Video Google: A Text Retrieval Approach to Object Matching in Videos

Josef Sivic, Frederik Schaffalitzky, Andrew Zisserman

Visual Geometry Group
University of Oxford
The vision …

Enable video, e.g. a feature length film, to be searched on its visual content with the same ease and success as a Google search of text documents.

“Run Lola Run” (‘Lola Rennt’) [Tykwer, 1999]

“Groundhog Day” [Rammis, 1993]
Visually defined search

Given an object specified in one frame, retrieve all shots containing the object:

- must handle viewpoint change
- must be efficient at run time

Example: Groundhog Day
Example: Groundhog Day

73 keyframes retrieved
53 correct, first incorrect ranked 27

Rank: 12 35 50 69
Example: Run Lola Run

query region

close-up

Ranked returned key-frames (selection)

rank 9

rank 16

rank 22

rank 25
Outline

1. Viewpoint invariant content based image retrieval
   • match key frames
   • e.g. Groundhog Day: 170K frames, 1K shots, 5K key frames

2. Benefits of using video over individual images
   • use information from all frames

3. Lessons from text retrieval
   • represent frames by a set of ‘visual words’

4. Extensions
   • external objects
   • object aspects
1. Viewpoint invariant content based image retrieval
Problem statement

Retrieve key frames containing the same object

Various key frames from ‘Run Lola Run’
Approach

• Determine regions (segmentation) and vector descriptors in each frame which are invariant to camera viewpoint changes

• Match descriptors between frames using invariant vectors
Invariance requirements

• A significant translation results in differing surface foreshortening (also scale changes)

• also differing intensity
Local invariance requirements

• Geometric: 2D affine transformation

\[
\begin{pmatrix}
  x' \\
  y'
\end{pmatrix} = \begin{bmatrix}
  A \\
\end{bmatrix} \begin{pmatrix}
  x \\
  y
\end{pmatrix} + \begin{pmatrix}
  t_x \\
  t_y
\end{pmatrix}
\]

where A is a 2x2 non-singular matrix

• Photometric: 1D affine transformation

\[ I_1(x) = aI_2(x) + b \]

• Objective: compute image descriptors invariant to this class of transformations
Viewpoint covariant segmentation

- **Characteristic scales (size of region)**
  - Lindeberg and Garding ECCV 1994
  - Lowe ICCV 1999
  - Mikolajczyk and Schmid ICCV 2001

- **Affine covariance (shape of region)**
  - Baumberg CVPR 2000
  - Matas et al BMVC 2002
  - Mikolajczyk and Schmid ECCV 2002
  - Schaffalitzky and Zisserman ECCV 2002
  - Tuytelaars and Van Gool BMVC 2000

- **Miscellaneous others**
  - Tell and Carlsson ECCV 2000
Covariant regions, type I

Shape adapted interest point neighbourhoods – cover same scene region

1. Detect interest points
2. Determine scale using extrema of Laplacian of Gaussian
3. Determine elliptical shape using stationary condition on second moment gradient matrix

See Mikolajczyk & Schmid, and Schaffalitzky & Zisserman ECCV 2002
Covariant regions, type II:

Maximally Stable regions (MSR)

1. Segment using watershed algorithm, and track connected components as threshold value varies.

2. An MSR is detected when the area of the component is stationary

See Matas et al BMVC 2002
Example: Maximally stable regions
Example of covariant regions

1000+ descriptors per frame

- Shape adapted regions
- Maximally stable regions
Viewpoint invariant description

• Elliptical viewpoint covariant regions
  • Shape Adapted regions
  • Maximally Stable Regions

• Map ellipse to circle and orientate by dominant direction

• Represent each region by SIFT descriptor (128-vector) [Lowe 1999]
  • see Mikolajczyk and Schmid CVPR 2003 for a comparison
Return to Problem statement

Retrieve key frames containing the same object

Various key frames from ‘Run Lola Run’
Pros and cons of a set of viewpoint invariant descriptors

**Advantage:**
- local descriptors are robust to partial occlusion
- global descriptors e.g. colour histograms and texture histograms, are not

**Disadvantage:**
- too much individual invariance
- each region can rotate independently (by different amounts)
- use semi-local and global spatial relations to verify matches
Approach

• Determine regions (segmentation) and vector descriptors in each frame which are invariant to camera viewpoint changes

• Match descriptors between frames using invariant vectors

• Verify / reject frame matches using spatial consistency and multiple view geometric relations:
  • spatial neighbours match
  • homographies
  • epipolar geometry

• Frames are matched if a sufficient number of regions satisfy the geometric relations between views
Example

512 x 768 pixel images
Goal: establish matches
A sub-set of the shape adapted interest points

2300 interest points reduced to one third (770) after adaptation
Stage (1): invariant indexing

Match closest vectors

• all vectors within a ball in invariant space
Stage (2): neighbourhood consensus

Loose constraint on **spatial** consistency for a putative match

- compute the $K$ closest matched neighbours of the point in each image
- require that at least $N$ of these must match

Here $K = 10$, $N = 1$
Stricter photometric and geometric filters can be applied

e.g. 1: local verification
Matched invariants do not imply that the regions match
  • regions match if cross-correlation after registration is below threshold

e.g. 2: local geometry
• neighbouring correspondences consistent with the affine transformation

![Diagram showing image 1 and image 2 with corresponding points indicated by lines and dots.](image)
Detail

key-frames

matched points

shot 4

shot 9
pairs

Stage (1)
invariant indexing

Stage (2)
neighbourhood consensus

rank retrieved key frames by number of matches
2. Video: the benefits of having contiguous frames
Track regions through shot …

Constant velocity dynamical model and correlation

- Shape adapted regions
- Maximally stable regions
The benefits of contiguous frames within a shot …

Track regions throughout a shot and aggregate information:

1. Reject unstable tracks (threshold on track length)
   • Increases discrimination of descriptors

2. Compute mean descriptor over track
   • Increases accuracy (signal to noise) of descriptor

3. Compute noise covariance over tracks
   • Defines Mahalanobis distance for descriptor space
One region detail

- Full frame
- Benefit of SIFT
- Affine normalized region
- Region close-up
A region tracked over 70 frames

First 8 dimensions (columnwise) of the SIFT descriptor evolving with time.

