# Correspondence and image matching

Josef Sivic

Czech Institute of Informatics, Robotics and Cybernetics,

Czech Technical University in Prague

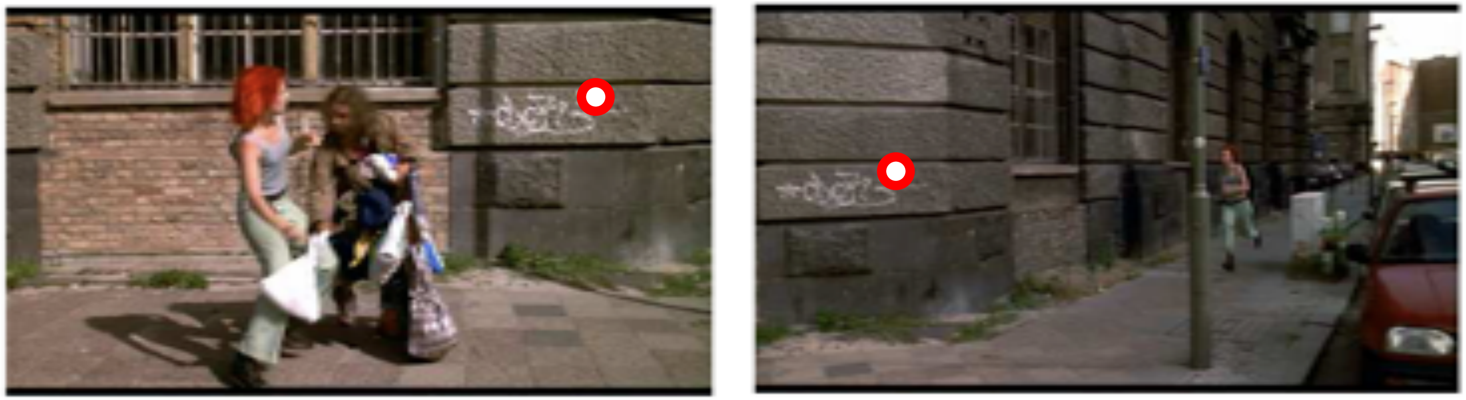
INRIA, WILLOW, ENS/INRIA/CNRS UMR 8548

Departement d'Informatique, Ecole Normale Supérieure, Paris

With slides from: O. Chum, K. Grauman, **S. Lazebnik**, B. Leibe, D. Lowe, J. Philbin, J. Ponce, D. Nister, C. Schmid, N. Snavely, A. Zisserman

# Image matching and recognition with local features

The goal: establish correspondence between two or more images



$$\mathrm{x} = \mathrm{PX} \qquad \mathrm{x}' = \mathrm{P}'\mathrm{X}$$

P : $3 \times 4$ matrix

X : 4-vector

x : 3-vector

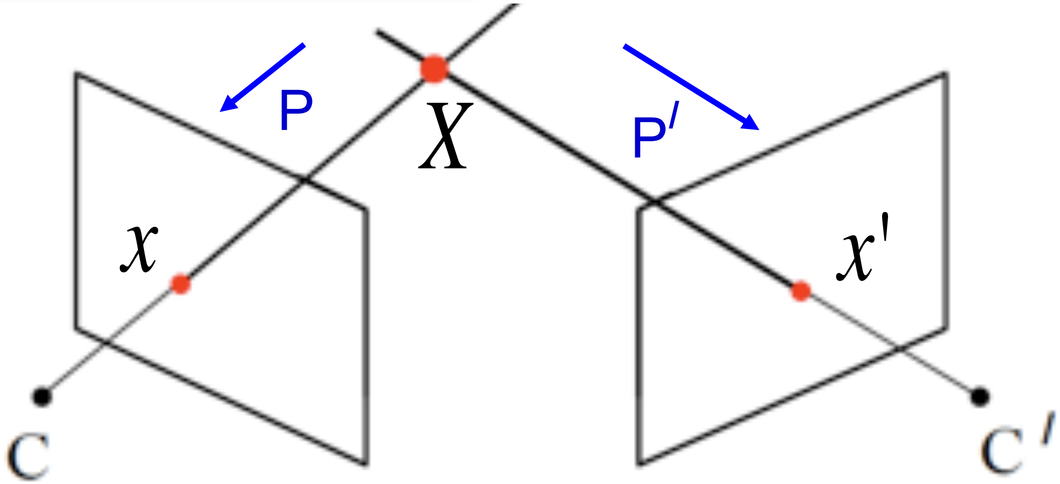



Image points x and x' are in correspondence if they are projections of the same 3D scene point X.

Images courtesy A. Zisserman

Example I: Wide baseline matching and 3D reconstruction

Establish correspondence between two (or more) images.

[Schaffalitzky and Zisserman ECCV 2002]

Example I: Wide baseline matching and 3D reconstruction

Establish correspondence between two (or more) images.


[Schaffalitzky and Zisserman ECCV 2002]

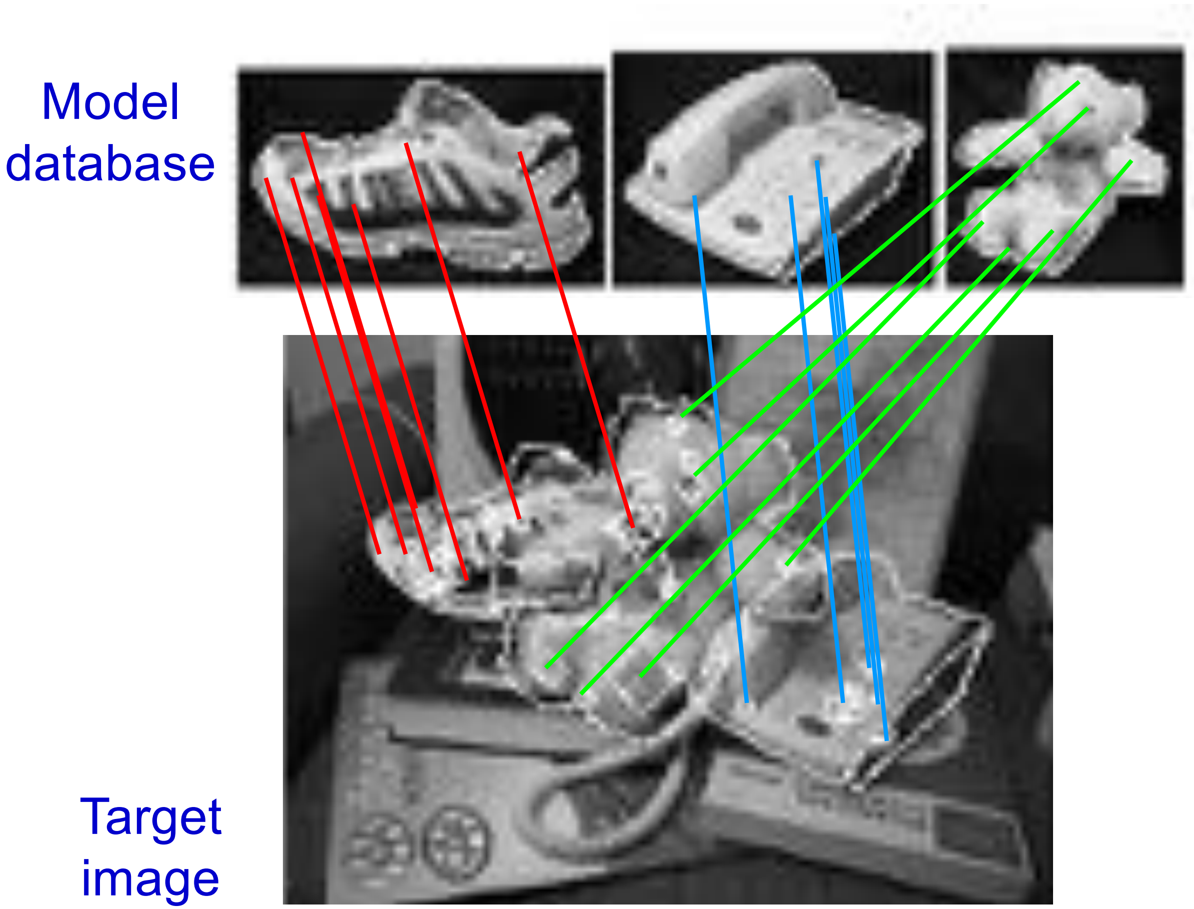
# [Agarwal, Snavely, Simon, Seitz, Szeliski, ICCV'09] – Building Rome in a Day

57,845 downloaded images, 11,868 registered images. This video: 4,619 images.

# 3D reconstruction – capturing reality

# Example II: Object recognition

Establish correspondence between the target image and (multiple) images in the model database.



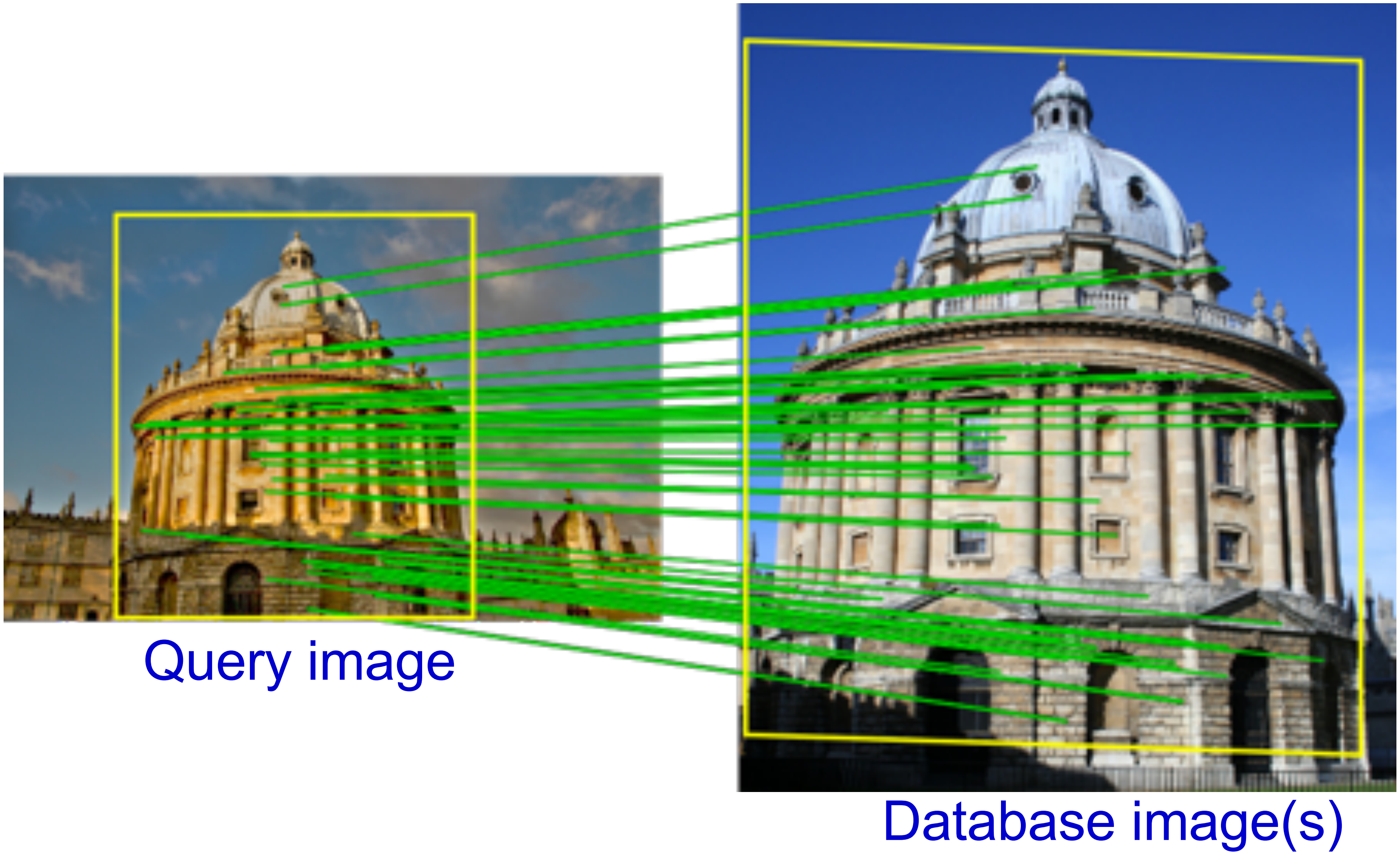

[D. Lowe, 1999]

# Example III: Visual search

Given a query image, find images depicting the same place / object in a large unordered image collection.

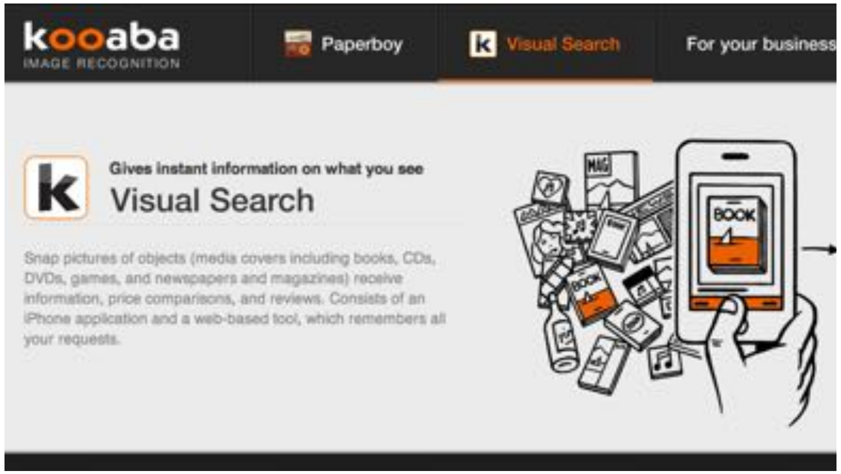
Find these landmarks ...in these images and 1M more

Establish correspondence between the query image and all images from the database depicting the same object / scene.

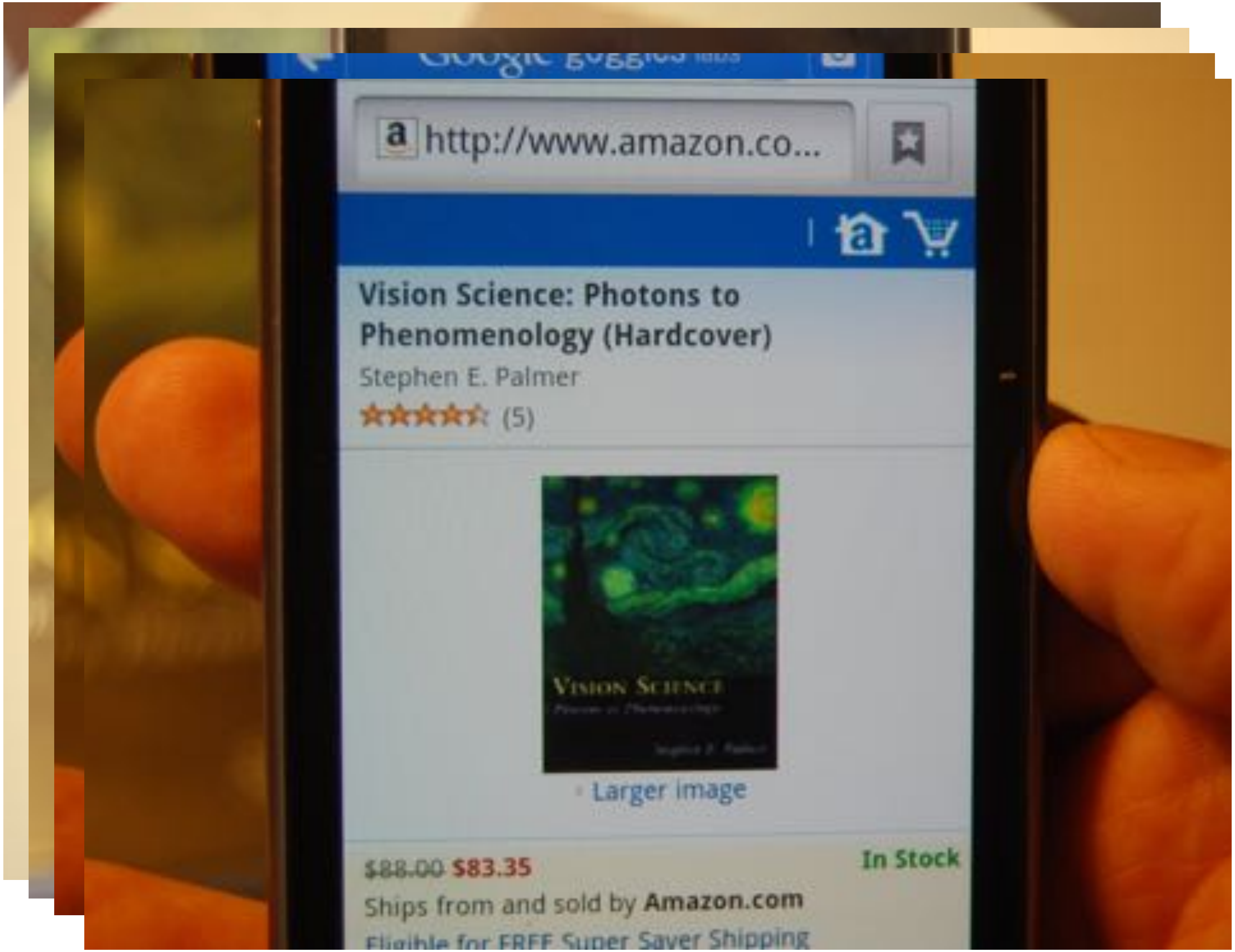


# Mobile visual search

Bing visual scan


# Example



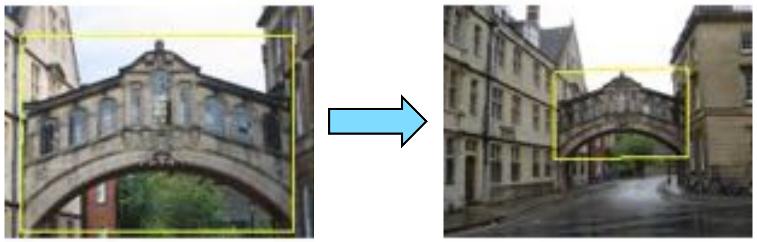

Slide credit: I. Laptev

# Visual navigation for autonomous robotics

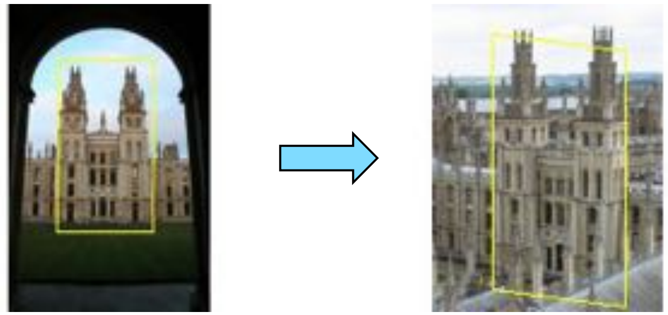

http://mrg.robots.ox.ac.uk/theme/localisation/

# Why is it difficult?

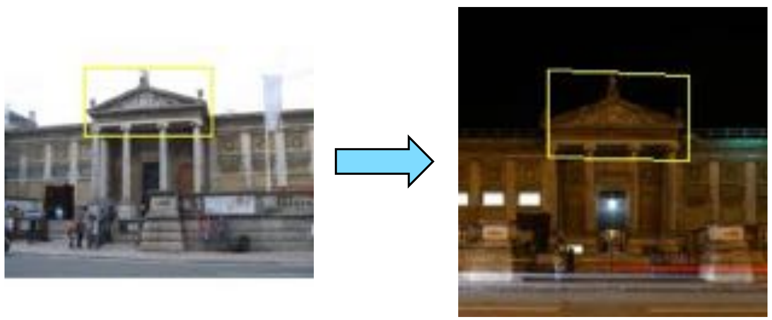
Want to establish correspondence despite possibly large changes in scale, viewpoint, lighting and partial occlusion



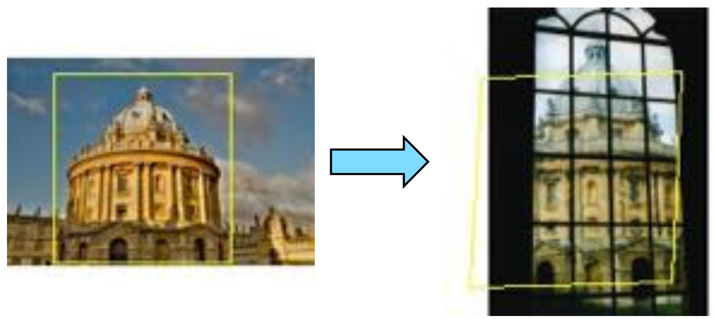
Scale



Viewpoint



Lighting



Occlusion

… and the image collection can be very large (e.g. 1M images)

# Approach

0. **Pre-processing:**
   - Detect local features.
   - Extract descriptor for each feature.

1. **Matching:** Establish tentative (putative) correspondences based on local appearance of individual features (their descriptors).

2. **Verification:** Verify matches based on semi-local / global geometric relations.

3. **Learnable representations** for visual correspondence

# Outline: feature detection

Edges

Corners

**Blobs**

Contours

Regions

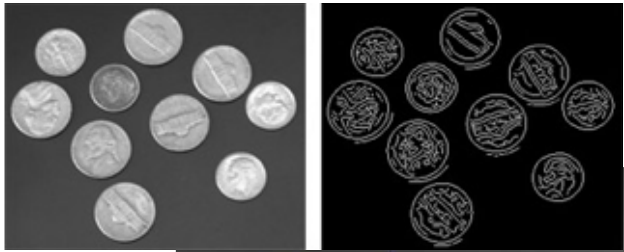


Image regions [Felzenszwalb et al., 2014]

Contours/lines

*Mi-points, angles*

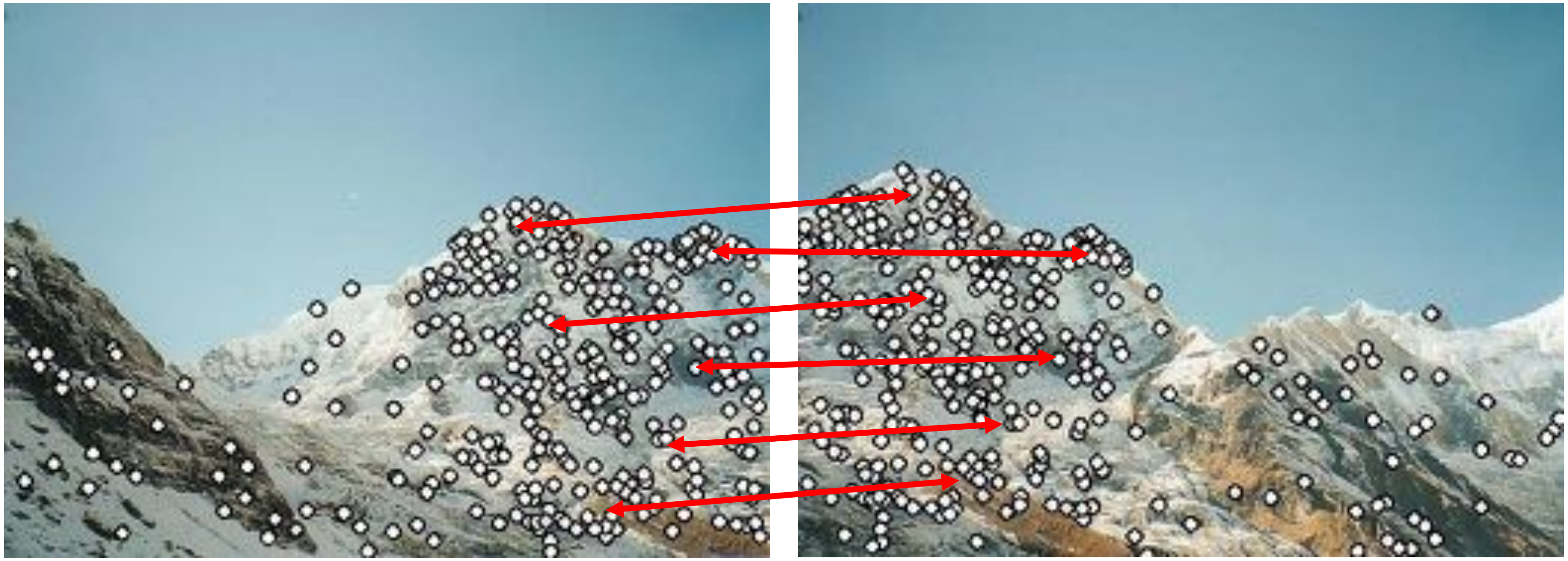
# Why extract features?

- Motivation: panorama stitching
  - We have two images – how do we combine them?

Slide: S. Lazebnik

# Why extract features?

- Motivation: panorama stitching
  - We have two images – how do we combine them?



Step 1: extract features

Step 2: match features

Slide: S. Lazebnik

# Why extract features?

- Motivation: panorama stitching
  - We have two images – how do we combine them?

