody>
</table>
The proposed algorithm can account for the structure and layout of features by using pairwise constraints.

The pairwise similarity of interest point pairs are defined based on orientation consistency and feature similarity.

Our experiments suggest that the matching algorithm produces robust matches even under large changes in viewpoints.

Future work: Extension to classification and detection of objects with matches produced.
Thank you
http://www-sigproc.eng.cam.ac.uk/~esn21