Close-ups of sample frames
3. Lessons from text retrieval –
google like object retrieval
Text retrieval overview

Stemming

Represent words by stems, e.g. “walking”, “walks” → “walk”

Stop-list

Reject the very common words, e.g. “the”, “a”, “of”

Inverted file

<table>
<thead>
<tr>
<th>d1</th>
<th>d2</th>
<th>d3</th>
<th>d4</th>
</tr>
</thead>
<tbody>
<tr>
<td>common</td>
<td>people</td>
<td>sculpture</td>
<td>common</td>
</tr>
<tr>
<td>people</td>
<td></td>
<td>sculpture</td>
<td>common</td>
</tr>
<tr>
<td>common</td>
<td></td>
<td></td>
<td>people</td>
</tr>
<tr>
<td>people</td>
<td></td>
<td></td>
<td>common</td>
</tr>
</tbody>
</table>

Ideal book index:

<table>
<thead>
<tr>
<th>Term</th>
<th>List of hits (occurrences in documents)</th>
</tr>
</thead>
<tbody>
<tr>
<td>People</td>
<td>[d1:hit hit hit], [d4:hit hit hit] …</td>
</tr>
<tr>
<td>Common</td>
<td>[d1:hit hit], [d3: hit hit], [d4: hit hit hit] …</td>
</tr>
<tr>
<td>Sculpture</td>
<td>[d2:hit], [d3: hit hit hit] …</td>
</tr>
</tbody>
</table>

- word matches are pre-computed
Ranking

- frequency of words in document (tf-idf)
- proximity weighting (google)
- PageRank (google)
Building a visual vocabulary

Vector quantize descriptors

- k-means clustering with Mahalanobis distance

Implementation

- use 48 shots from Lola = 200K track descriptors (means)
- 6K clusters for Shape Adapted regions
- 10K clusters for Maximally Stable regions
Why cluster separately?

- the two types of regions cover different and independent scene regions
- they may be thought of as different vocabularies for describing the scene
Samples of visual words:

Shape adapted regions

Maximally stable regions
Matching on visual words

- No loss of matching accuracy against standard nearest neighbour matching
- Use same visual vocabulary for matching frames outside training shots in Lola i.e. for unseen shots
- Use same visual vocabulary for matching within Groundhog Day
Video representation

• in advance: represent all key frames by visual words

assign visual words

• video is represented as a (weighted) matrix of occurrences

visual word list

key frames

store frequency and position of words
Matching stages for query region

Stage 1: match to key frames based on the visual word histogram

visual word
histogram for
query region

visual word
list

key frames

score kf 1

score kf 2
Stage 2: spatial consistency

- Discard mismatches
  - require spatial agreement with the neighbouring matches

- Compute matching score
  - score each match with the number of agreement matches
  - accumulate the score from all matches

NB very weak measure of spatial consistency
Example

query region
Visual indexing using text retrieval methods

Initial matches:
- on common visual words

Stop-list:
- Stop top 5% (large clusters of over 3K points) and bottom 10%

Spatial consistency ranking on 15 nearest neighbours:
- reject matches with no support
- rank frame by number of supporting matches
Example: Run Lola Run

20 keyframes retrieved
all correct
Appraisal

• matches on visual words are “pre-computed”, so at run-time retrieval is immediate (0.1s on a 2Gh machine)

• can search for objects and combinations of objects that were not considered when matches pre-computed

• only failures (against ground truth) are when descriptors are missing: e.g. motion or optical blur
4. Extensions
Extension I: searching from other sources

So far query frame chosen from within video ….

Extend to use external images to search for particular objects or places

e.g. for product placement or company logos or particular type of vehicle or building
Retrieve shots from Lola and Groundhog Day
Retrieved shots in groundhog day for search on Sony logo
Extension II: Object level grouping

The problem: searching on one object visual aspect will not return other aspects

A solution: use motion within a shot to group aspects
Motion based grouping example

Input shot

- Track detected regions through shot
- Group tracks into independently moving objects using rigidity
- Use tracks for each object to associate visual aspects
Off the shelf tracking

Tracks repaired using inter-frame matching
  • jump short gaps
  • fill in missed regions

only tracks longer than 10 frames shown
Grouping using robust affine factorization

3 largest groups
Object level matches

Extend from image based retrieval to accessing an object model defined over multiple images.
Object level matches – van example

Query frame with outlined query region

Examples of retrieved frames
Futures – research issues

• Segmentation that commutes with viewpoint
  • have seen a few examples here, more are required to cope with varying scene structure

• Deformable/articulated objects – Luc Van Gool’s talk

• Import further ideas from text retrieval, e.g. latent semantic indexing to find visual content

• Further reading: papers at ICCV 03 and ECCV 04