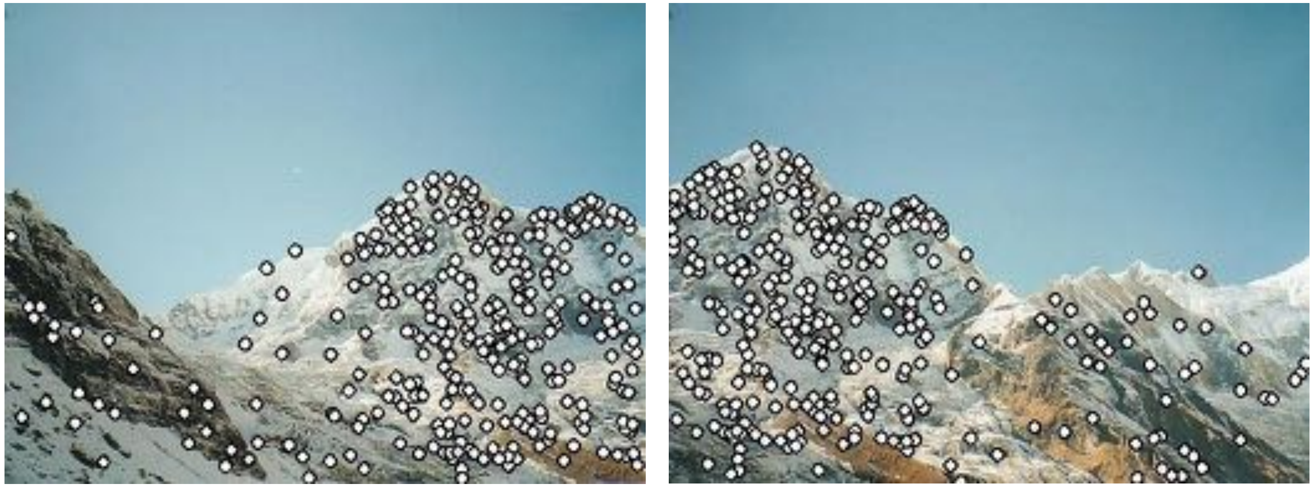
Step 1: extract features

Step 2: match features

Step 3: align images

Slide: S. Lazebnik

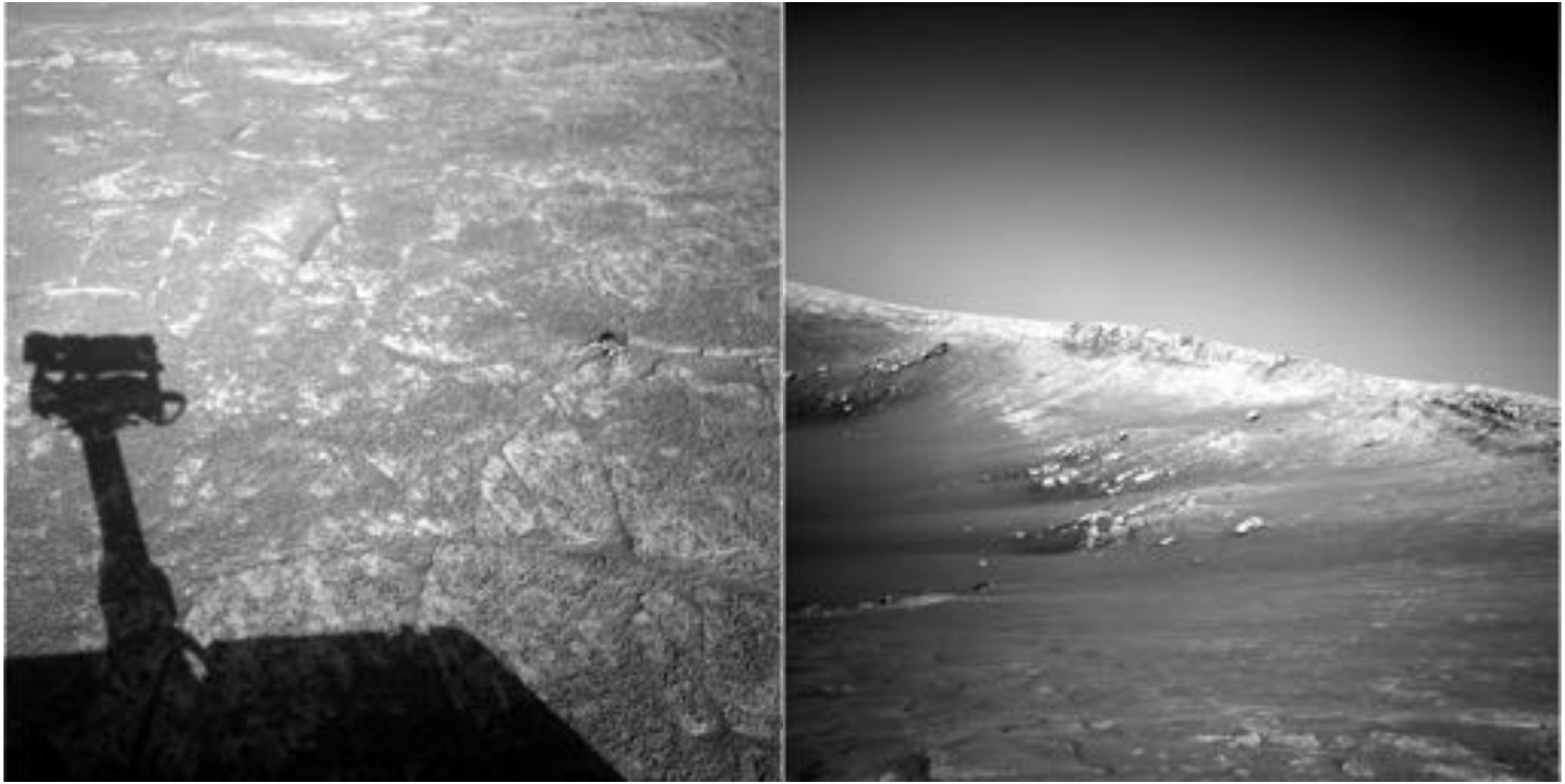
# Characteristics of good features



- Repeatability
  - The same feature can be found in several images despite geometric and photometric transformations
- Saliency
  - Each feature is distinctive
- Compactness and efficiency
  - Many fewer features than image pixels
- Locality
  - A feature occupies a relatively small area of the image; robust to clutter and occlusion

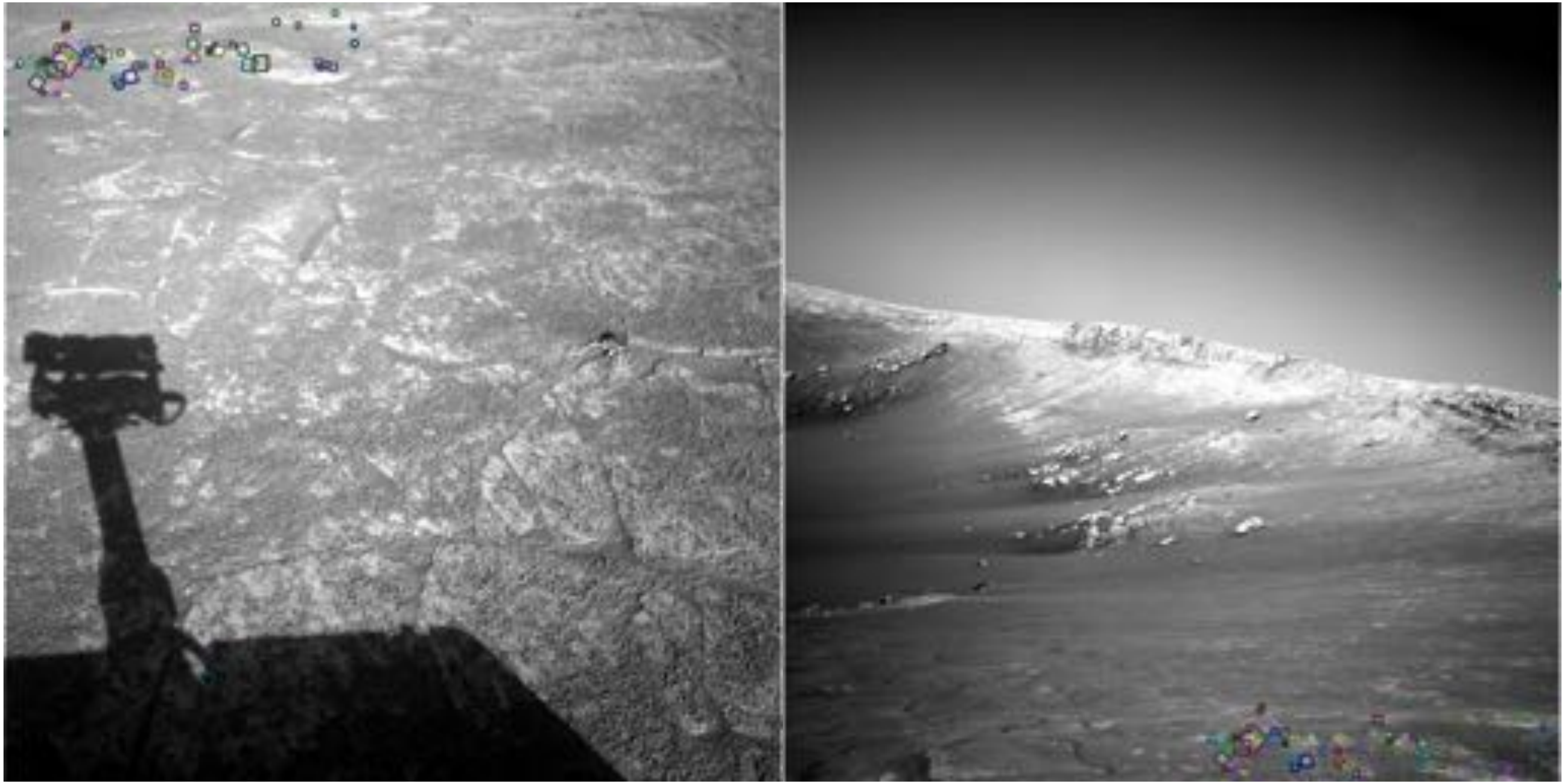
Slide: S. Lazebnik

# A hard feature matching problem



NASA Mars Rover images

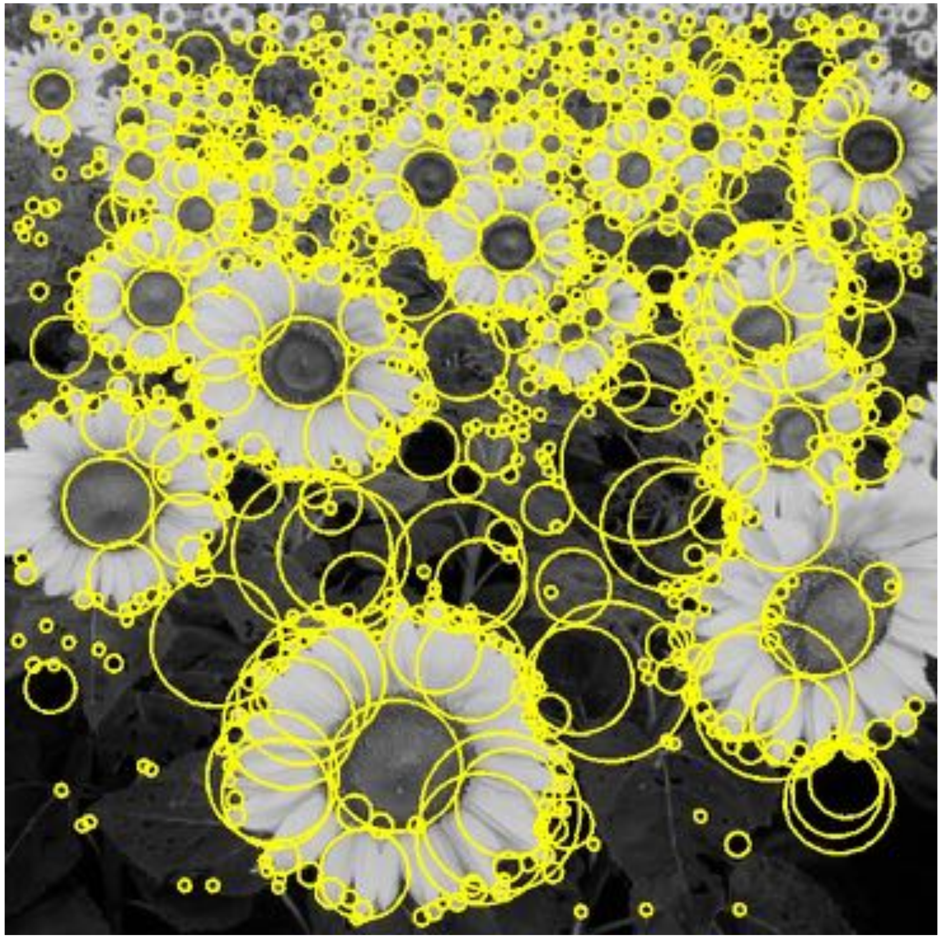
# Answer below (look for tiny colored squares…)



NASA Mars Rover images
with SIFT feature matches
Figure by Noah Snavely

Slide: S. Lazebnik

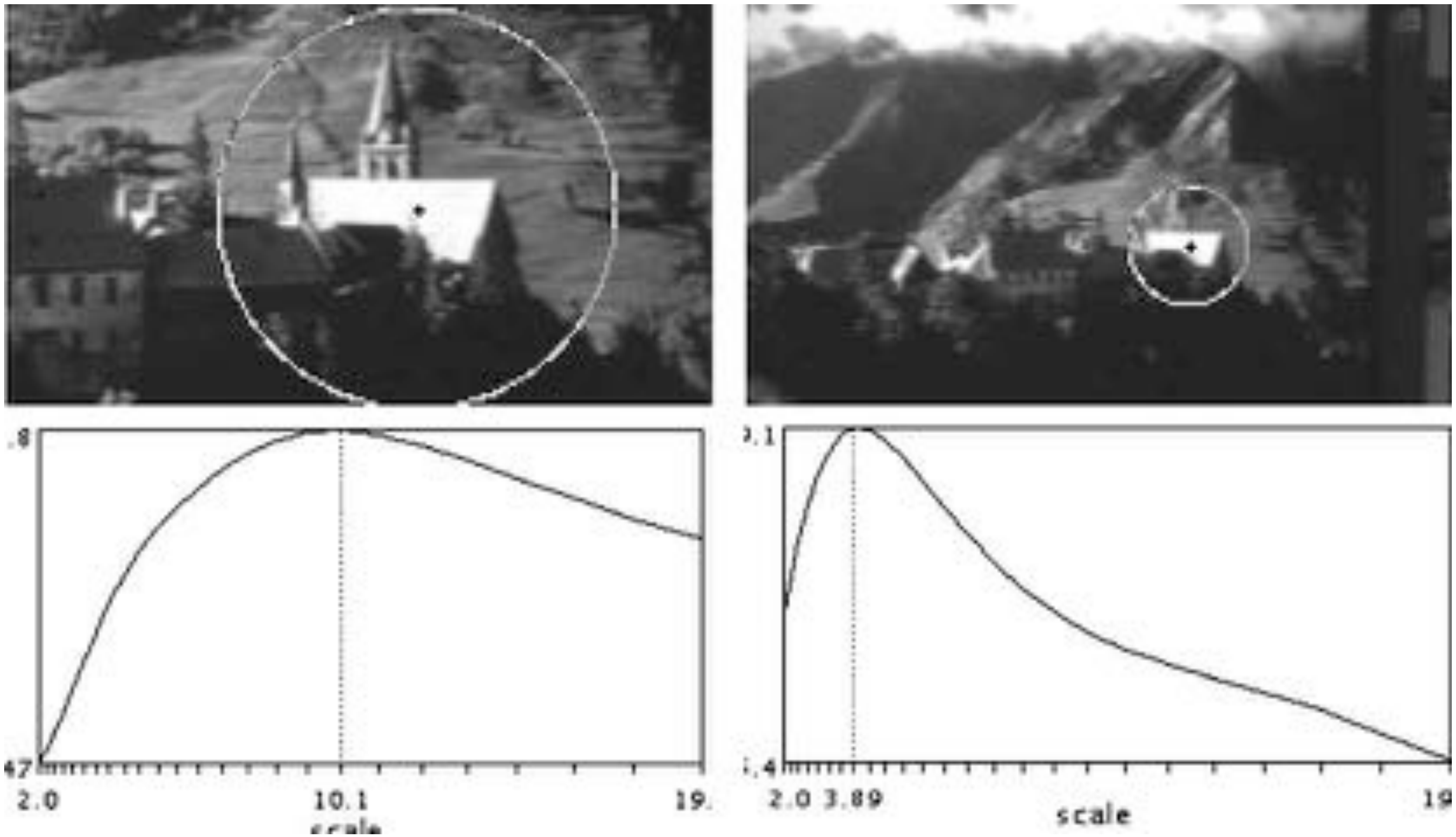
# Blob detection



Slide: S. Lazebnik

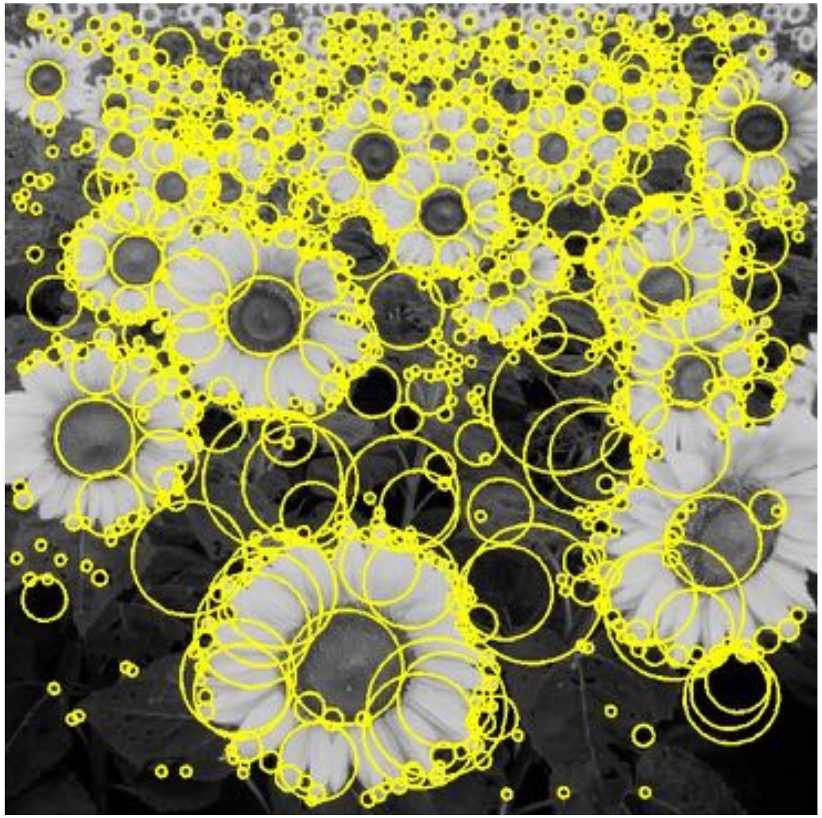
# Feature detection with scale selection

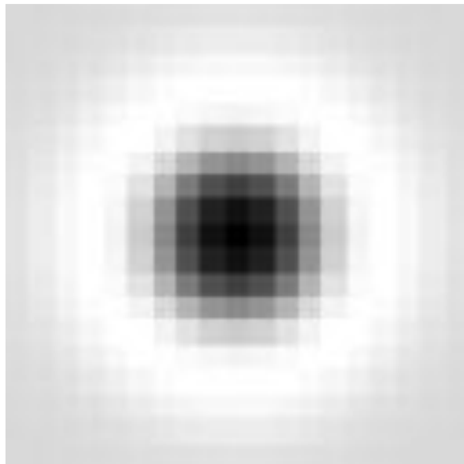
- We want to extract features with characteristic scale that is *covariant* with the image transformation



# Blob detection: basic idea

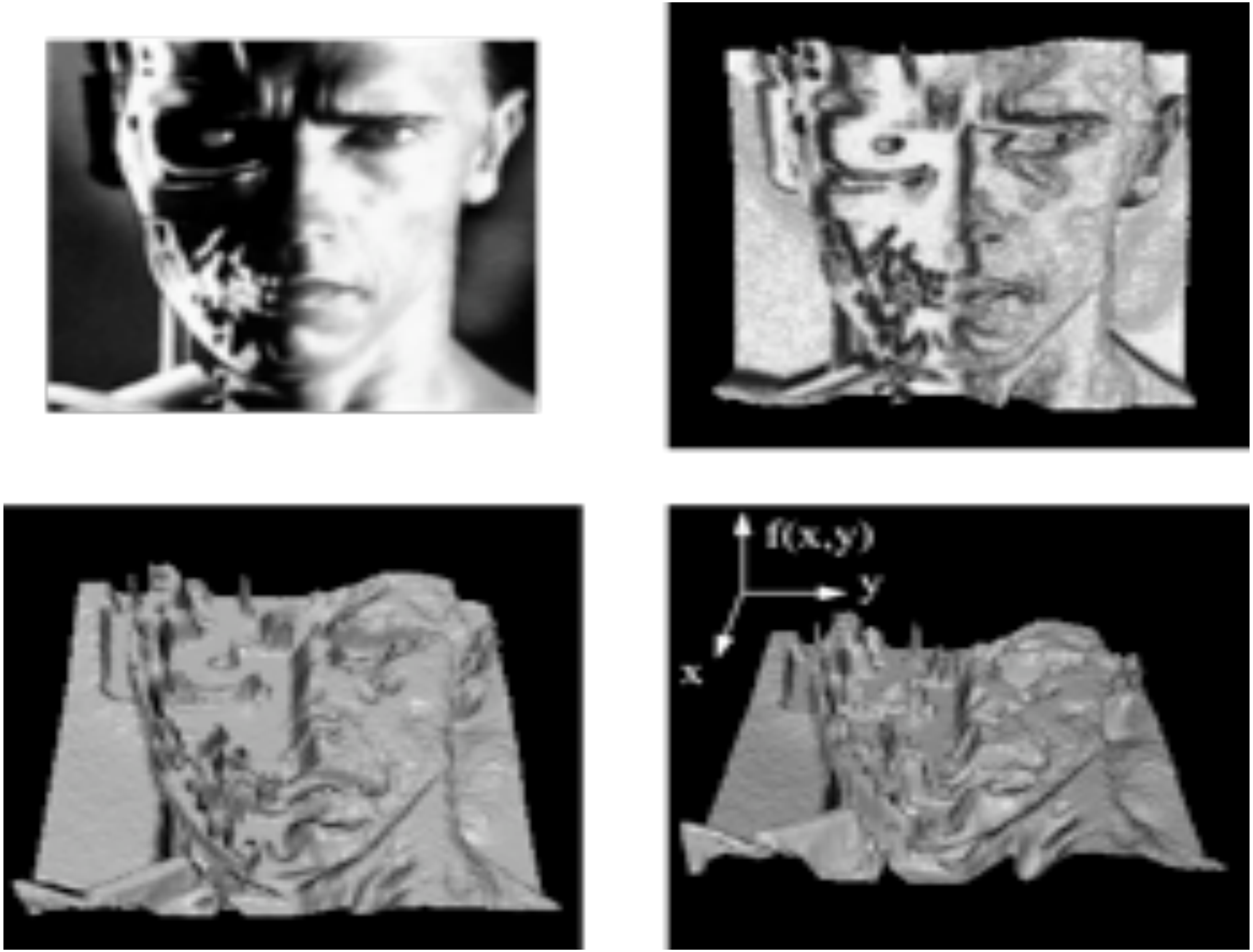
- To detect blobs, convolve the image with a "blob filter" at multiple scales and look for maxima of filter response in the resulting *scale space*





Slide: S. Lazebnik

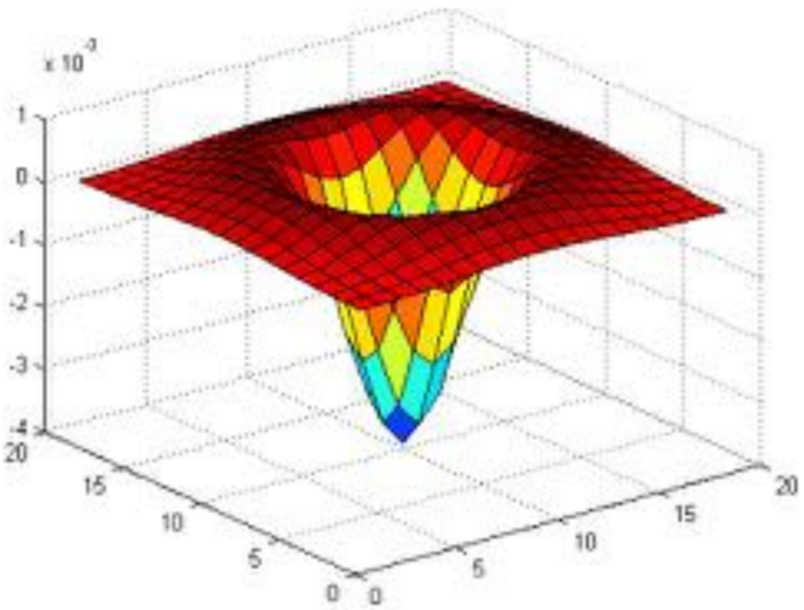
# Images as functions

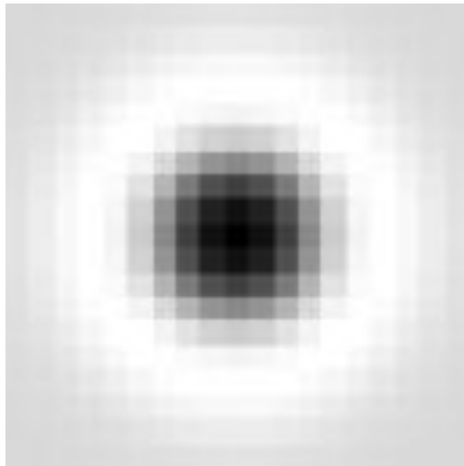



Source: S. Seitz

# Blob filter

Laplacian of Gaussian: Circularly symmetric operator for blob detection in 2D

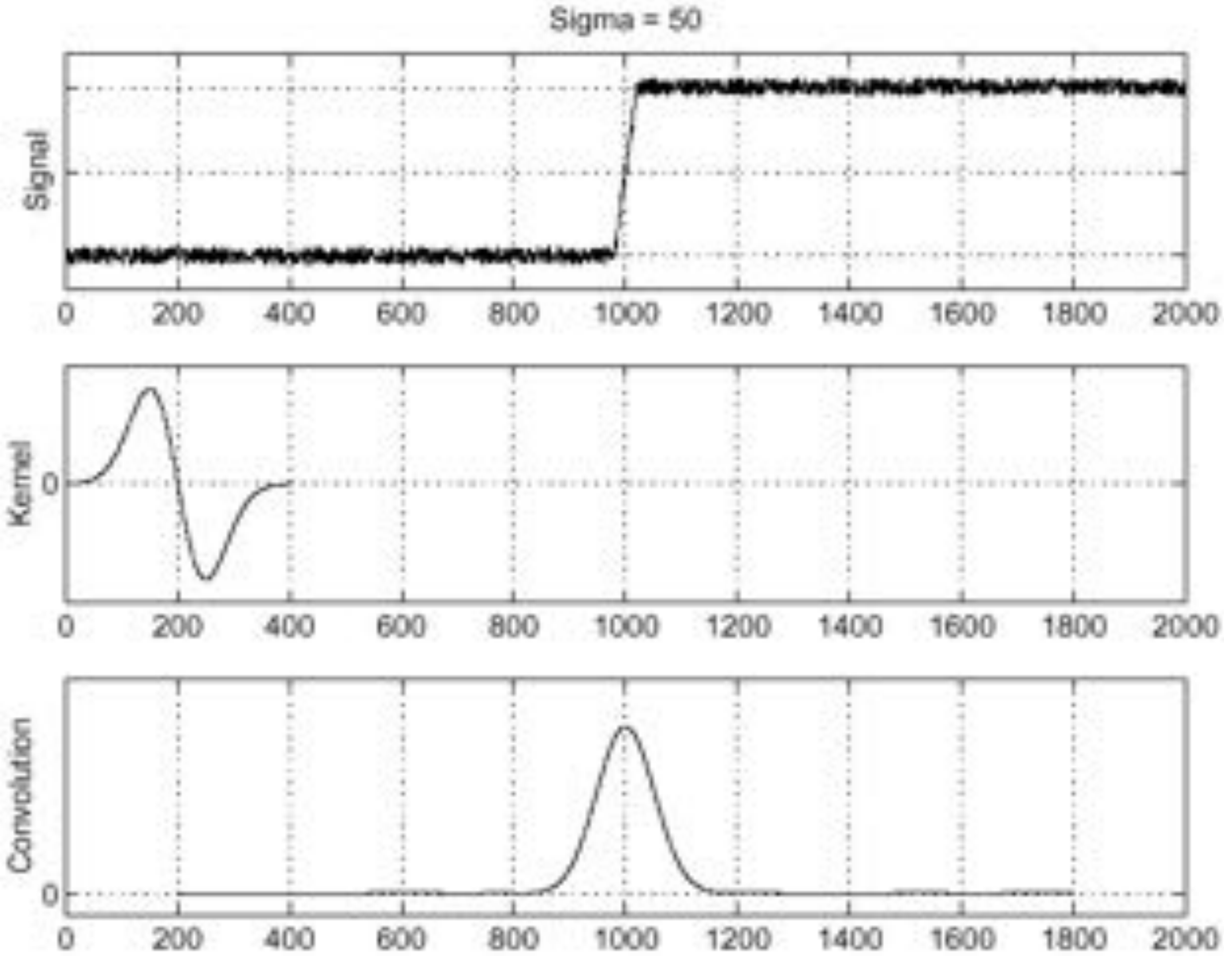


$$\nabla^2 g = \frac{\partial^2 g}{\partial x^2} + \frac{\partial^2 g}{\partial y^2}$$

Slide: S. Lazebnik

# Recall: Edge detection




$f$ — Edge

$\frac{d}{dx} g$ — Derivative of Gaussian

$f * \frac{d}{dx} g$ — Edge = maximum of derivative

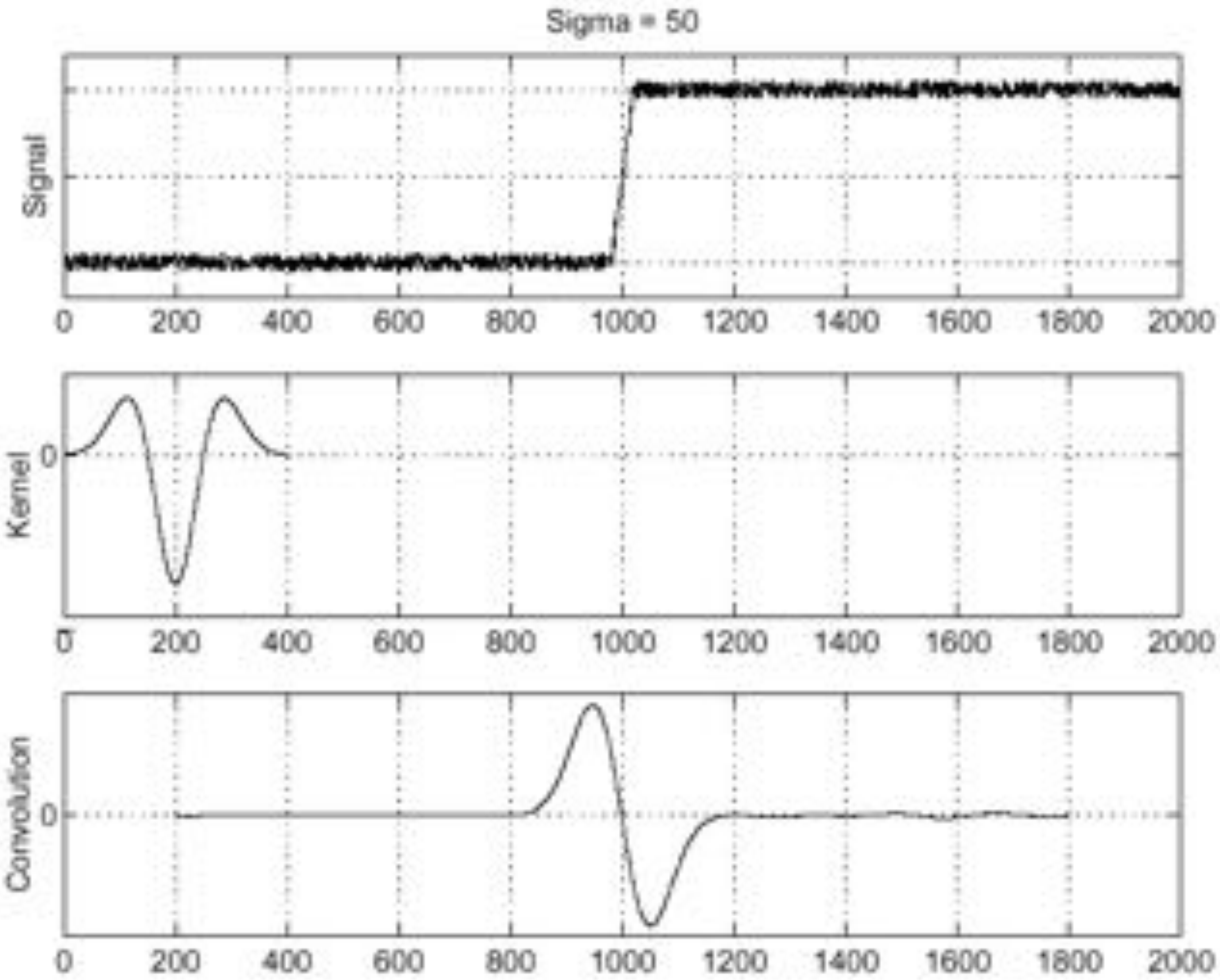
Source: S. Seitz

# Edge detection, Take 2

$f$

$$\frac{d^2}{dx^2} g$$

$$f * \frac{d^2}{dx^2} g$$




Edge

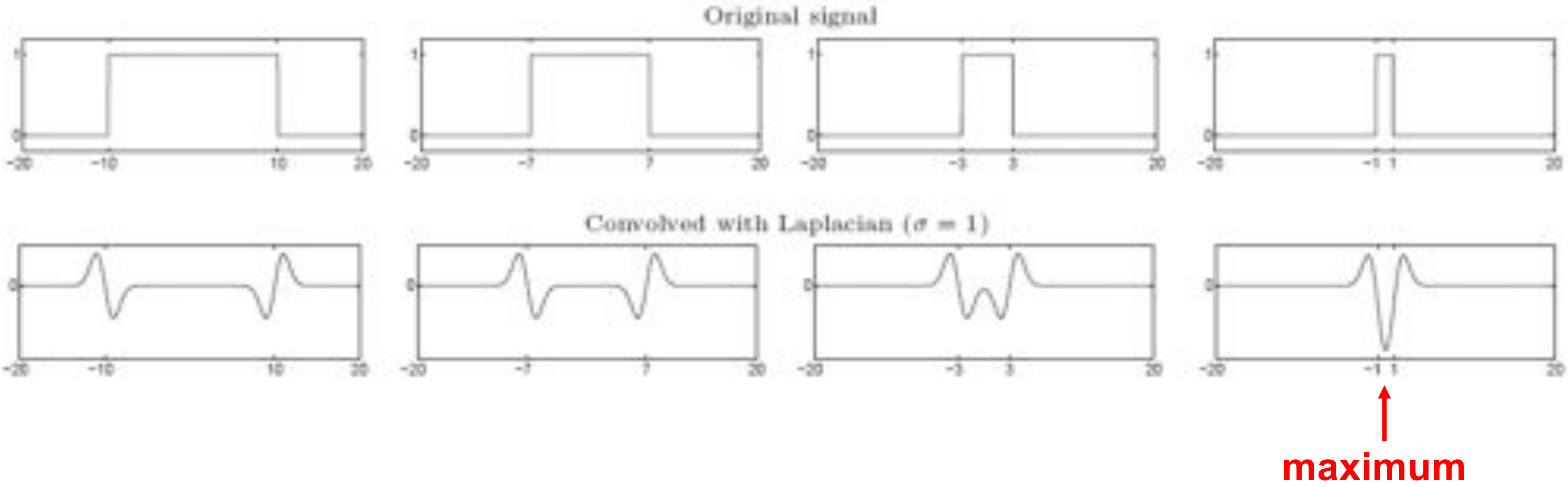
Second derivative of Gaussian (Laplacian)

Edge = zero crossing of second derivative

Source: S. Seitz

# From edges to blobs

- Edge = ripple
- Blob = superposition of two ripples




**Spatial selection**: the magnitude of the Laplacian response will achieve a maximum at the center of the blob, provided the scale of the Laplacian is "matched" to the scale of the blob

Slide: S. Lazebnik

# Scale-space blob detector: Example

Slide: S. Lazebnik

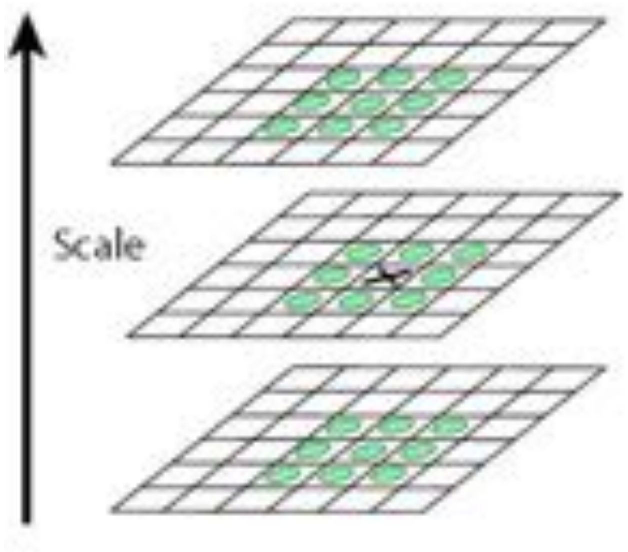
# Scale-space blob detector: Example

sigma = 11.9912

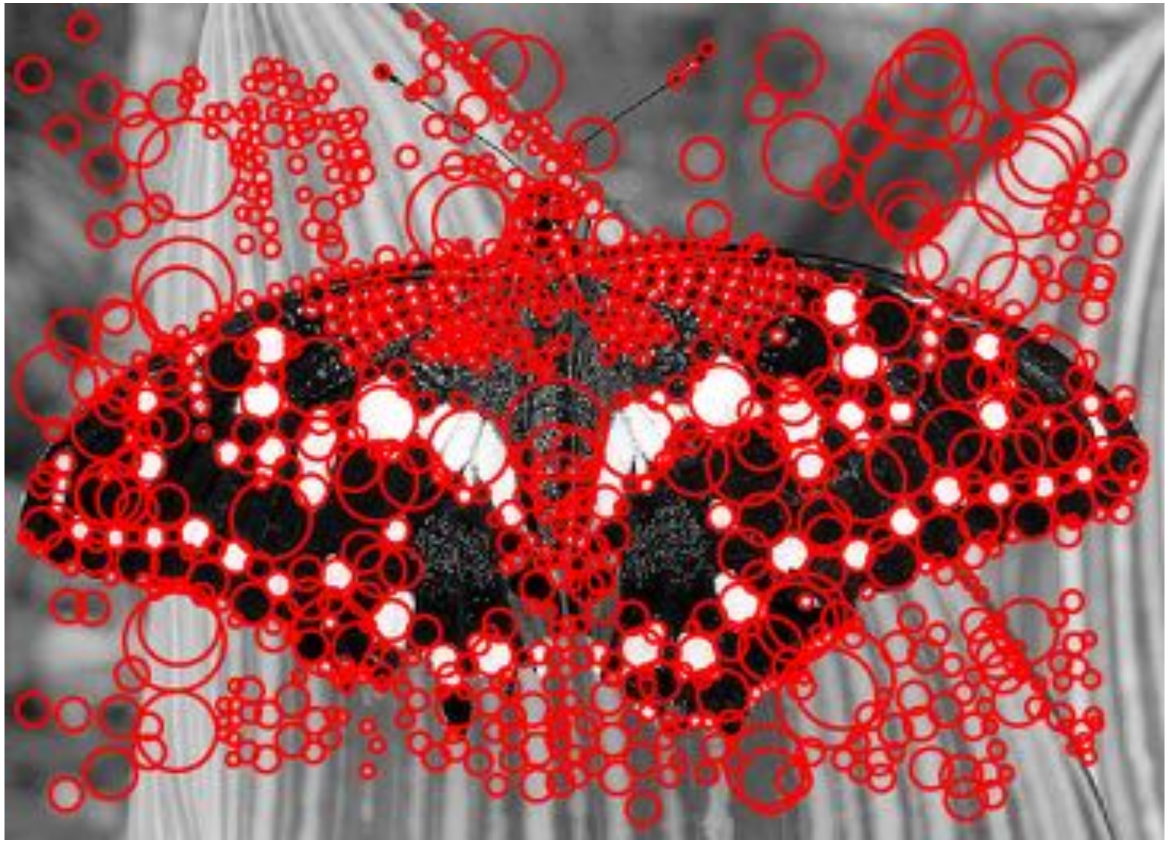
Slide: S. Lazebnik

# Scale-space blob detector

1. Convolve image with scale-normalized Laplacian at several scales
2. Find maxima of squared Laplacian response in scale-space



# Scale-space blob detector: Example

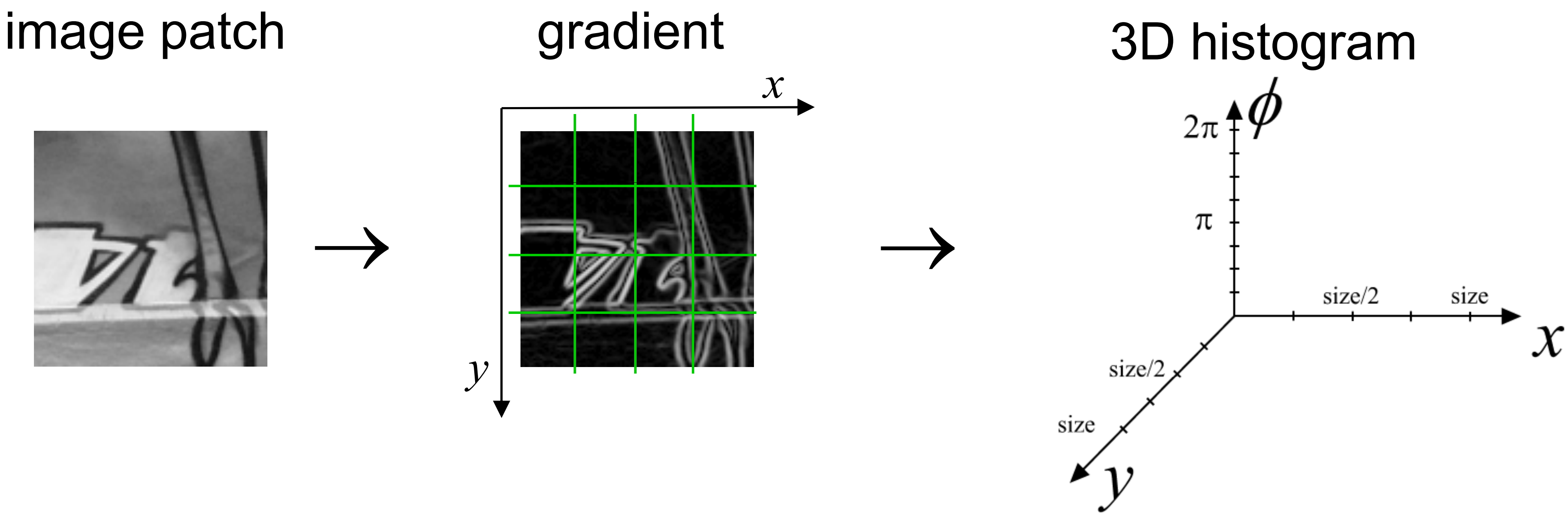


Slide: S. Lazebnik

# SIFT descriptors

4x4 spatial grid, 8 bins for gradient orientation

$\Rightarrow$ dimension 128

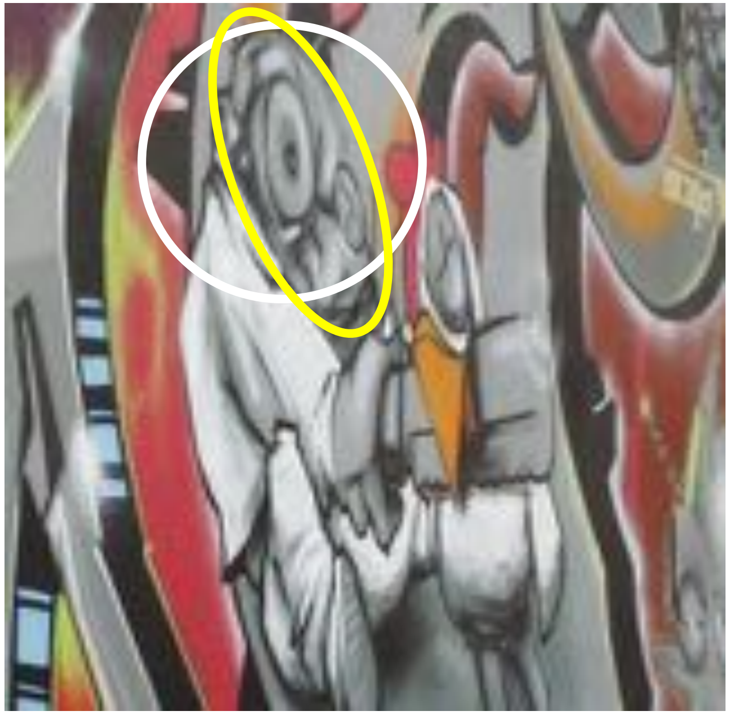



David G. Lowe. **"Distinctive image features from scale-invariant keypoints."** *IJCV* 60 (2), pp. 91-110, 2004.

Slide: S. Lazebnik

# Affine adaptation

- Affine transformation approximates viewpoint changes for roughly planar objects and roughly orthographic cameras



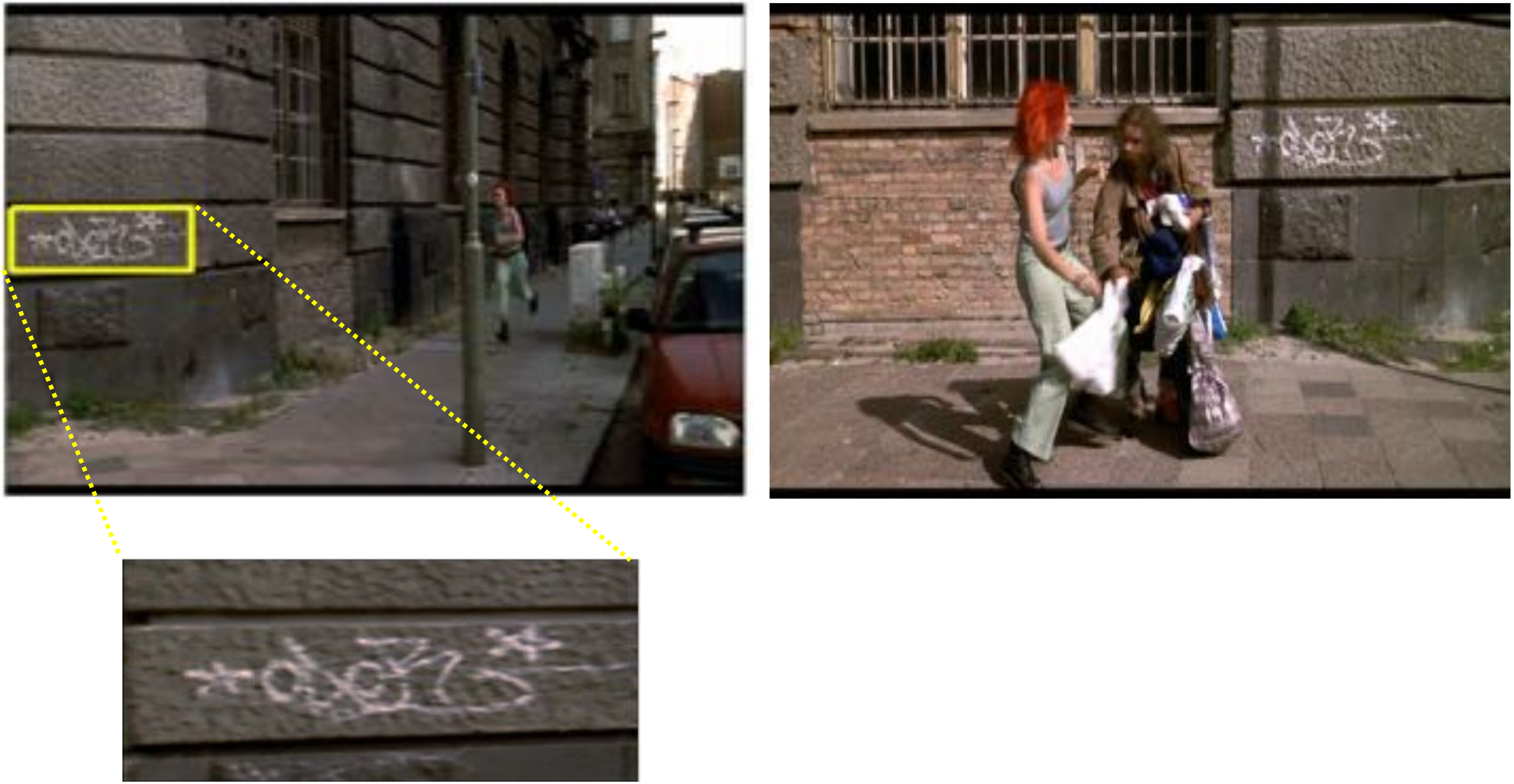
## Approach

0. **Pre-processing:**
   - Detect local features.
   - Extract descriptor for each feature.

1. **Matching:** Establish tentative (putative) correspondences based on local appearance of individual features (their descriptors).

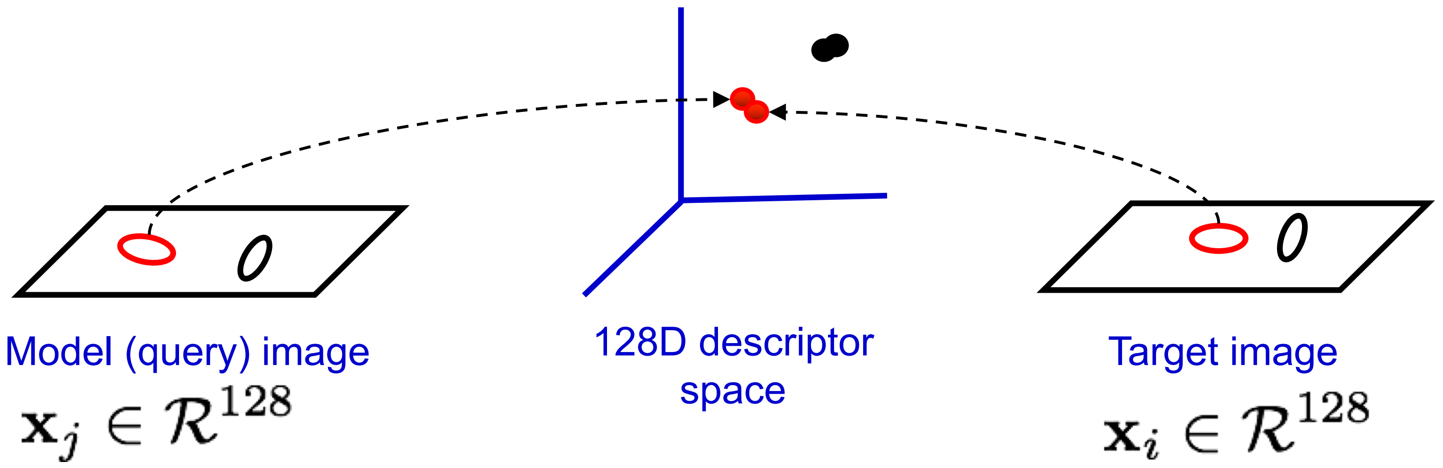
2. **Verification:** Verify matches based on semi-local / global geometric relations.

# Example I: Two images -“Where is the Graffiti?”



# Step 1. Establish tentative correspondence

Establish tentative correspondences between object model image and target image by nearest neighbour matching on SIFT vectors




Need to solve some variant of the "nearest neighbor problem" for all feature vectors, $\mathbf{x}_j \in \mathcal{R}^{128}$, in the query image:

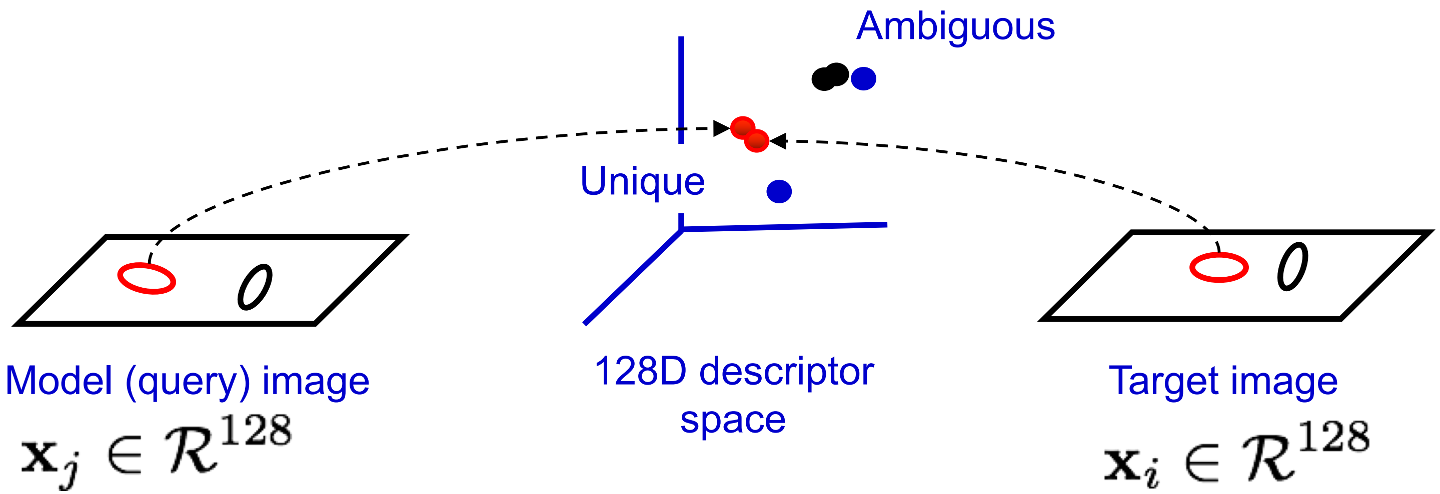
$$\forall j \; NN(j) = \arg\min_i ||\mathbf{x}_i - \mathbf{x}_j||,$$

where, $\mathbf{x}_i \in \mathcal{R}^{128}$ , are features in the target image.

Can take a long time if many target images are considered.

# Step 1. Establish tentative correspondence

Examine the distance to the 2nd nearest neighbour [Lowe, IJCV 2004]




If the 2nd nearest neighbour is much further than the 1st nearest neighbour

Match is more "unique" or discriminative.

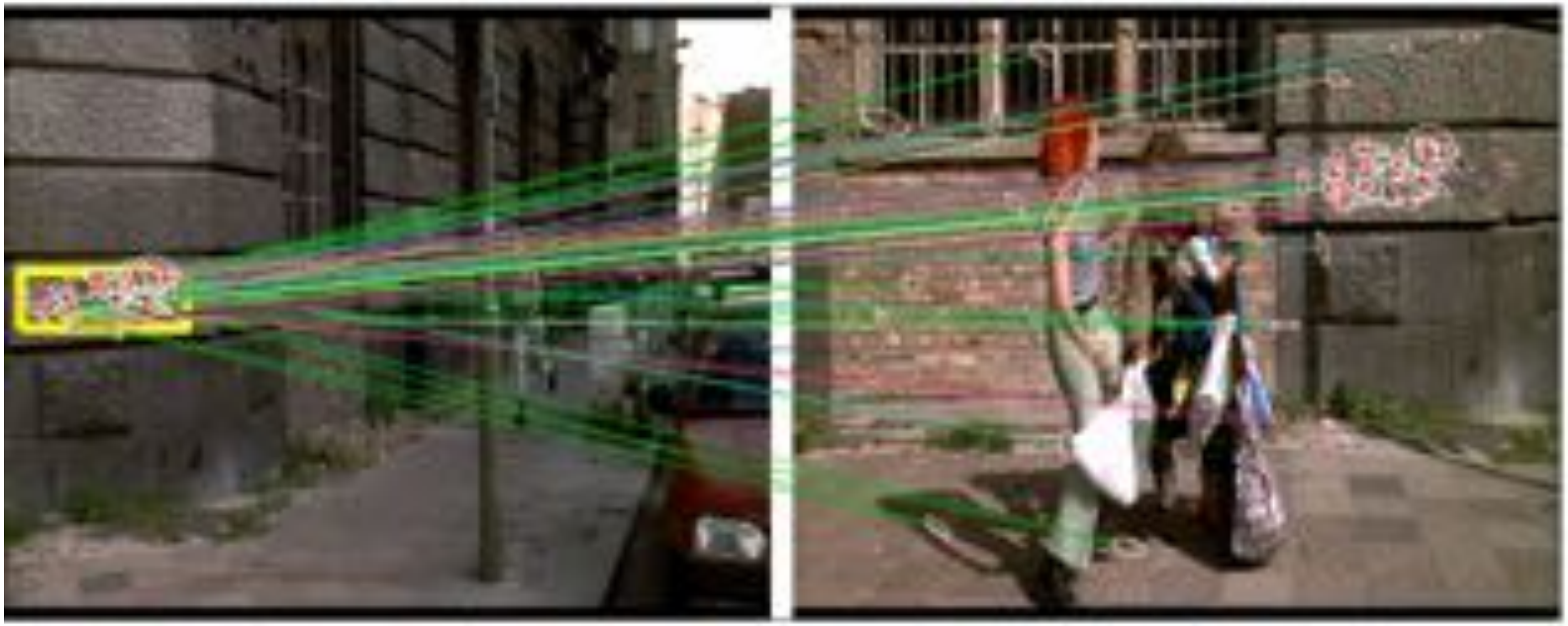
Measure this by the ratio: $r = d_{1NN} / d_{2NN}$

r is between 0 and 1

r is small the match is more unique.

Works very well in practice.

# Problem with matching on local descriptors alone



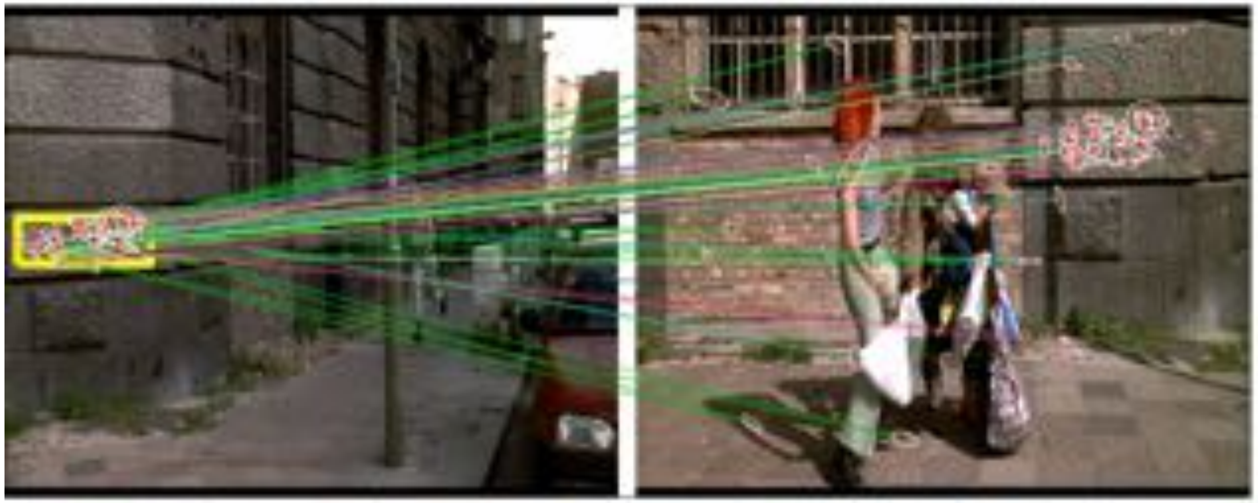
- too much individual invariance
- each region can affine deform independently (by different amounts)
- locally appearance can be ambiguous

Solution: use semi-local and global spatial relations to verify matches.

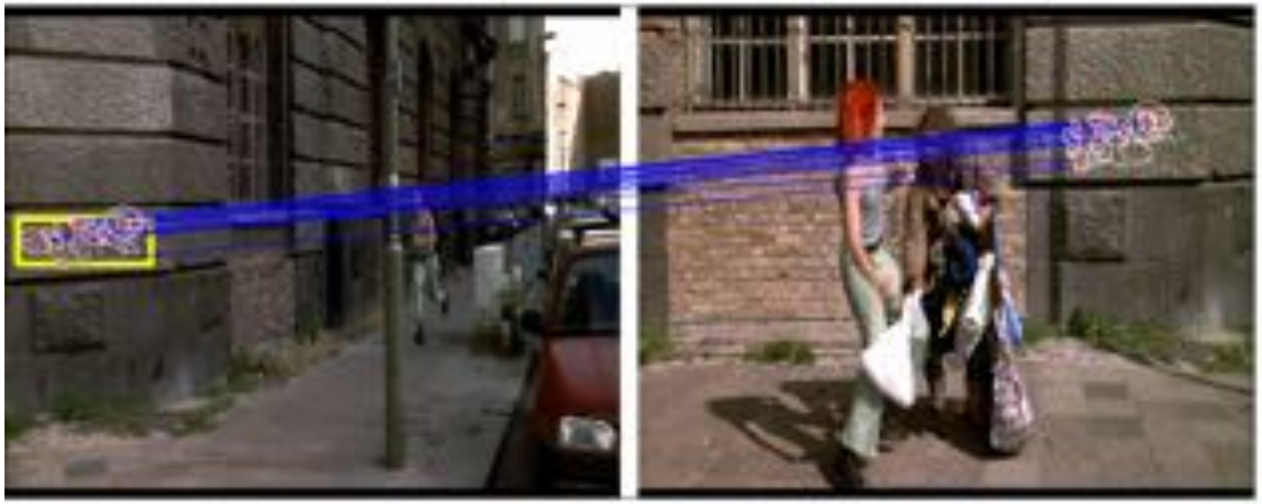
# Example I: Two images -“Where is the Graffiti?”

Initial matches

Nearest-neighbor search based on appearance descriptors alone.



After spatial verification



# Approach

0. **Pre-processing:**
   - Detect local features.
   - Extract descriptor for each feature.

1. **Matching:** Establish tentative (putative) correspondences based on local appearance of individual features (their descriptors).

2. **Verification:** Verify matches based on semi-local / global geometric relations.

## Step 2: Spatial verification (now)

a. Semi-local constraints

Constraints on spatially close-by matches

b. Global geometric relations

Require a consistent global relationship between all matches

# Semi-local constraints: Example I. – neighbourhood consensus

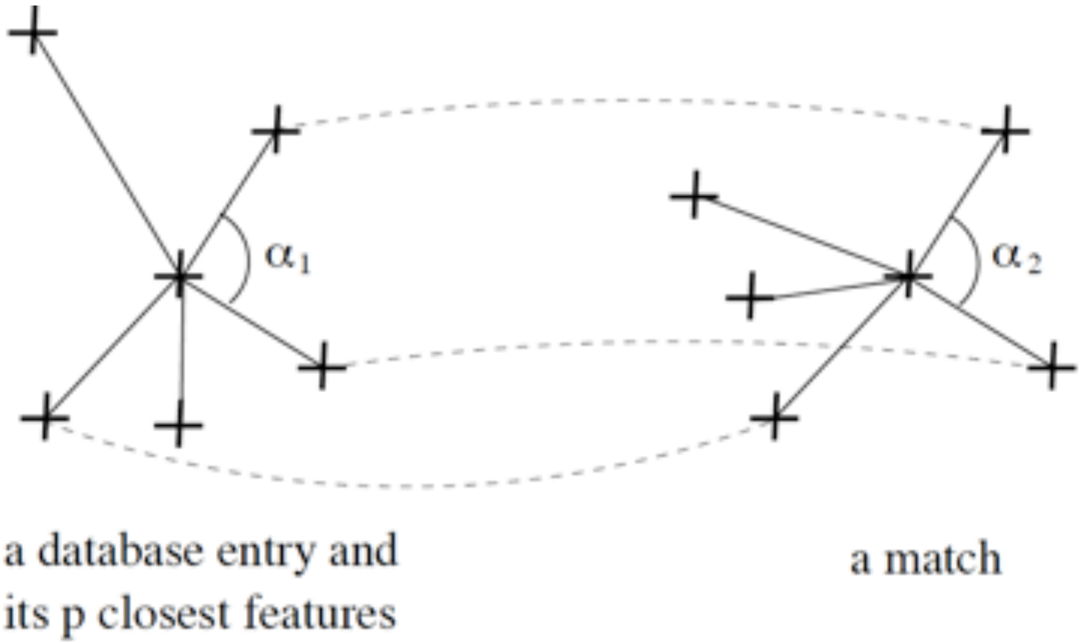
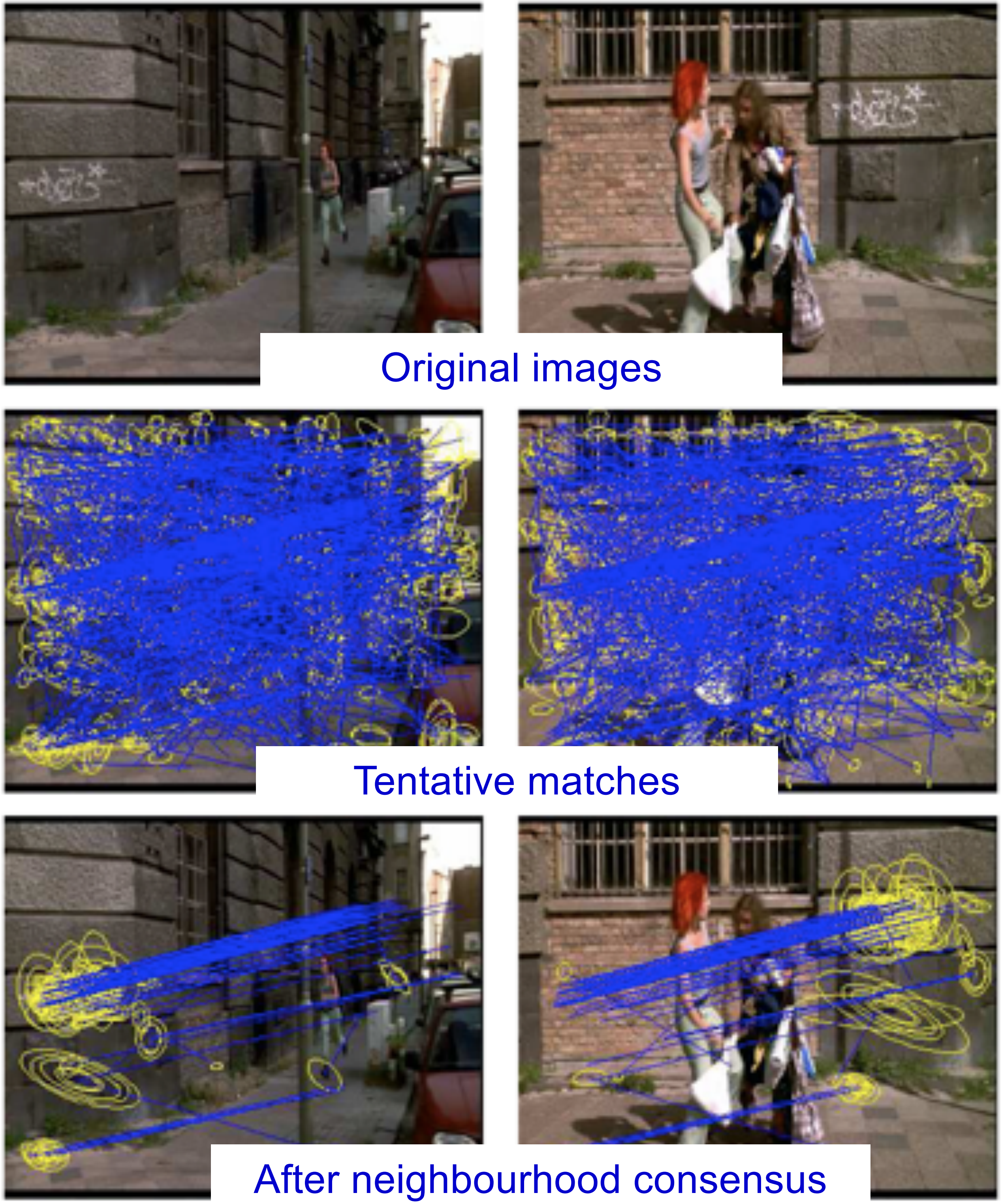



Fig. 4. Semi-local constraints : neighbours of the point have to match and angles have to correspond. Note that not all neighbours have to be matched correctly.

[Schmid&Mohr, PAMI 1997]

Semi-local constraints:

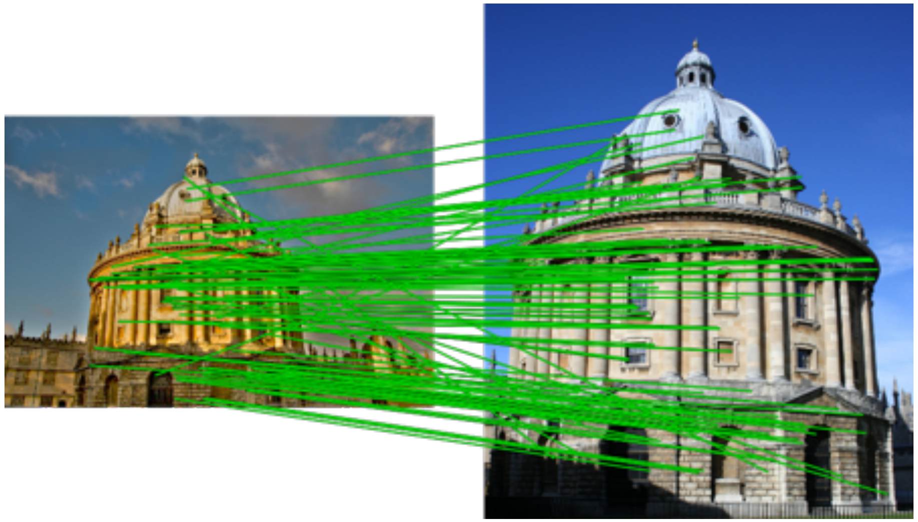
Example I. – neighbourhood consensus



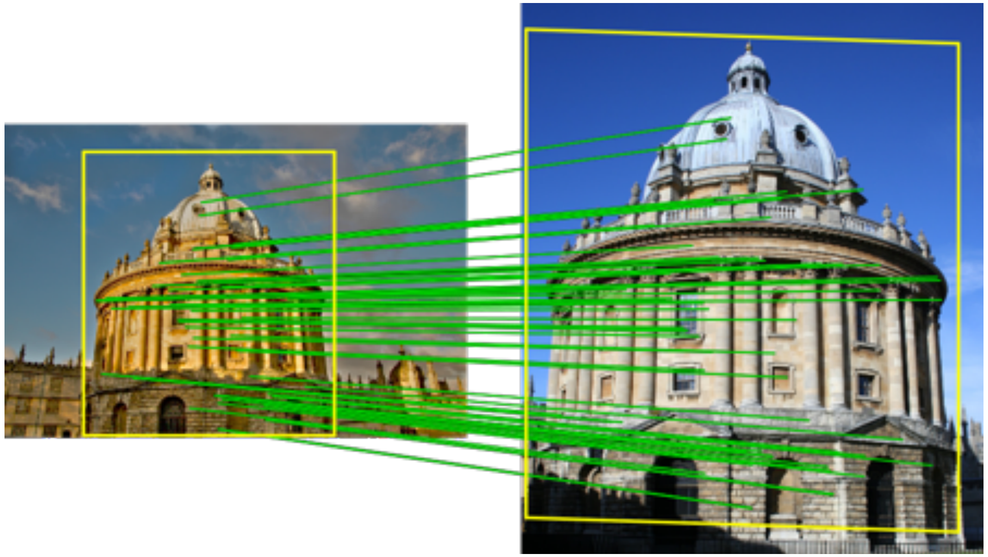

[Schaffalitzky & Zisserman, CIVR 2004]

# Geometric verification with global constraints

- All matches must be consistent with a global geometric relation / transformation.
- Need to simultaneously (i) estimate the geometric relation / transformation and (ii) the set of consistent matches



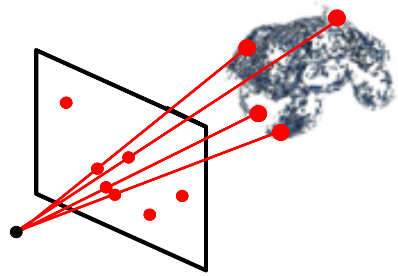
Tentative matches



Matches consistent with an affine transformation

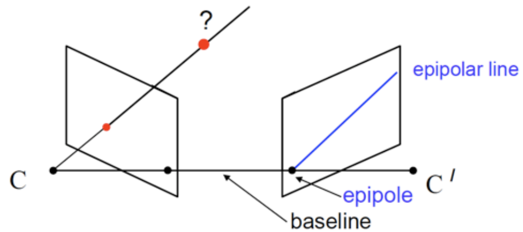
# Examples of global constraints

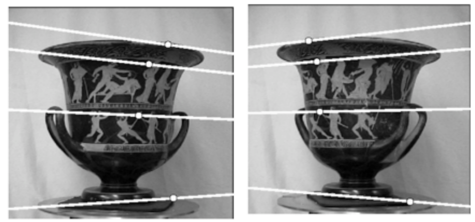
## 1 view and known 3D model.



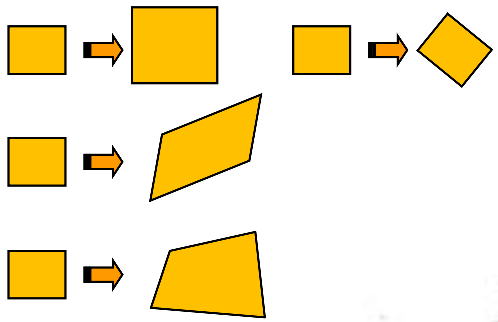
- Consistency with a (known) 3D model.

## 2 views



- Epipolar constraint
- **2D transformations**
    - Similarity transformation
    - Affine transformation
    - Projective transformation



## N-views

Are images consistent with a 3D model?

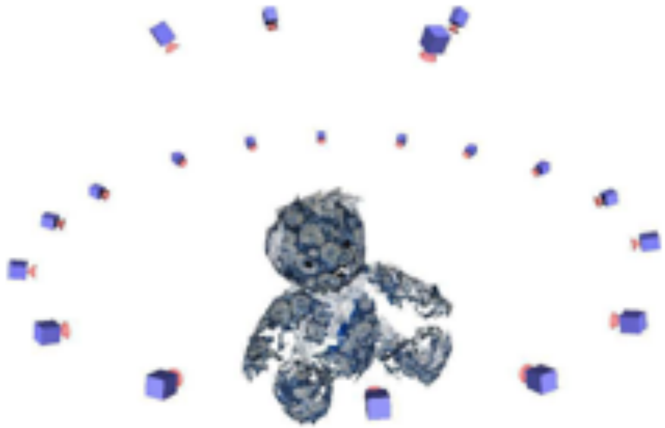
# 3D constraint: example

- Matches must be consistent with a 3D model

Offline: Build a 3D model

3 (out of 20) images used to build the 3D model

(a)



Recovered 3D model

[Lazebnik, Rothganger, Schmid, Ponce, CVPR'03]

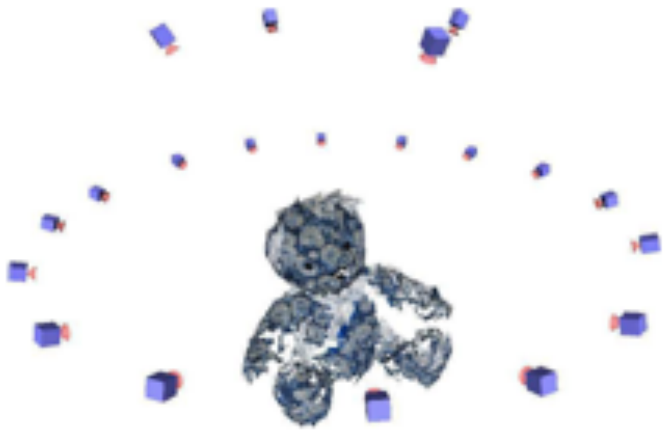
# 3D constraint: example

- Matches must be consistent with a 3D model

Offline: Build a 3D model

3 (out of 20) images used to build the 3D model

(a)



Recovered 3D model

At test time:

Object recognized in a previously unseen pose

Recovered pose

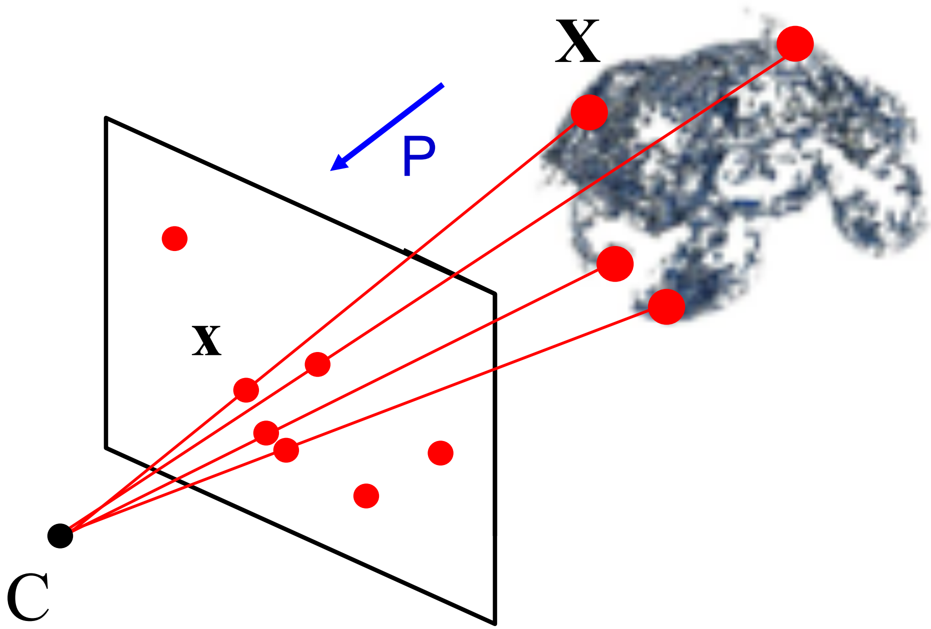
(d)

[Lazebnik, Rothganger, Schmid, Ponce, CVPR’03]

# 3D constraint: example

Given 3D model (set of known 3D points X's) and a set of measured 2D image points x,

find camera matrix P and a set of geometrically consistent correspondences $x \leftrightarrow X$.




$$\mathbf{x} = \mathrm{P}\mathbf{X}$$

P : 3 × 4 matrix

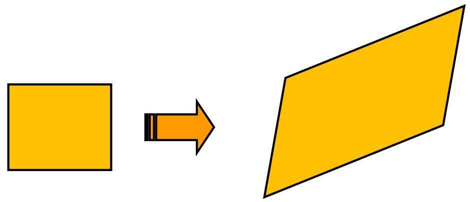
$\mathbf{X}$ : 4-vector

$\mathbf{x}$ : 3-vector

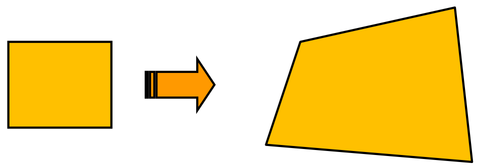
# 2D transformation models

Similarity
(translation,
scale, rotation)

Affine

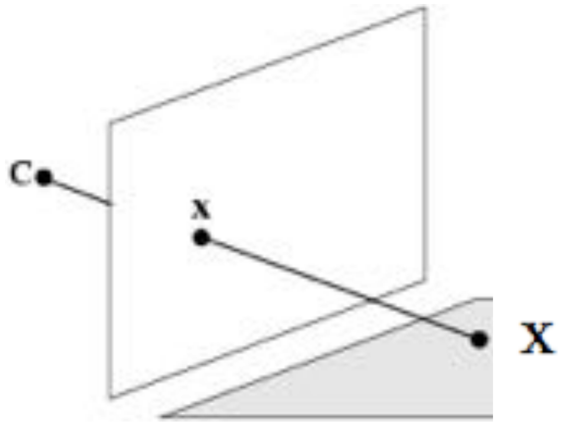


Projective
(homography)



Why are 2D planar transformations important?

# Recall perspective projection




$$\mathbf{x} = \mathrm{P}\mathbf{X}$$

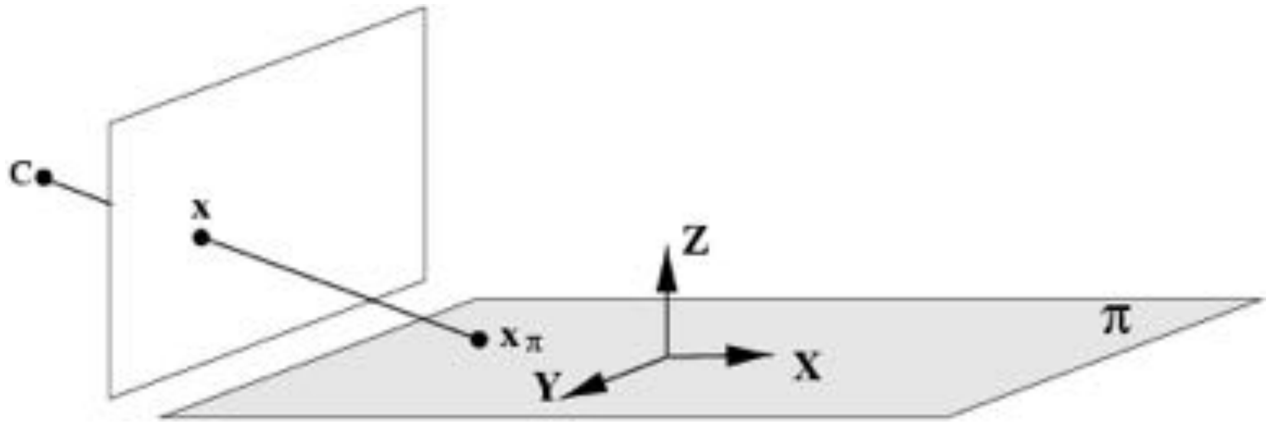
P : $3 \times 4$ matrix

$\mathbf{X}$ : 4-vector

$\mathbf{x}$ : 3-vector

Slide credit: A. Zisserman

# Plane projective transformations




Choose the world coordinate system such that the plane of the points has zero z coordinate. Then the $3 \times 4$ matrix P reduces to

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{pmatrix} \mathrm{x} \\ \mathrm{y} \\ 0 \\ 1 \end{pmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{14} \\ p_{21} & p_{22} & p_{24} \\ p_{31} & p_{32} & p_{34} \end{bmatrix} \begin{pmatrix} \mathrm{x} \\ \mathrm{y} \\ 1 \end{pmatrix}$$

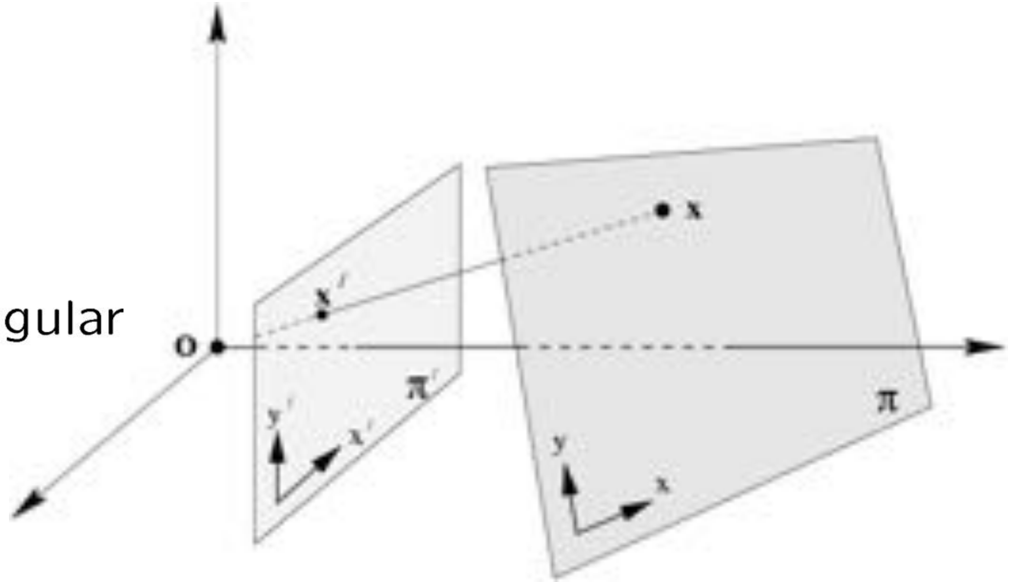
which is a $3 \times 3$ matrix representing a general plane to plane projective transformation.

Slide credit: A. Zisserman

# Projective transformations continued

$$\begin{pmatrix} x'_1 \\ x'_2 \\ x'_3 \end{pmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

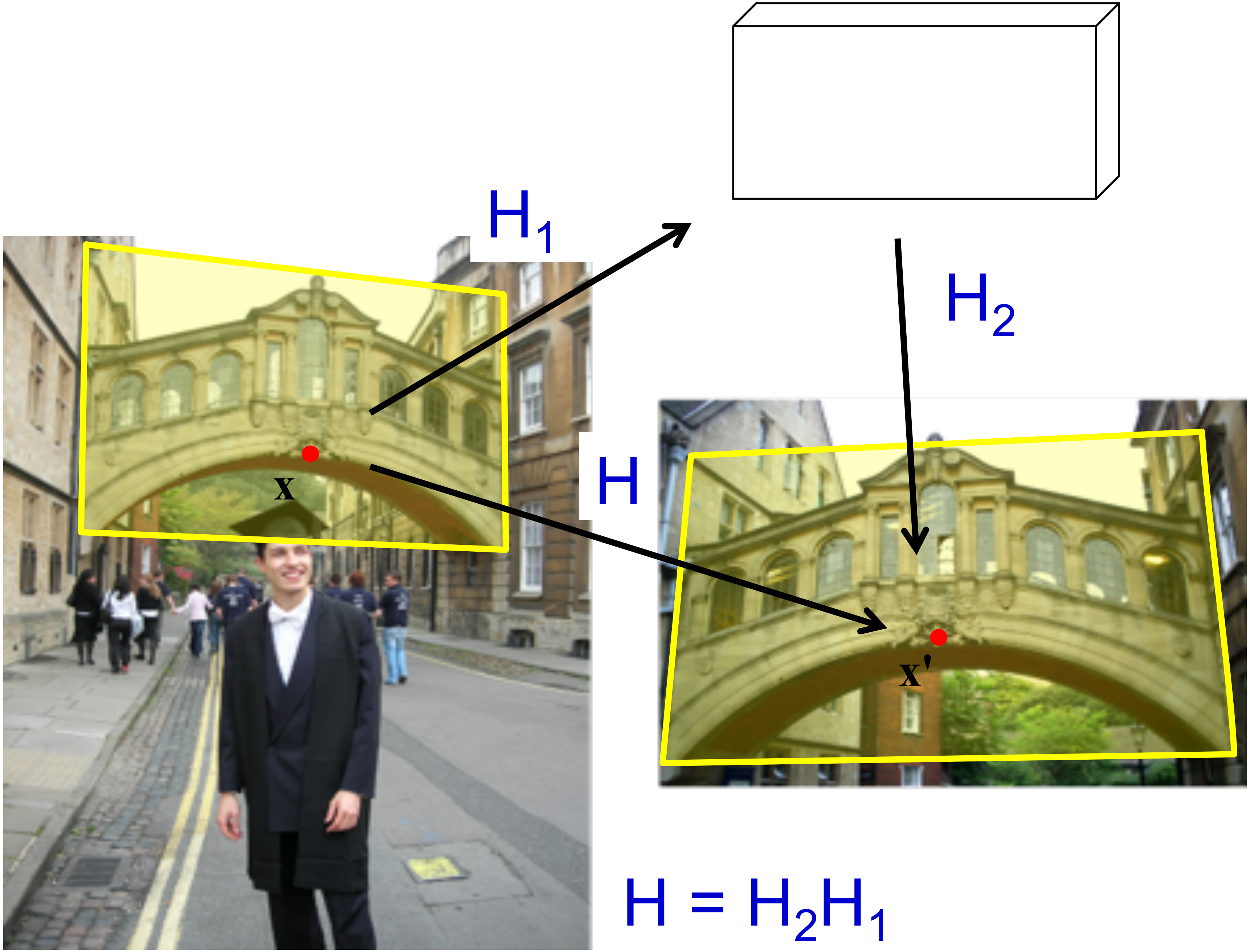
or $\mathbf{x}' = \mathtt{H}\mathbf{x}$, where $\mathtt{H}$ is a $3 \times 3$ non-singular homogeneous matrix.




- This is the most general transformation between the world and image plane under imaging by a perspective camera.
- It is often only the 3 x 3 form of the matrix that is important in establishing properties of this transformation.
- A projective transformation is also called a ``homography'' and a ``collineation''.
- H has 8 degrees of freedom. How many points are needed to compute H?

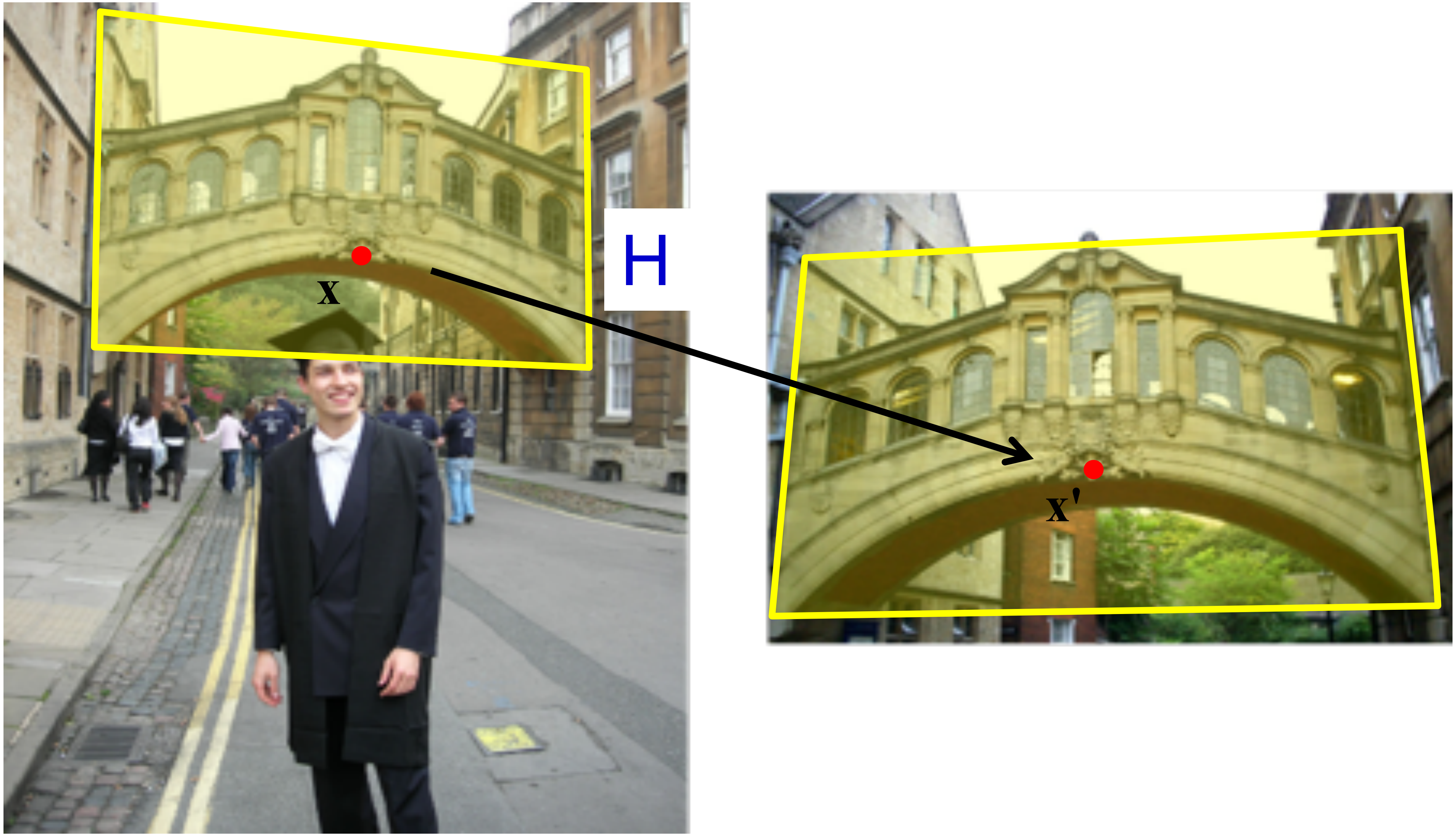
Slide credit: A. Zisserman

# Planes in the scene induce *homographies*

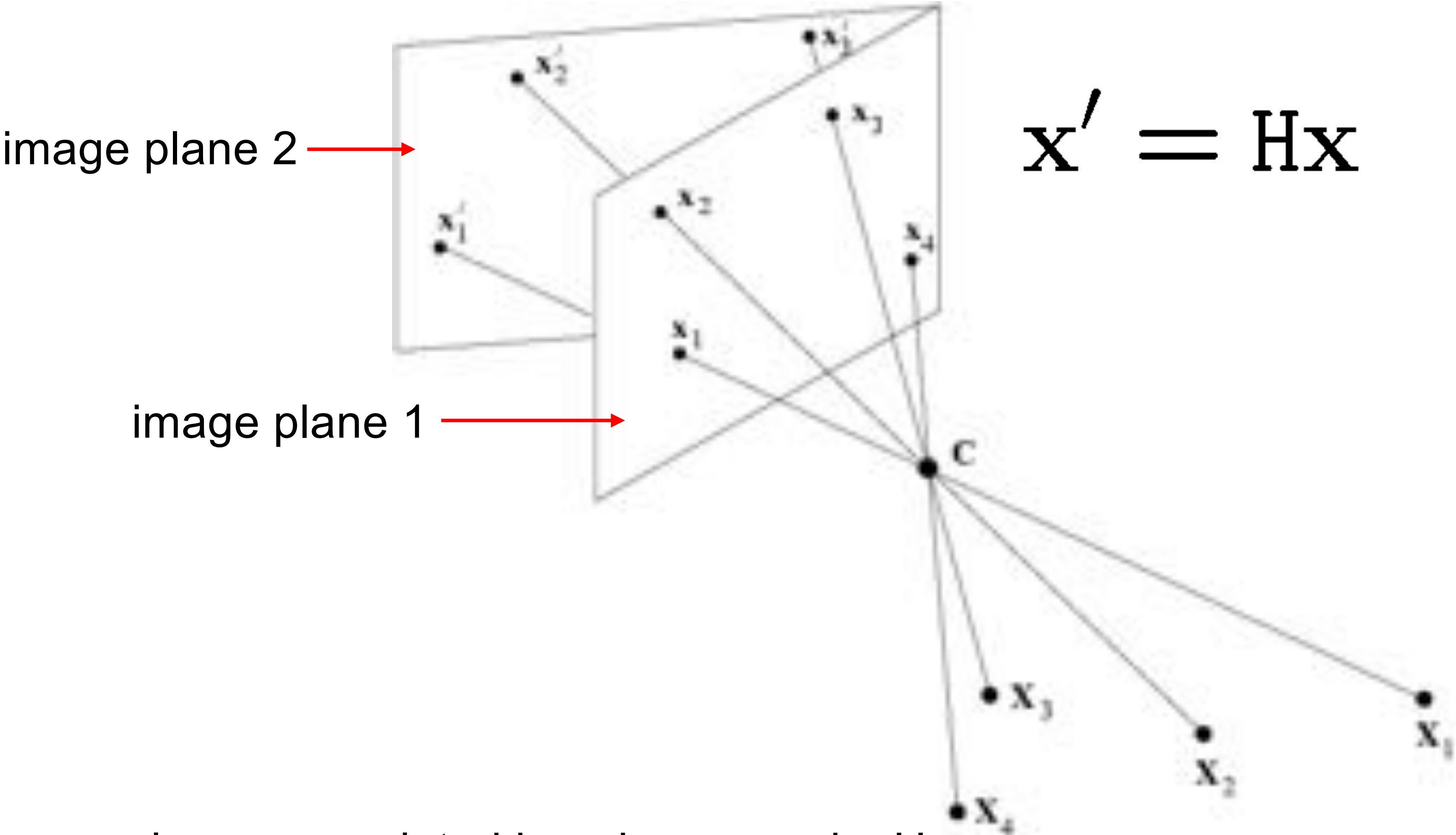


# Planes in the scene induce *homographies*

Points on the plane transform as $x' = H x$, where $x$ and $x'$ are image points (in homogeneous coordinates), and H is a 3x3 matrix.



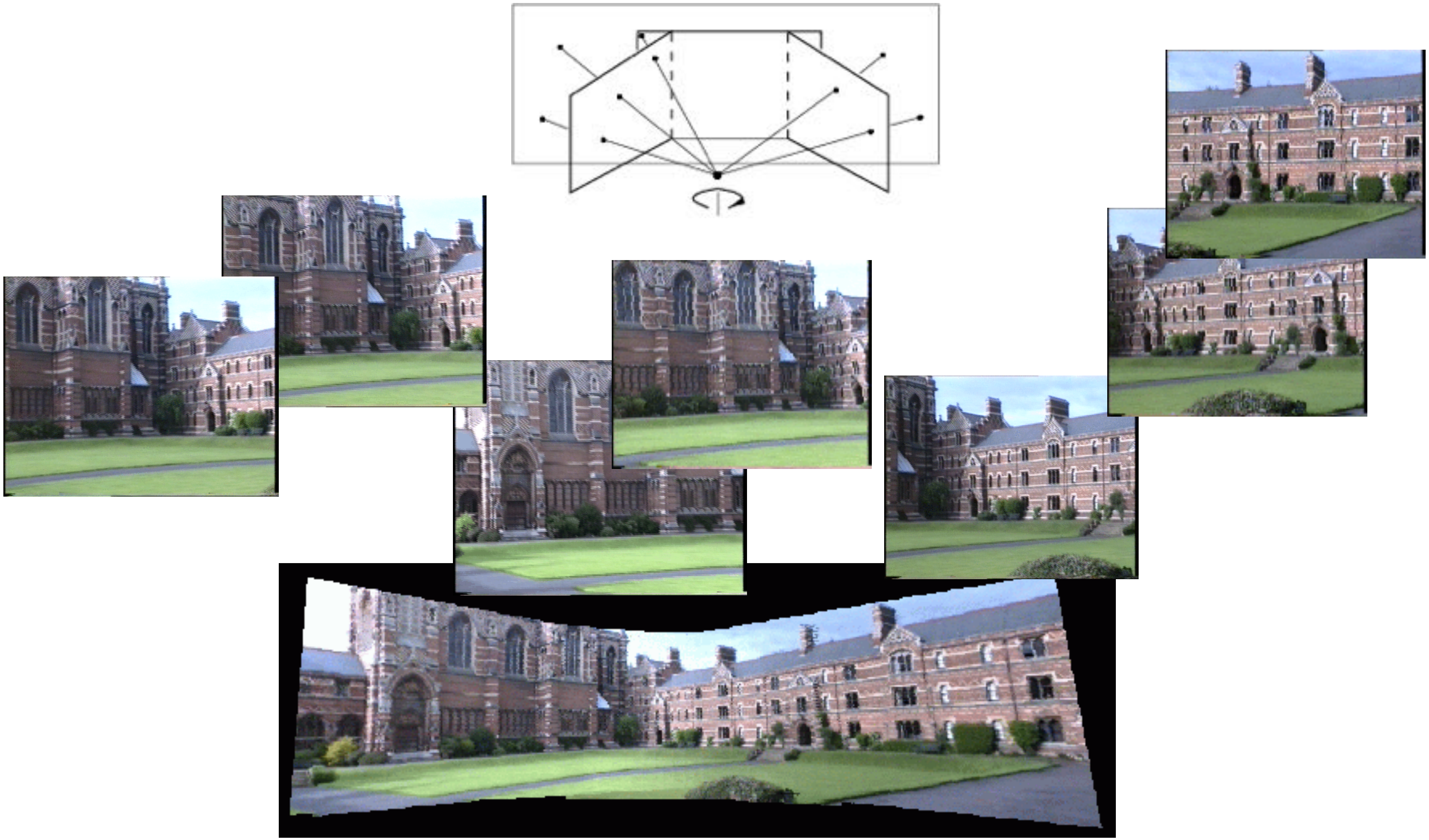
# Case II: Cameras rotating about their centre




$$\mathbf{x}' = H\mathbf{x}$$

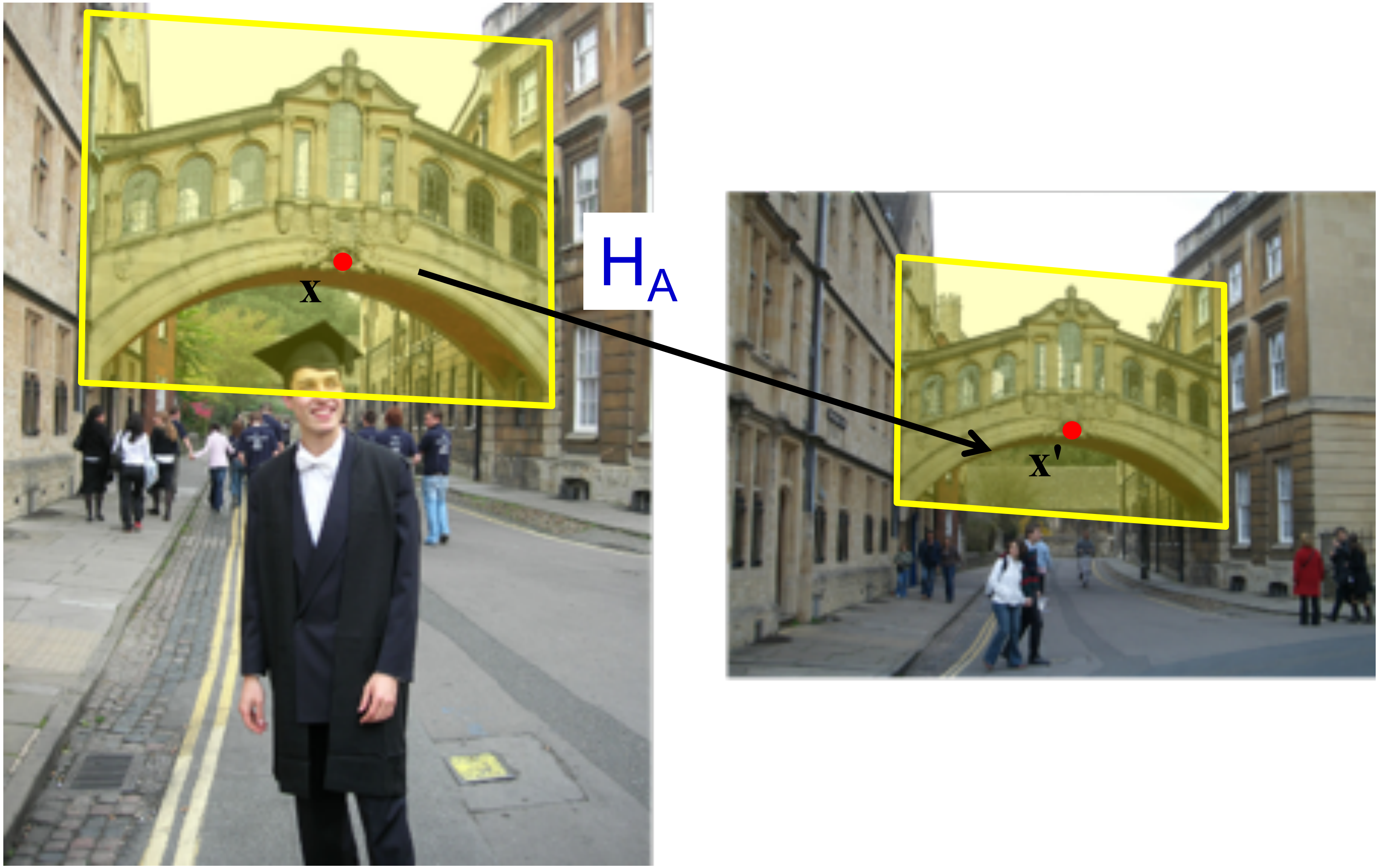
- The two image planes are related by a homography H
- H depends only on the relation between the image planes and camera centre, C, not on the 3D structure

# Case II: Example of a rotating camera



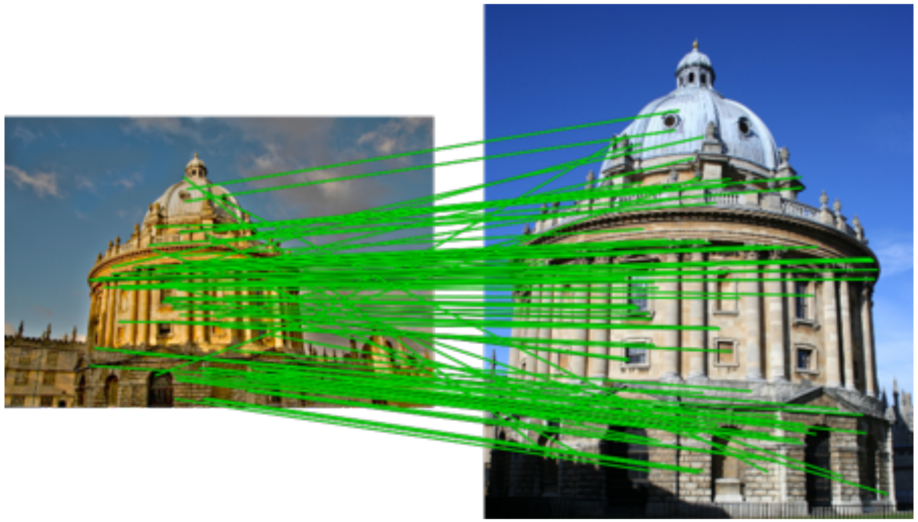
Images courtesy of A. Zisserman.

# Homography is often approximated well by 2D affine geometric transformation

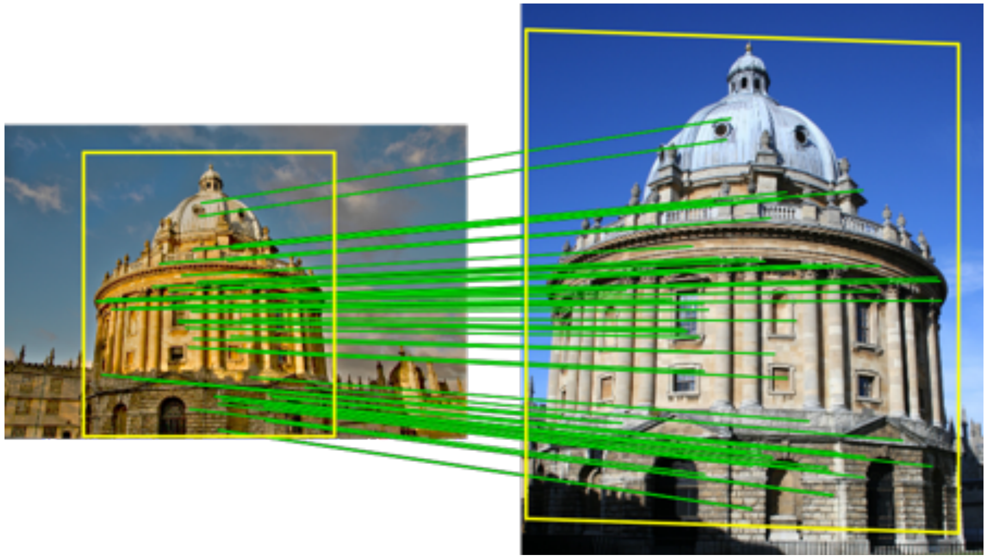


# Homography is often approximated well by 2D affine geometric transformation – Example II.

Two images with similar camera viewpoint



Tentative matches



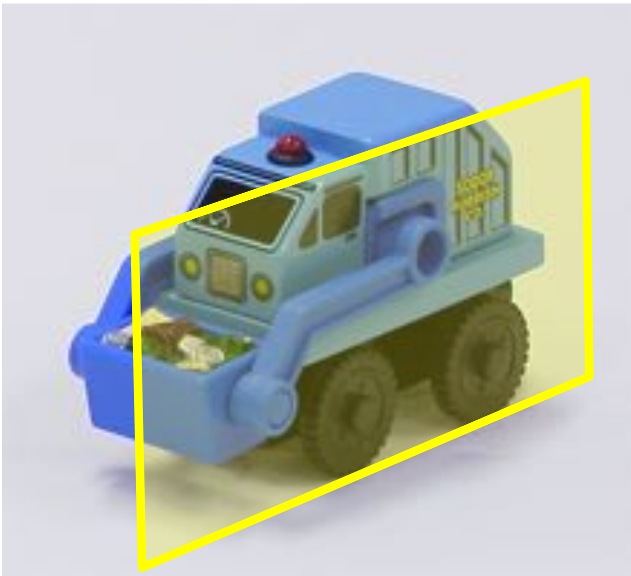
Matches consistent with an affine transformation

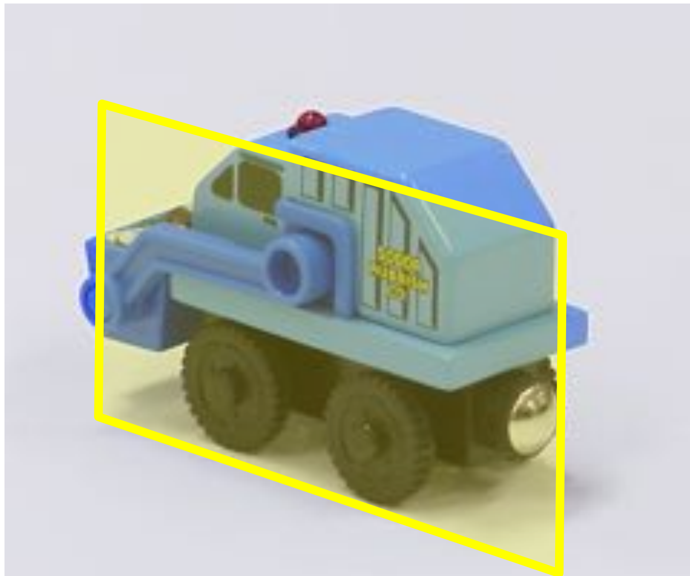
# Example: estimating 2D affine transformation

- Simple fitting procedure (linear least squares)
- Approximates viewpoint changes for roughly planar objects and roughly orthographic cameras
- Can be used to initialize fitting for more complex models

# Example: estimating 2D affine transformation

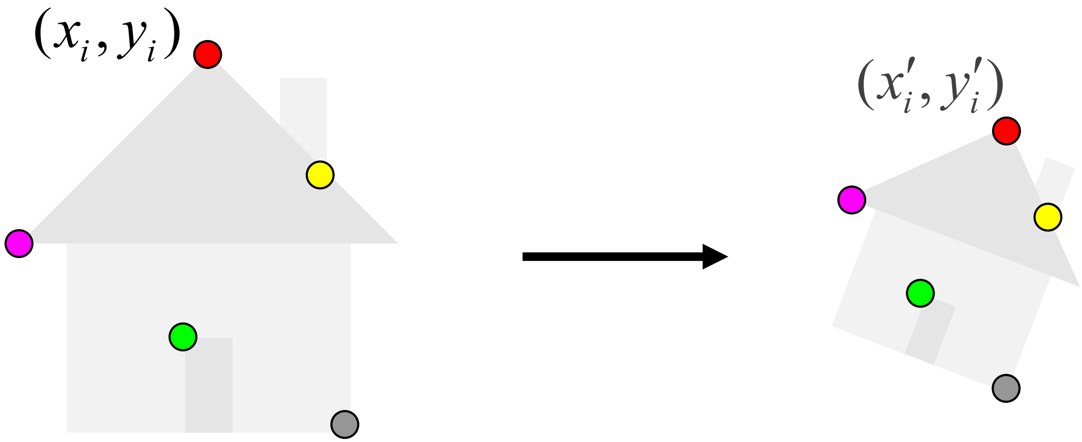
- Simple fitting procedure (linear least squares)
- Approximates viewpoint changes for **roughly planar objects** and **roughly orthographic cameras**
- Can be used to initialize fitting for more complex models





# Fitting an affine transformation

Assume we know the correspondences, how do we get the transformation?




$$\begin{bmatrix} x_i' \\ y_i' \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}$$

$$\begin{bmatrix} & & \cdots & & & \\ x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \\ & & \cdots & & & \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_1 \\ t_2 \end{bmatrix} = \begin{bmatrix} \cdots \\ x_i' \\ y_i' \\ \cdots \end{bmatrix}$$

# Fitting an affine transformation

$$
\begin{bmatrix} & & \cdots & & & \\ x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \\ & & \cdots & & & \end{bmatrix}
\begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_1 \\ t_2 \end{bmatrix}
=
\begin{bmatrix} \cdots \\ x_i' \\ y_i' \\ \cdots \end{bmatrix}
$$

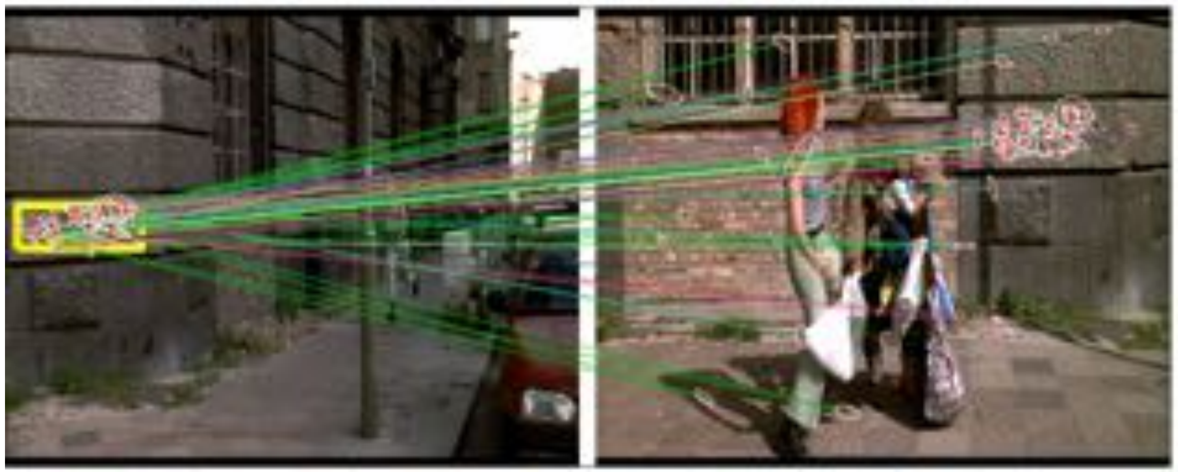
Linear system with six unknowns

Each match gives us two linearly independent equations: need at least three to solve for the transformation parameters

# Dealing with outliers

The set of putative matches may contain a high percentage (e.g. 90%) of outliers

How do we fit a geometric transformation to a small subset of all possible matches?

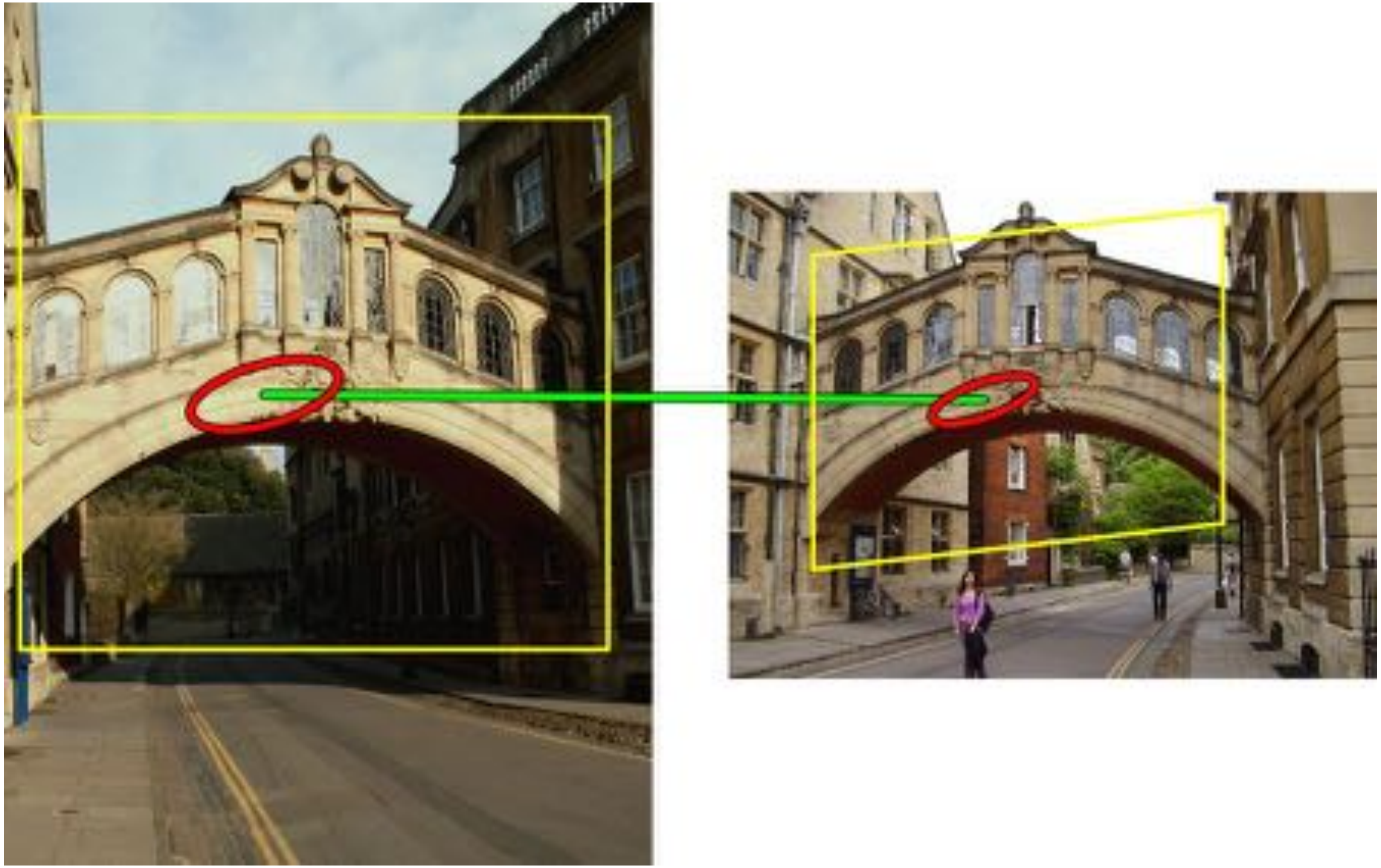


# Example: restricted affine transform

1. Test each correspondence

# Example: restricted affine transform

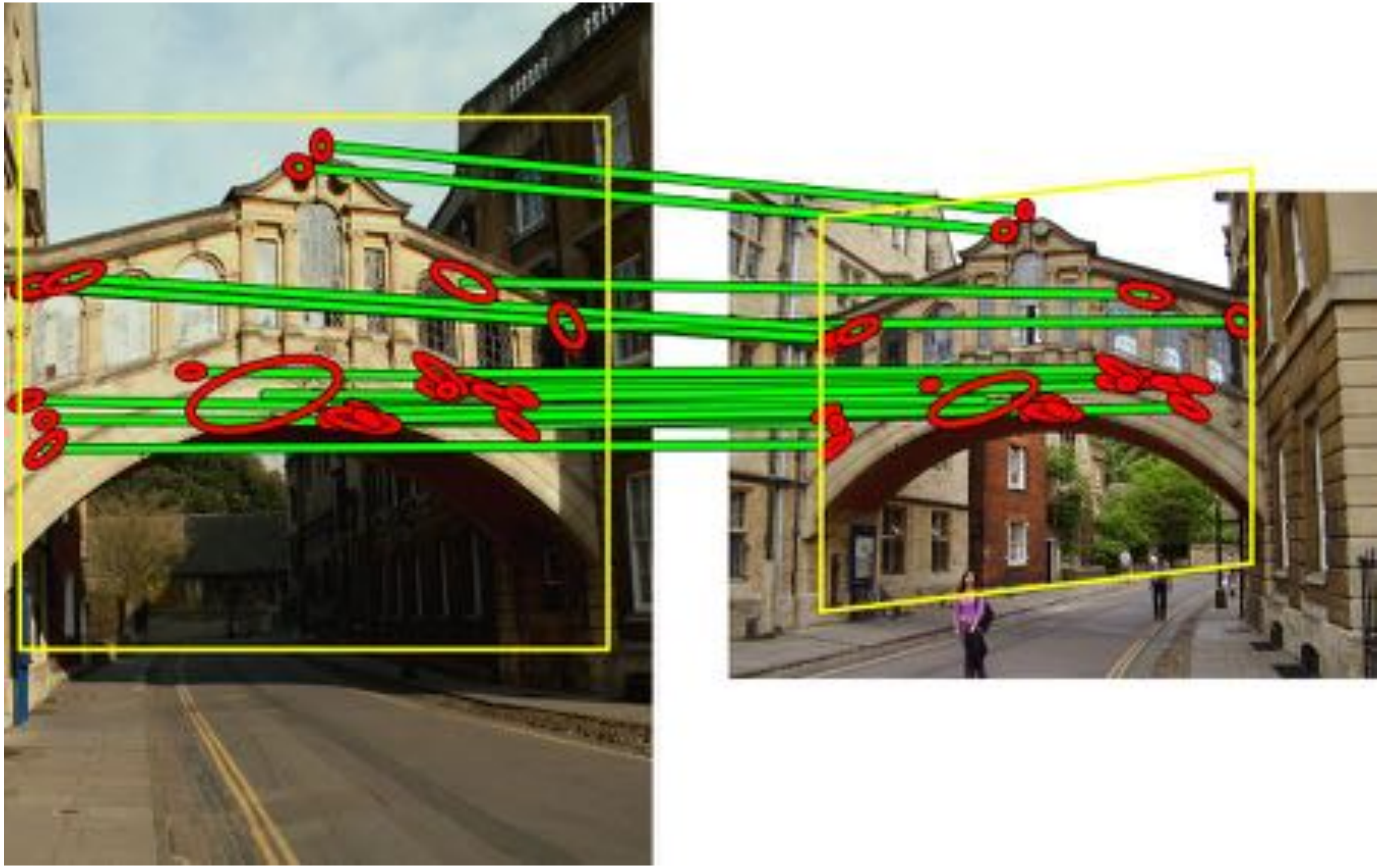
2. Compute a (restricted) planar affine transformation (5 dof)



Need just one correspondence

# Example: restricted affine transform

3. Score by number of consistent matches

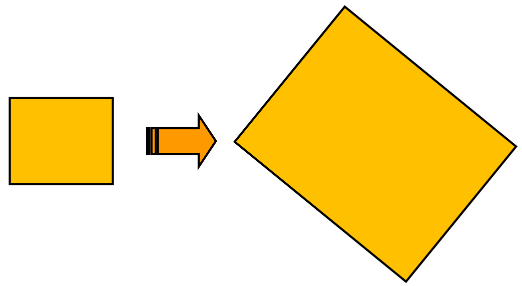


Re-estimate full affine transformation (6 dof)

# Example II: Similarity transformation

**Similarity transformation** is specified by four parameters: scale factor s, rotation θ, and translations $t_x$ and $t_y$.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = sR(\theta) \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$



Recall, each SIFT detection has: position ($x_i$, $y_i$), scale $s_i$, and orientation $\theta_i$.

How many correspondences are needed to compute similarity transformation?

# RANSAC (references)

M. Fischler and R. Bolles, “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,” Comm. ACM, 1981

R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision, 2nd ed., 2004.

Extensions:

B. Tordoff and D. Murray, “Guided Sampling and Consensus for Motion Estimation, ECCV’03

D. Nister, “Preemptive RANSAC for Live Structure and Motion Estimation, ICCV’03

Chum, O.; Matas, J. and Obdrzalek, S.: Enhancing RANSAC by Generalized Model Optimization, ACCV’04

Chum, O.; and Matas, J.: Matching with PROSAC - Progressive Sample Consensus , CVPR 2005

Philbin, J., Chum, O., Isard, M., Sivic, J. and Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching, CVPR’07

Chum, O. and Matas. J.: Optimal Randomized RANSAC, PAMI’08

Lebeda, Matas, Chum: Fixing the locally optimized RANSAC, BMVC’12 (code available).

# Geometric verification for visual search (references)


Schmid and Mohr, Local gray-value invariants for image retrieval, PAMI 1997

Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Object retrieval with large vocabularies and fast spatial matching. CVPR (2007)

Perdoch, M., Chum, O., Matas, J.: Efficient representation of local geometry for large scale object retrieval. CVPR (2009)

Wu, Z., Ke, Q., Isard, M., Sun, J.: Bundling features for large scale partial-duplicate web image search. In: CVPR (2009)

Jegou, H., Douze, M., Schmid, C.: Improving bag-of-features for large scale image search. IJCV 87(3), 316–336 (2010)

Lin, Z., Brandt, J.: A local bag-of-features model for large-scale object retrieval. ECCV 2010)

Zhang, Y., Jia, Z., Chen, T.: Image retrieval with geometry preserving visual phrases. In: CVPR (2011)

Tolias, G., Avrithis, Y.: Speeded-up, relaxed spatial matching. In: ICCV (2011)

Shen, X., Lin, Z., Brandt, J., Avidan, S., Wu, Y.: Object retrieval and localization with spatially-constrained similarity measure and k-nn re-ranking. In: CVPR. IEEE (2012)

H. Stewénius, S. Gunderson, J. Pilet. Size matters: exhaustive geometric verification for image retrieval, ECCV 2012.

# Summary

## Finding correspondences in images is useful for

- Image matching, panorama stitching
- Object recognition
- Large scale image search: next time

## Beyond local point matching

- Semi-local relations
- Global geometric relations:
    - Epipolar constraint
    - 3D constraint (when 3D model is available)
    - 2D tnfs: Similarity / Affine / Homography
- Algorithms:
    - RANSAC
    - [Hough transform]

$$\boxed{\mathbf{x}'^{\top}\mathbf{F}\mathbf{x} = 0}$$

$$\boxed{\mathrm{x} = \mathrm{PX}}$$

$$\boxed{\mathrm{x}' = \mathrm{Hx}}$$

# Convolutional neural networks for correspondence and instance-level recognition

## Still an active area of research with some successes.

**Instance level matching and retrieval:**

Babenko et al., ECCV 2014
Razavian et al., ArXiv 2014
Azizpour et al., ArXiv 2014
Babenko and Lempitsky, ICCV 2015
Gong et al., ECCV 2014
Altwaijry et al., CVPR 2015
Arandjelovic et al., CVPR 2016.
Radenovic and Chum, ECCV 2016.
A Gordo, J Almazan, J Revaud, D Larlus, ECCV 2016.

**Patch descriptors and correspondence:**

Verdie, Kwank, Fua and Lepetit, CVPR 2015
Fischer, A Dosovitskiy and T Brox, Arxiv, 2015
Simo-Serra, Trulls, Ferraz, Kokkinos, Fua, and Moreno-Noguer, CVPR 2015
Han, Leung, Jia, Sukthankar, and C Berg, CVPR 2015
Zagoruyko and Komodakis, CVPR 2015
Gwak, Savarese and Chandraker, ECCV 2016
KM Yi, E Trulls, V Lepetit, P Fua, ECCV 2016
Balntas, Johns, Tang, and Mikolajczyk, CVPR 2016
A Mishchuk, D Mishkin, F Radenovic, J Matas, NIPS 2017

**Dense correspondence for motion estimation**

Fischer, Dosovitskiy, Ilg, Häusser, Hazırbaş, Golkov, van der Smagt, Cremers and Brox, ICCV 2015
T Zhou, M Brown, N Snavely, DG Lowe, CVPR 2017

# Learnable representations for estimating visual correspondence

Ignacio Rocco and Josef Sivic

Inria, Ecole Normale Supérieure, PSL
and
Czech Technical University in Prague

# Goal


Source

Target

# Goal


$$\text{Source} \xrightarrow{\mathcal{T}} \text{Target}$$

# Goal

Source $\xrightarrow{\mathcal{T}}$ Target

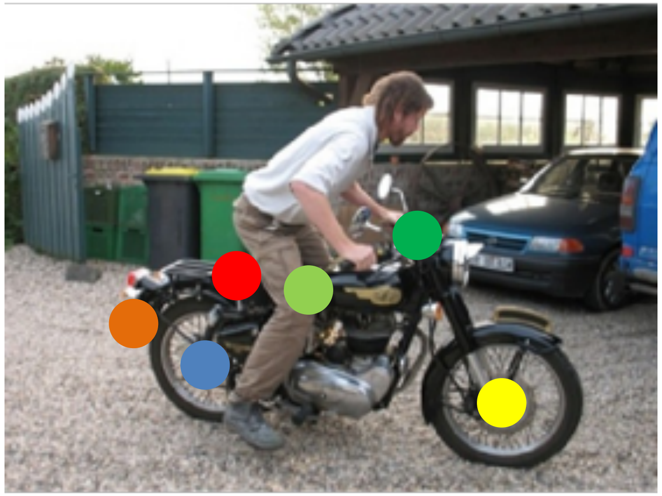
# Challenges

Substantial appearance differences

# Challenges


Presence of background clutter

# Challenges



Lack of large annotated image pair dataset

# Applications

Co-segmentation

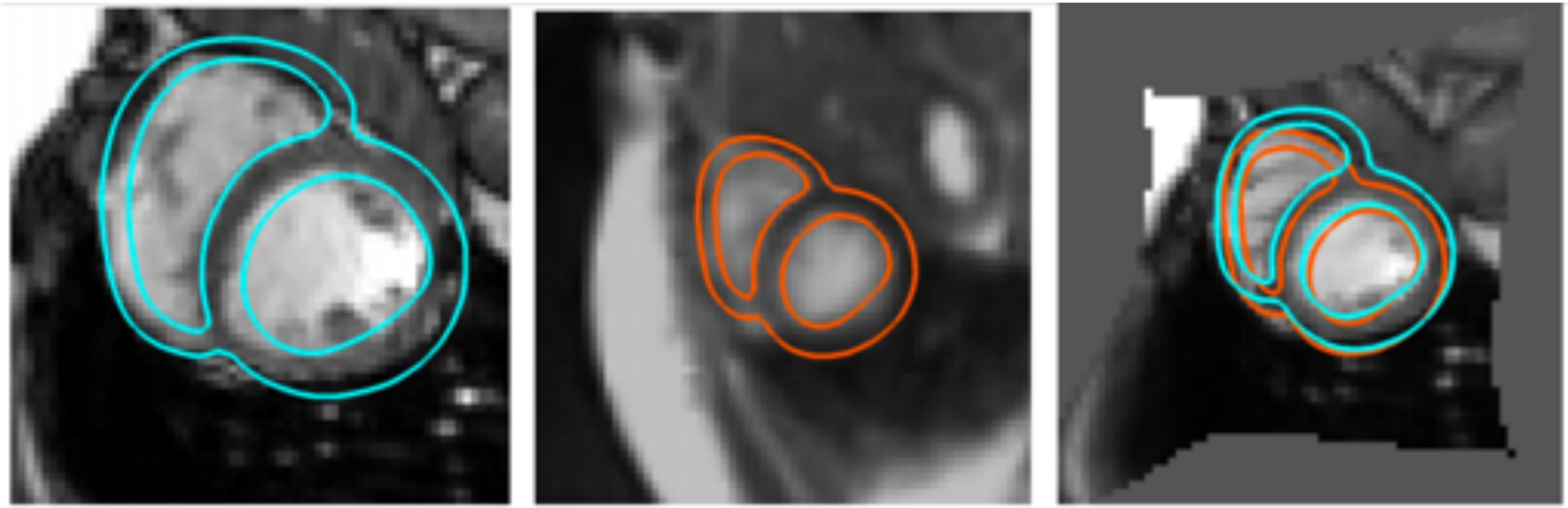
[Taniai et al. '16]

# Applications

Co-segmentation

[Taniai et al. '16]

# Applications



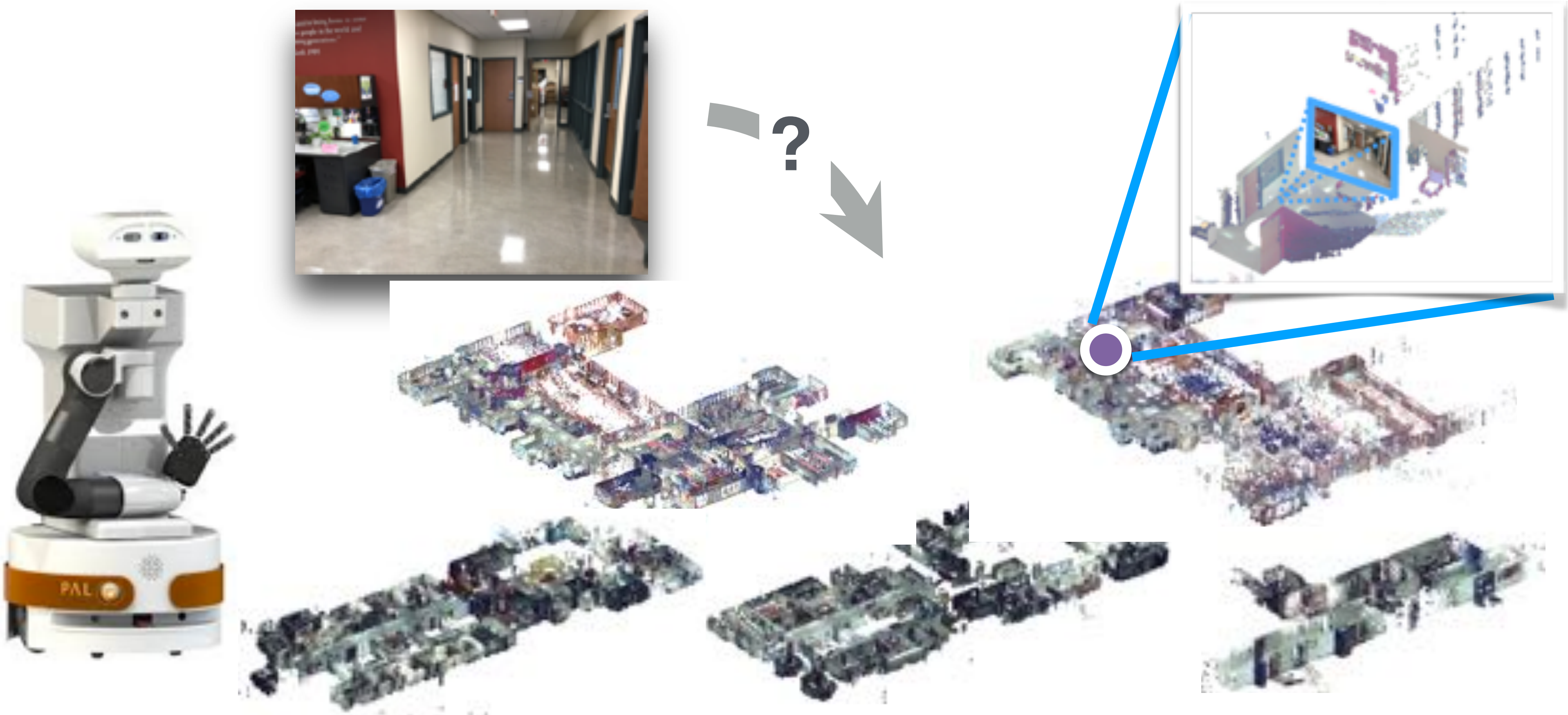
Medical image registration

[de Vos et al. '17, Rohé et al. '17]

# Applications

## Visual localization in indoor environments

[Taira et al., CVPR 2018]

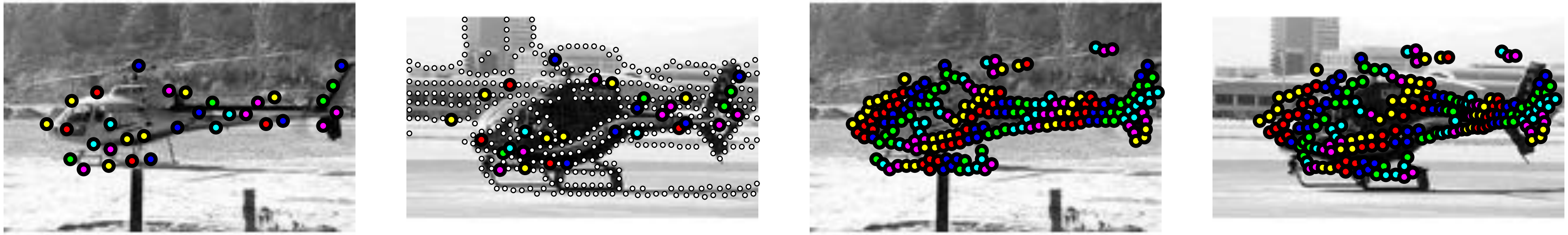


# Applications

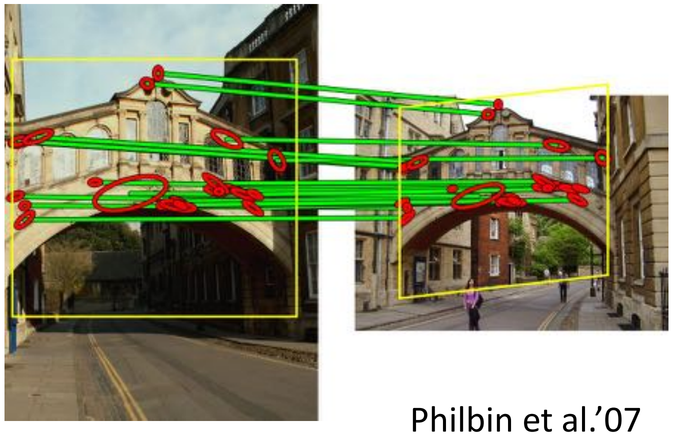
## Visual localization across changing conditions

[Sattler et al., CVPR 2018]

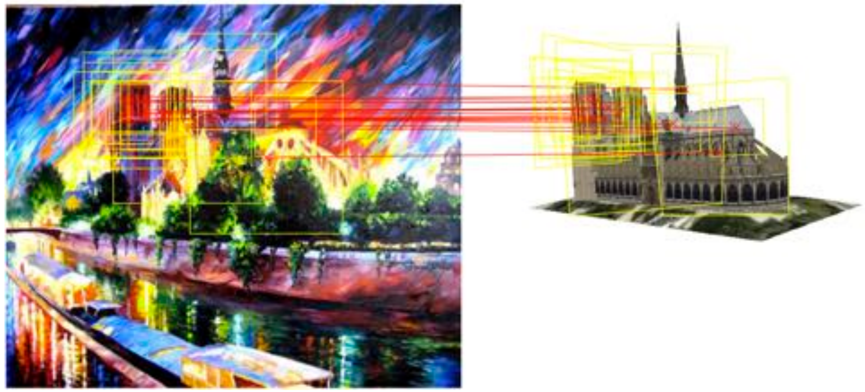
# Related work



[Berg and Malik’05]



Philbin et al.’07

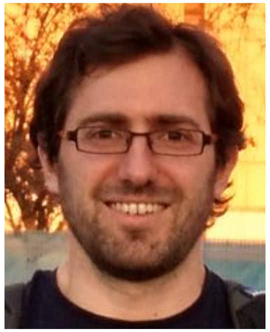


Aubry et al.’14

[Lamdan et al.’90, Leung et al.’95, Schmid and Mohr’97, Lowe’99, Fergus et al.’03, Berg and Malik’05, Philbin et al.’07, Liu et al.’08, Kim et al.’13, Revaud et al.’13, …]

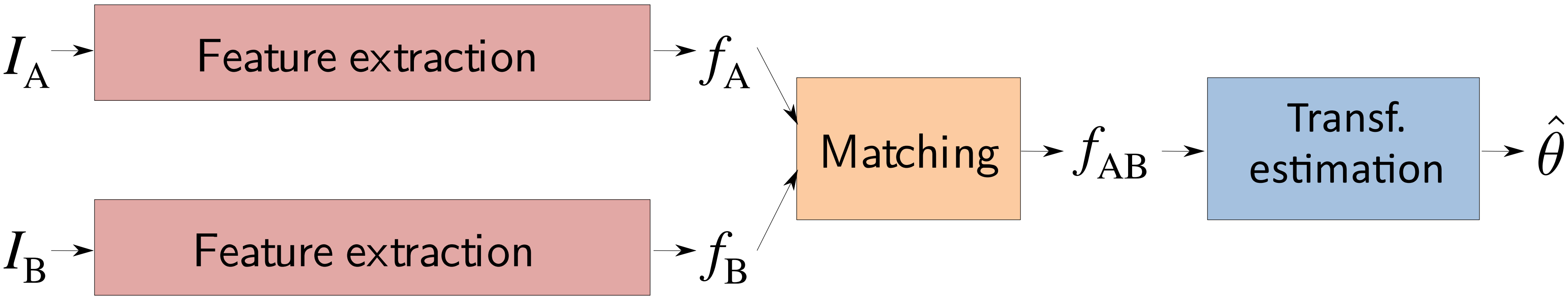
CZECH TECHNICAL UNIVERSITY IN PRAGUE

# Convolutional neural network architecture for geometric matching

Ignacio Rocco Relja Arandjelović Josef Sivic

# Classical image correspondence pipeline



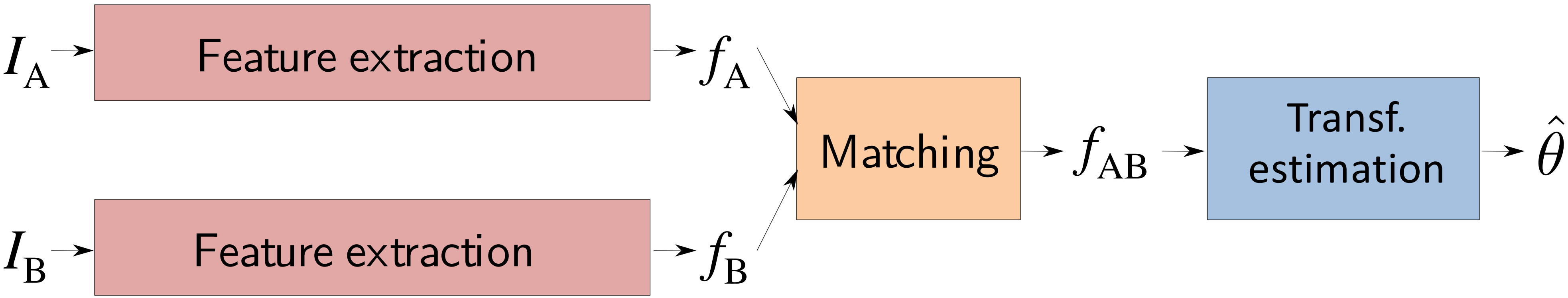

1. Feature extraction
(SIFT)

2. Matching
(Euclidean dist.+2nd NN test)

3. Transformation estimation
(RANSAC)

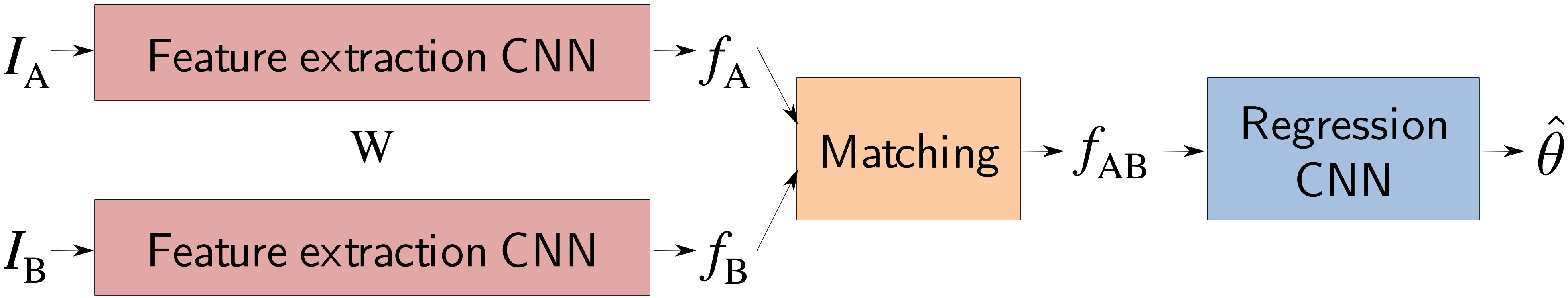
[Schmid and Mohr'97, Lowe'99, Berg'05, Philbin et al.'07, Liu et al.'08, Kim et al.'13, Revaud et al.'13, …]

# Classical image correspondence pipeline



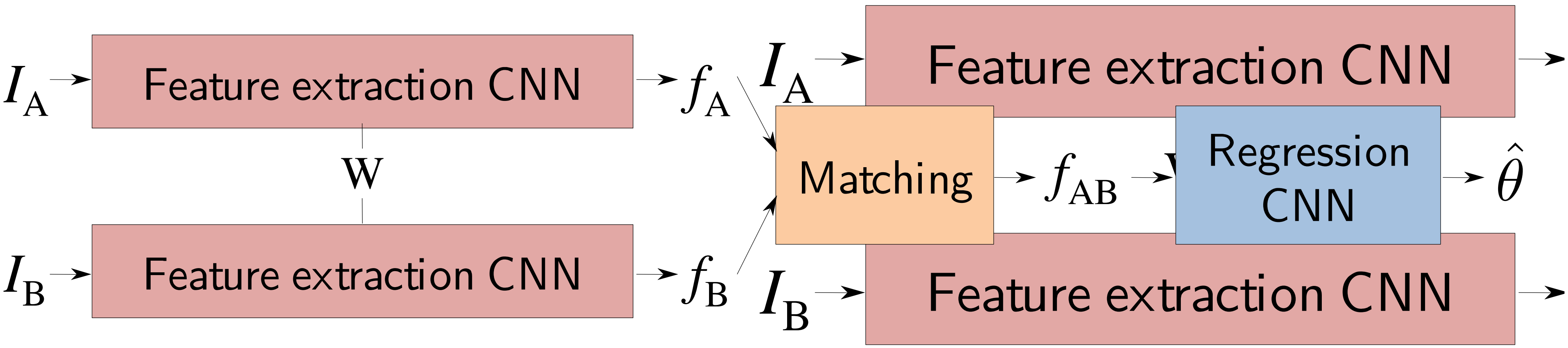

$\hat{\theta}$: geometric transformation parameters
(affine: 6-D vector)

# Proposed approach

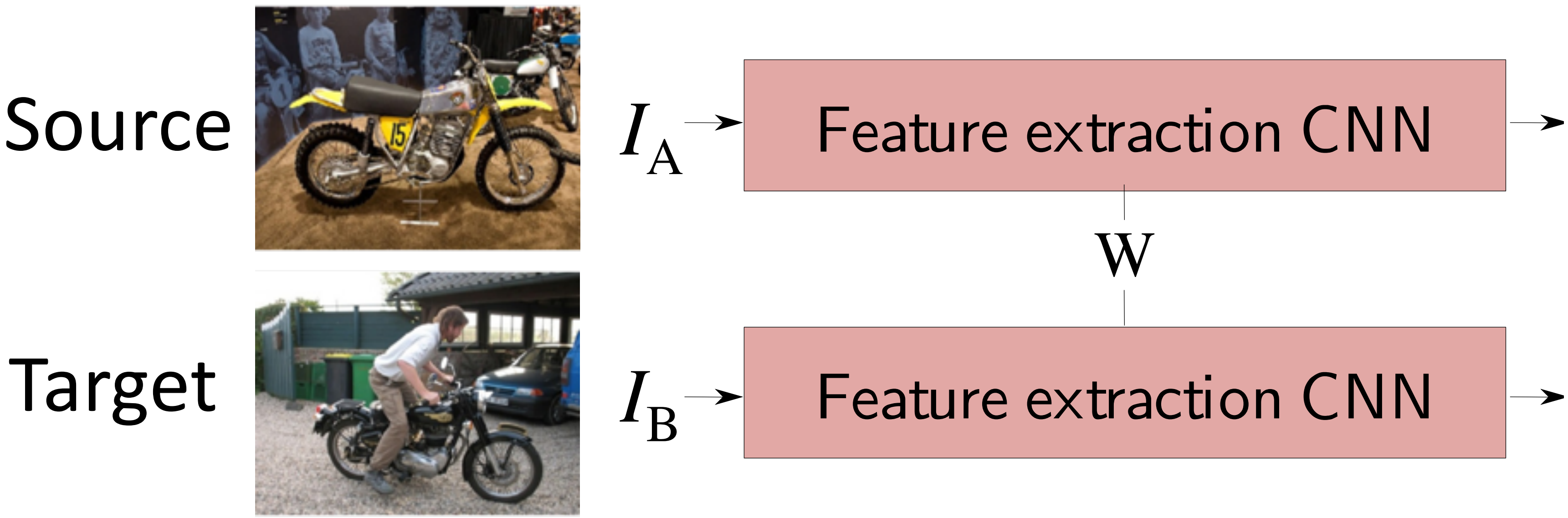



classical pipeline → CNN

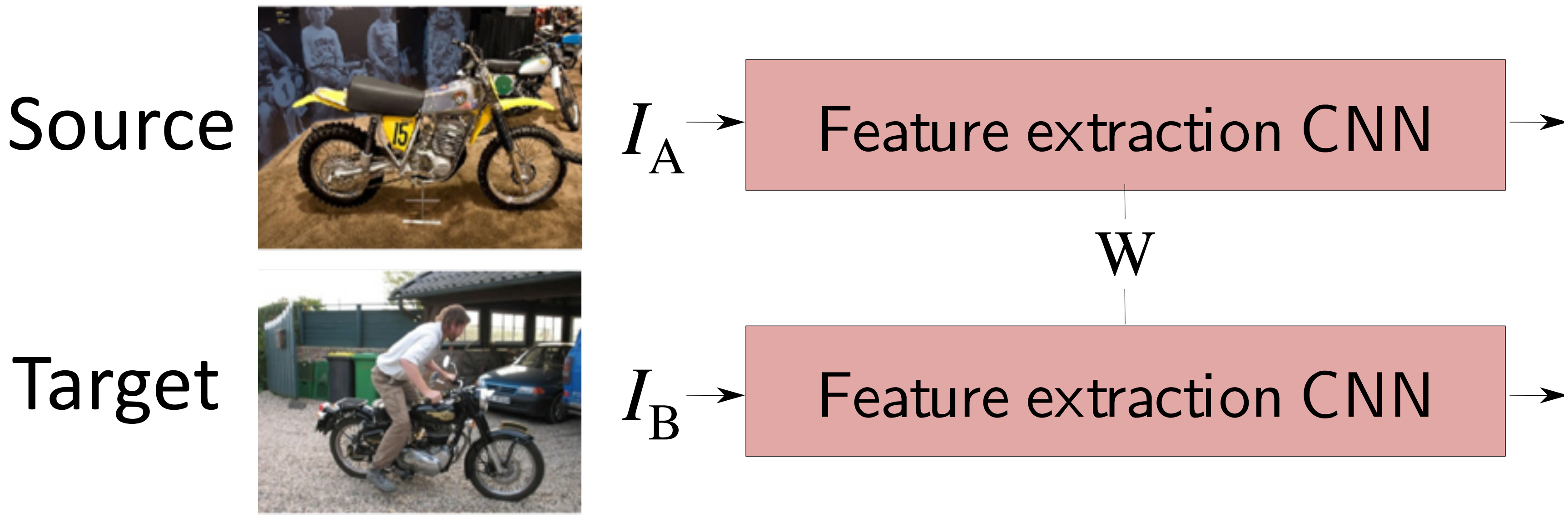
# Proposed approach



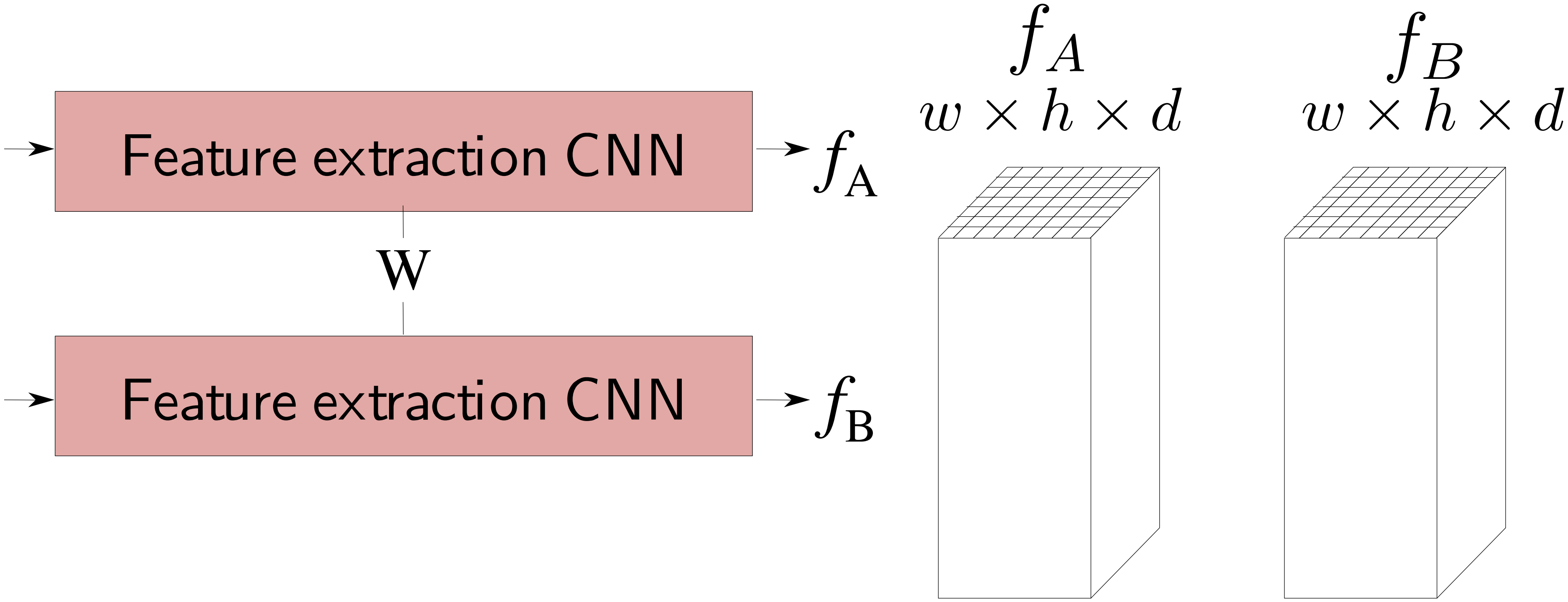
# Proposed approach



# Proposed approach

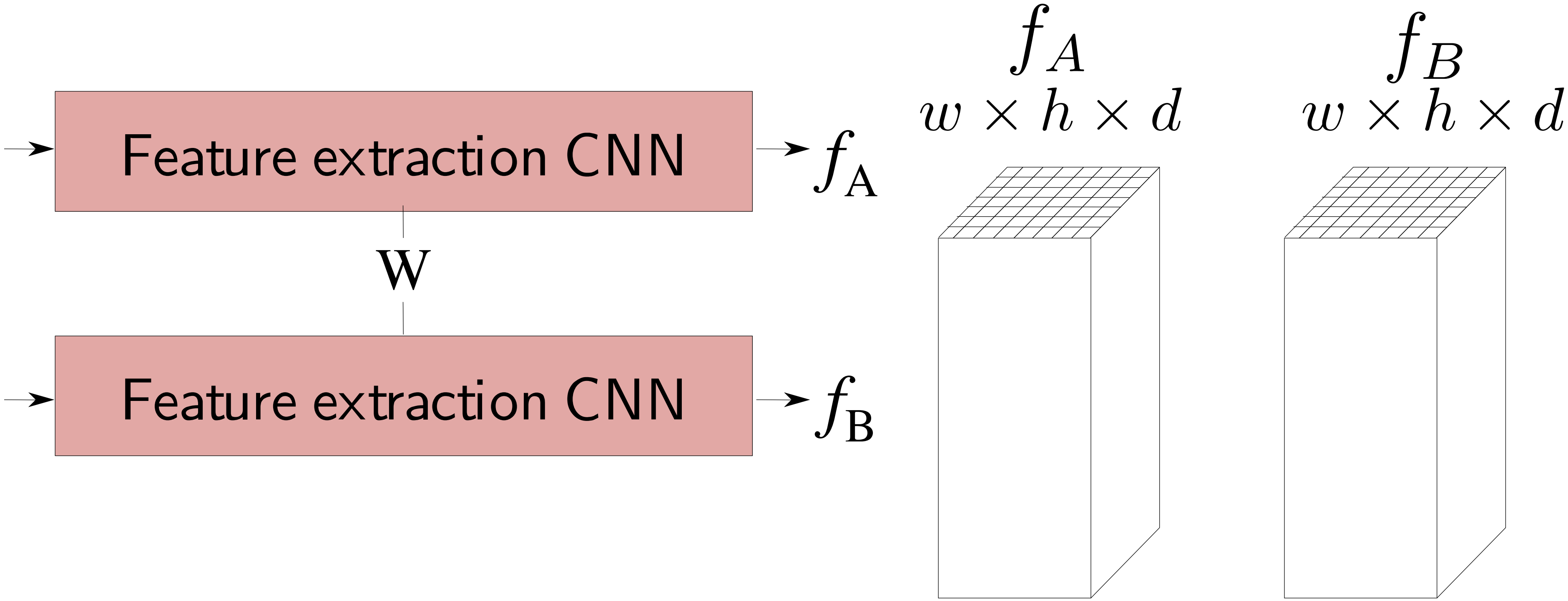


# Proposed approach

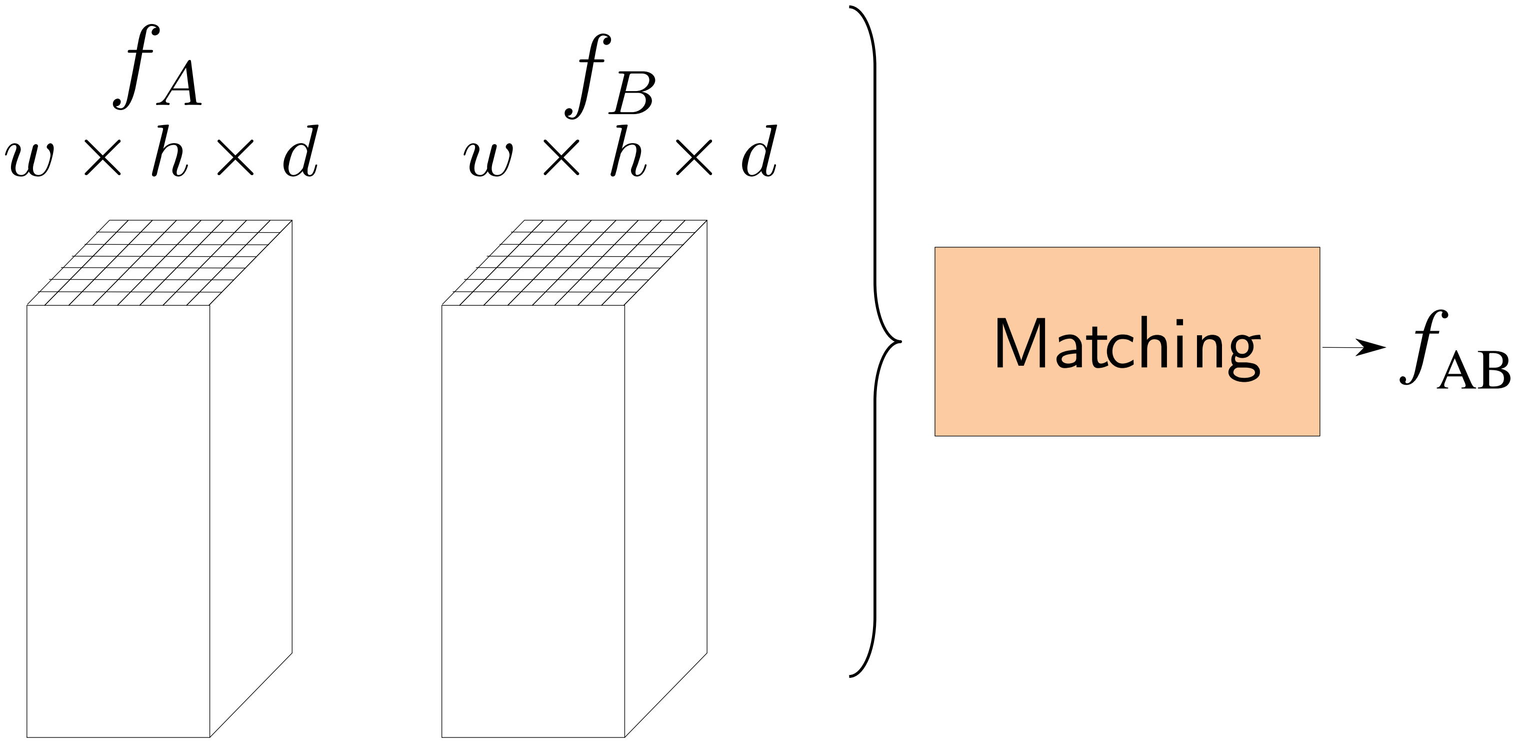



$w \times h$ grids of $d$-dim features

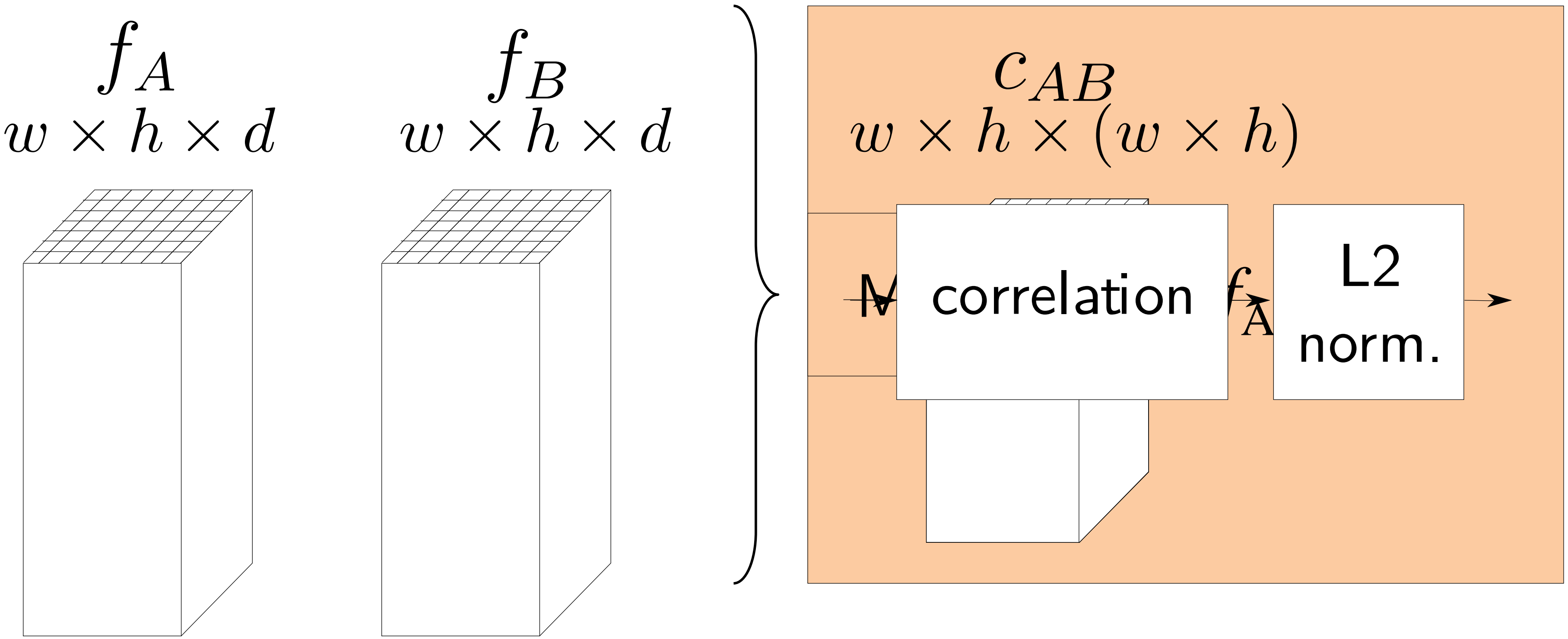
# Proposed approach



# Proposed approach

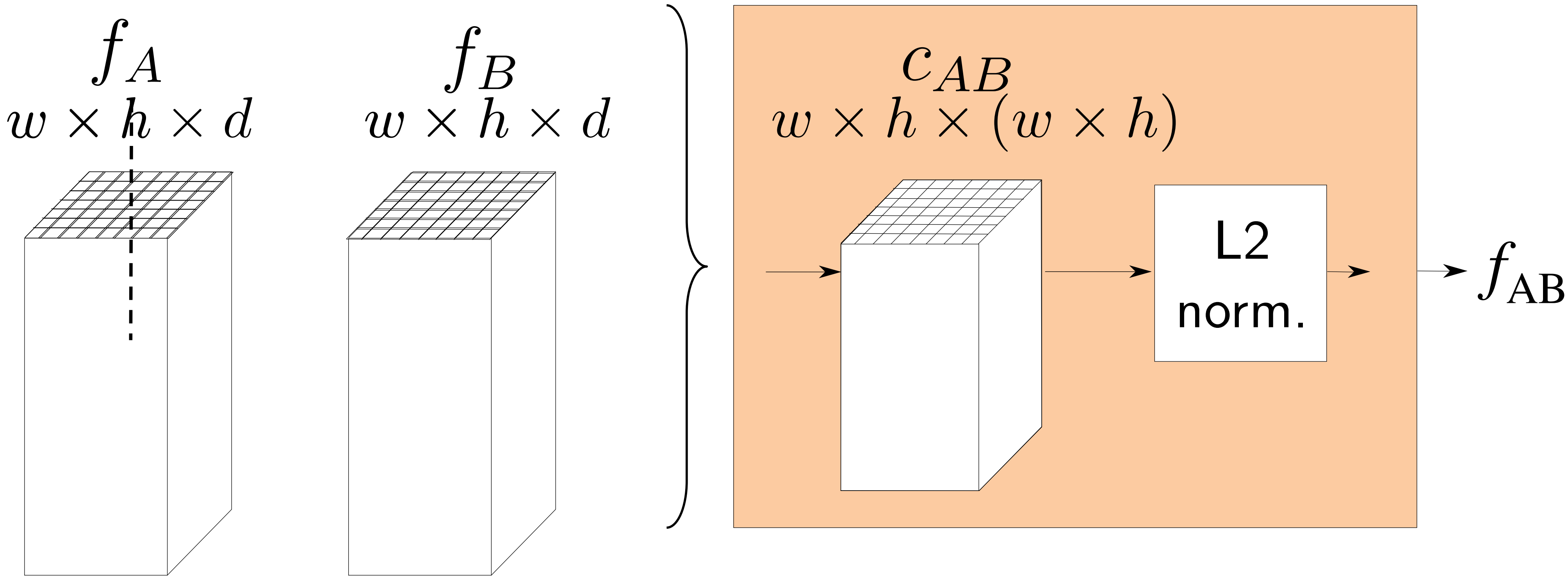


# Proposed approach




similar to [Weinzaepfel et al.'13, Fischer et al '15]

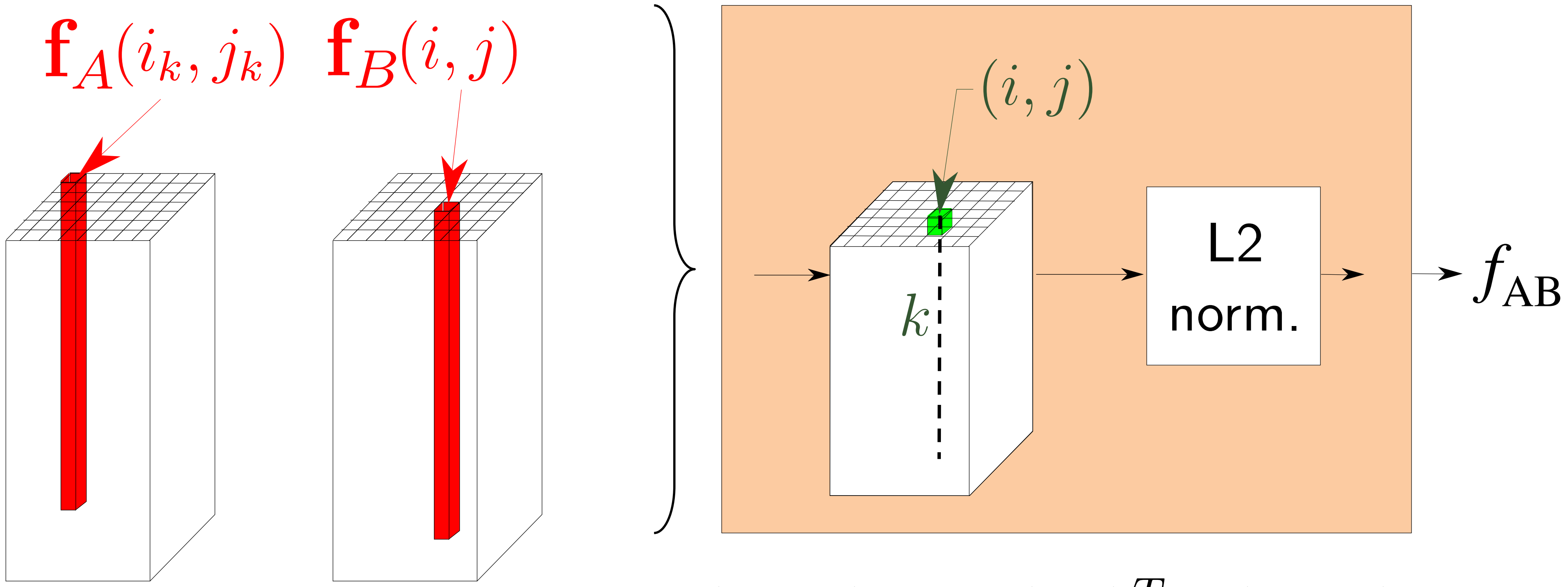
# Proposed approach




Dim. 3: flattened index of $f_B$

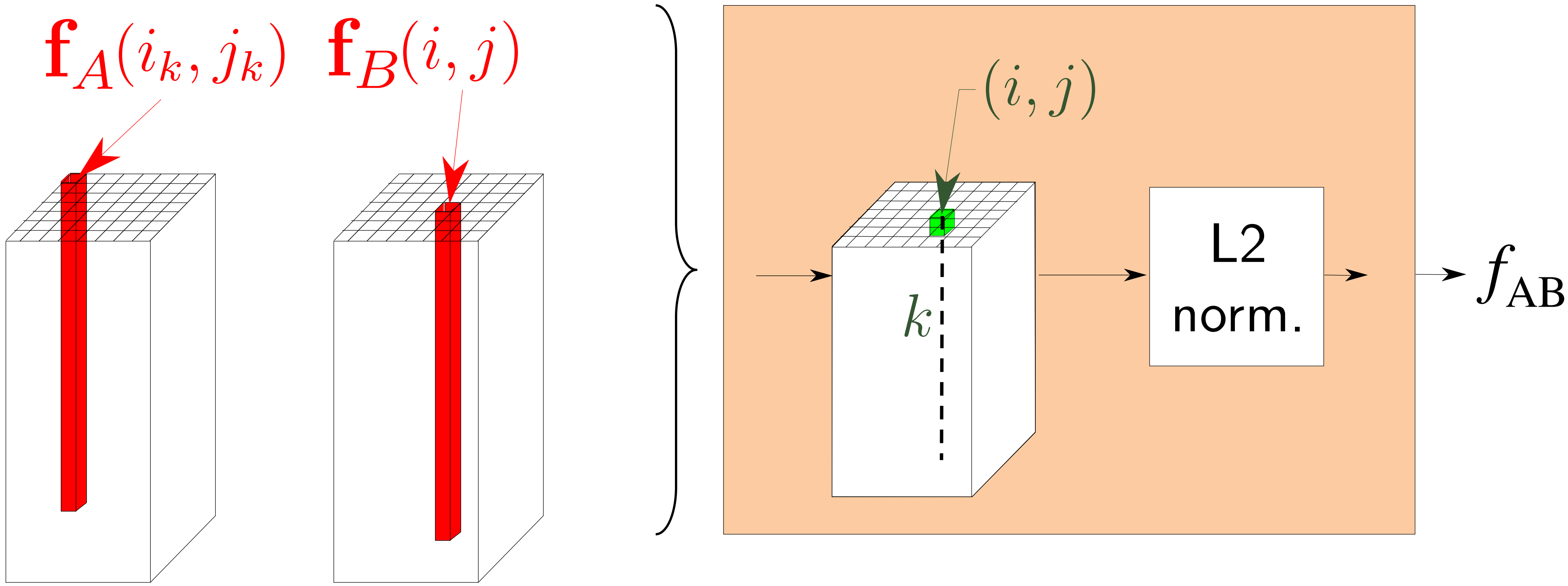
Dim. 1-2: index of $f_A$

# Proposed approach



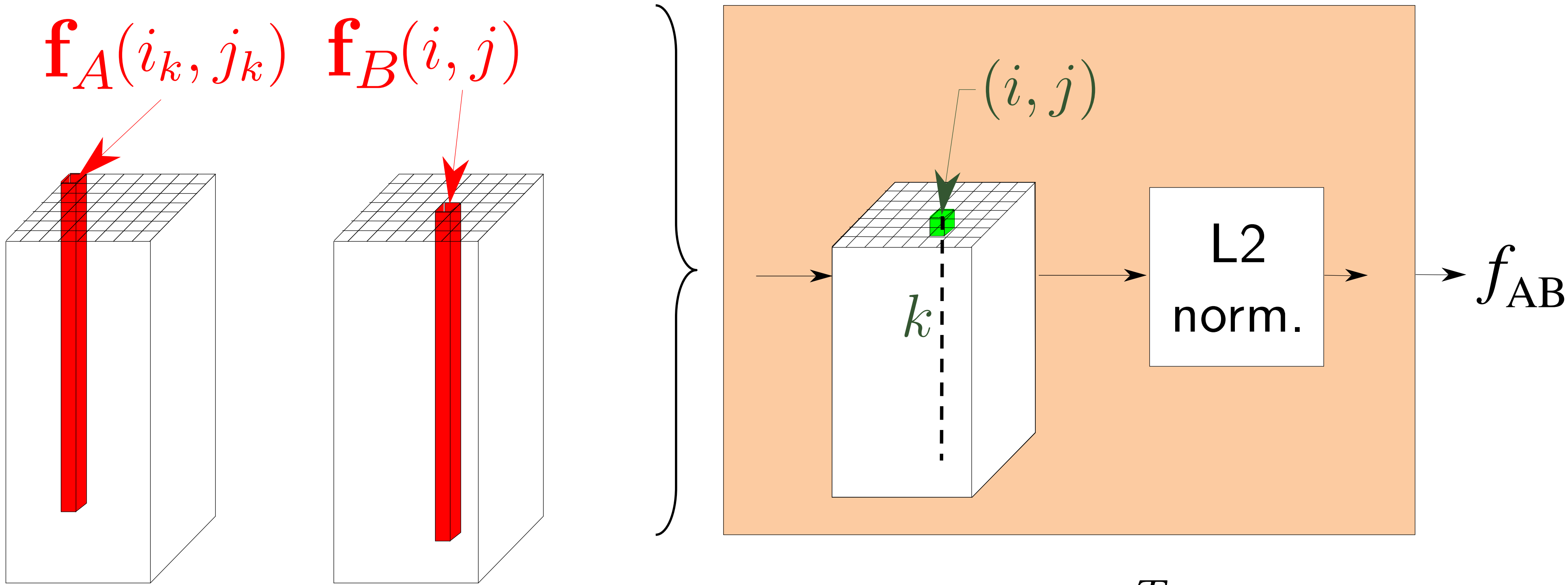

$$c_{AB}(i, j, k) = \mathbf{f}_B(i, j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



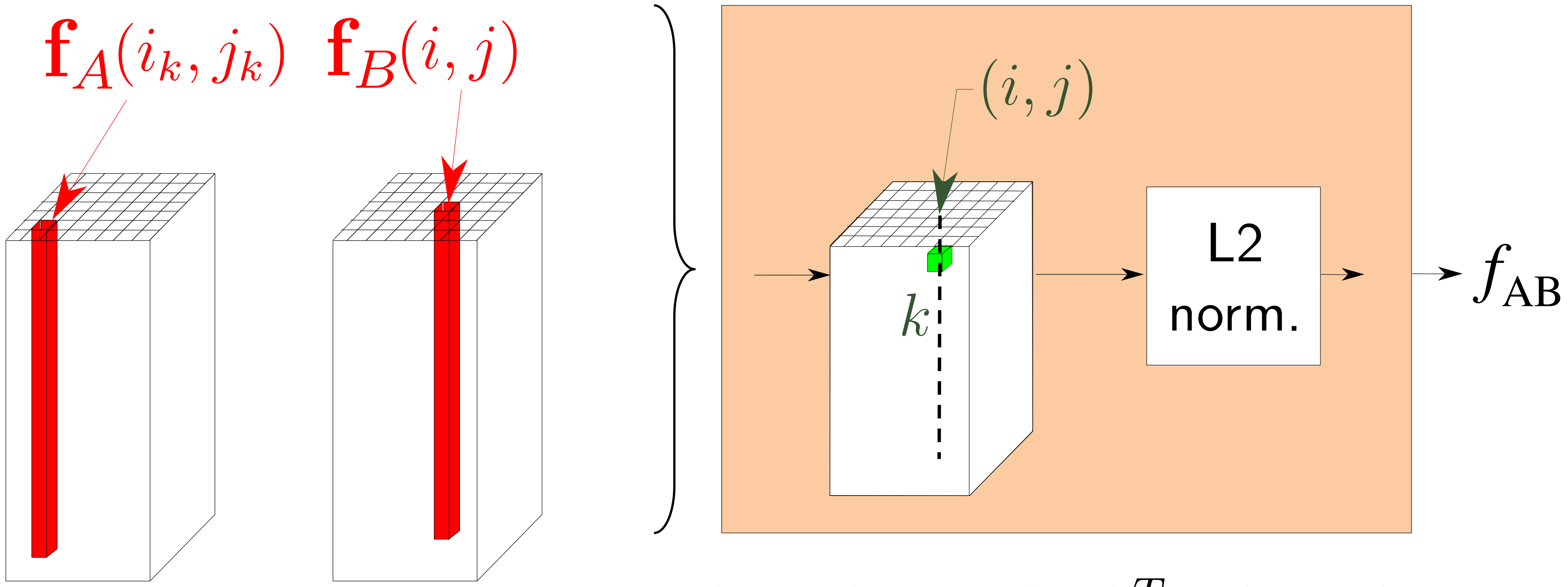

Output consists of similarity scores isolating the feature information

# Proposed approach



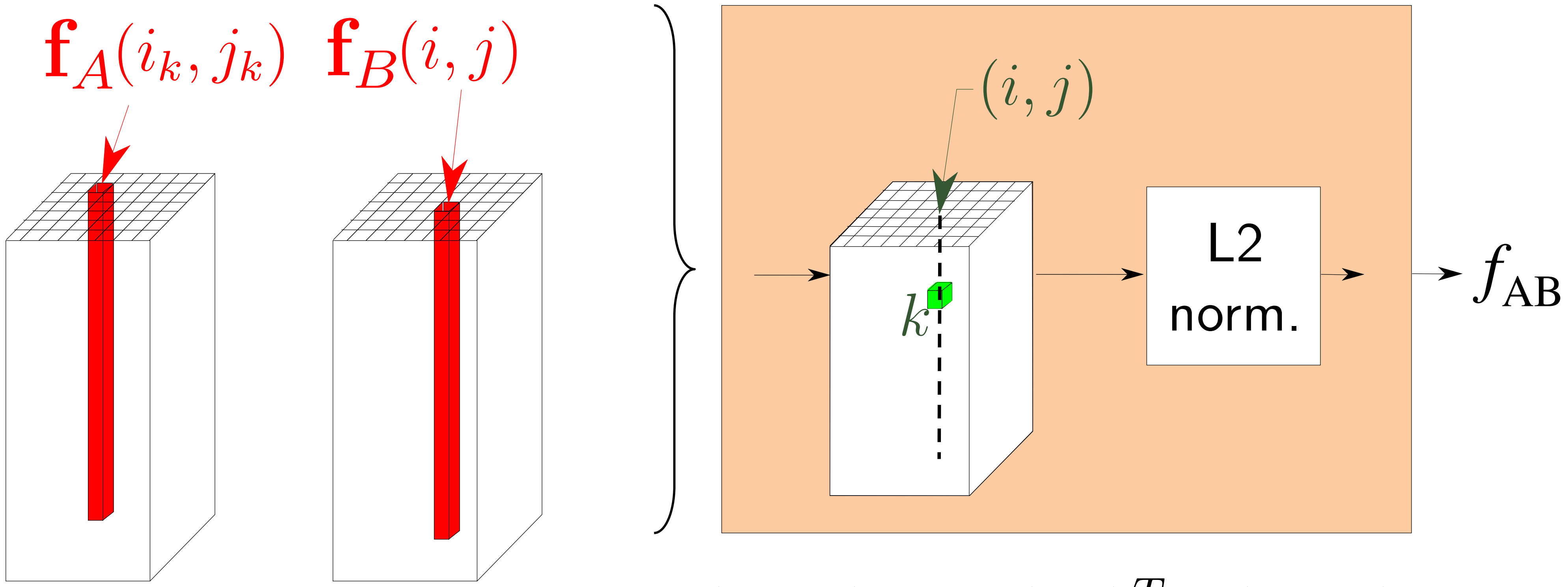

$$c_{AB}(i, j, k) = \mathbf{f}_B(i, j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



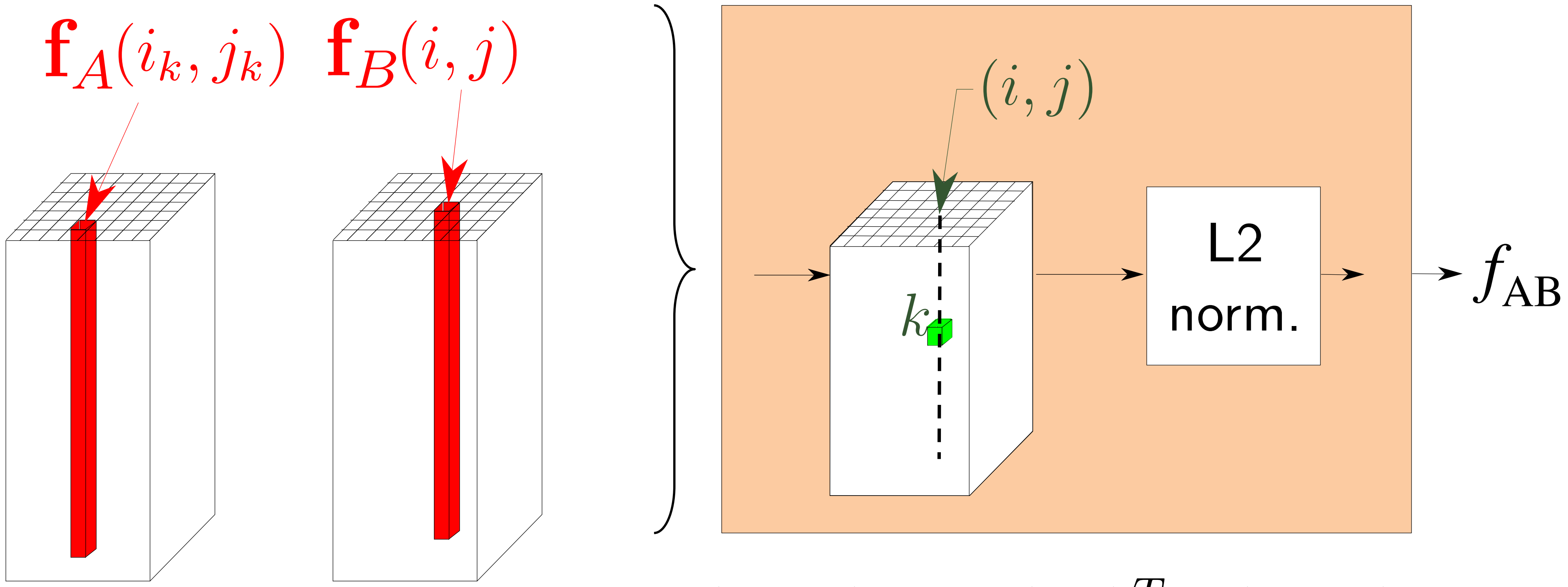

$$c_{AB}(i, j, k) = \mathbf{f}_B(i, j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



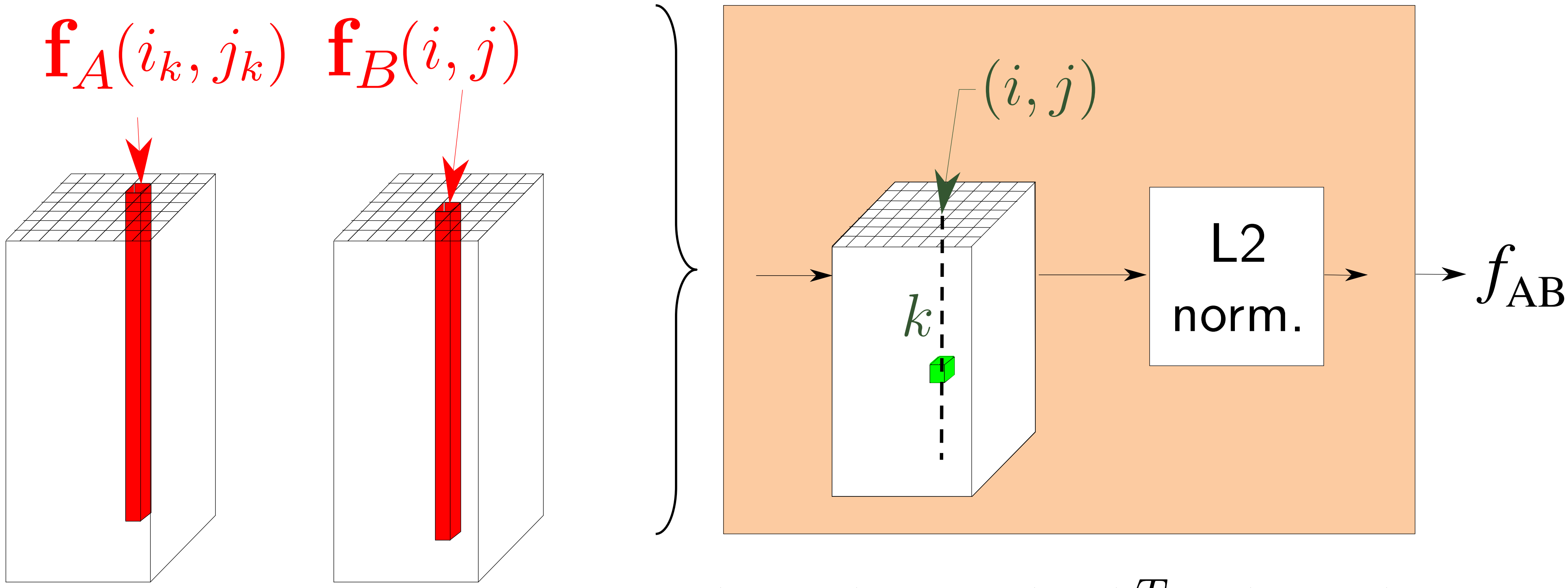

$$c_{AB}(i,j,k) = \mathbf{f}_B(i,j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



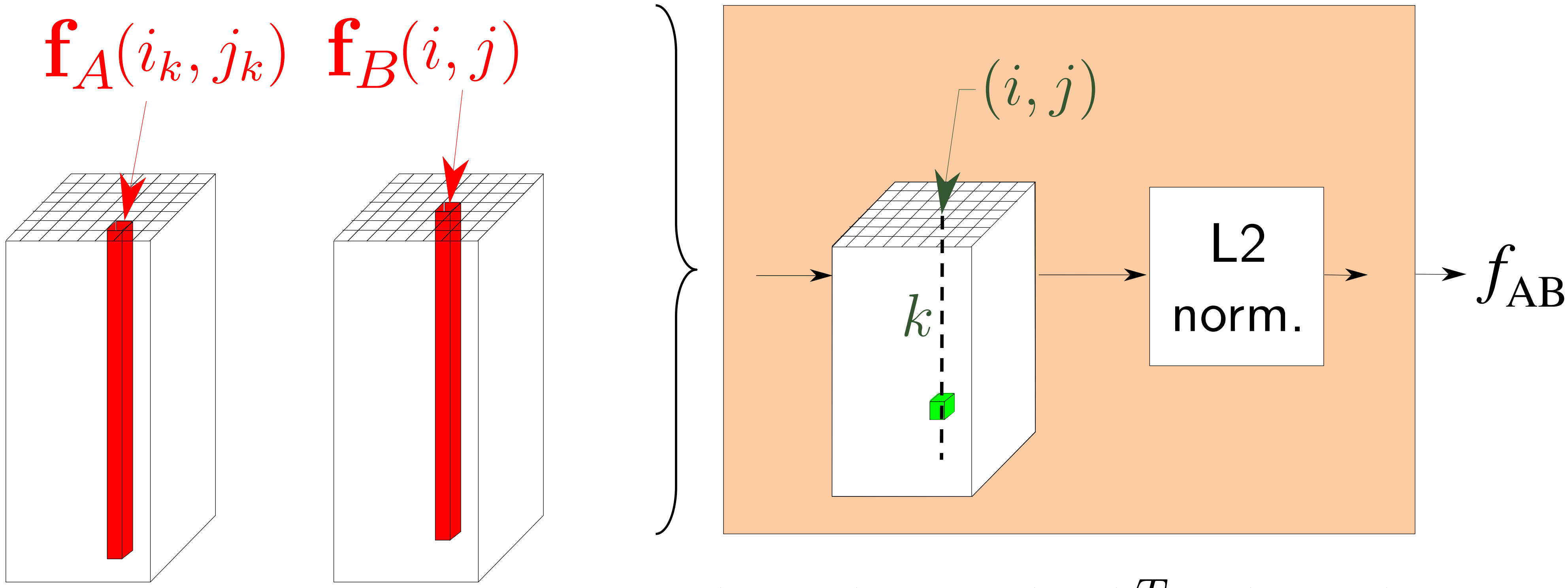

$$c_{AB}(i, j, k) = \mathbf{f}_B(i, j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



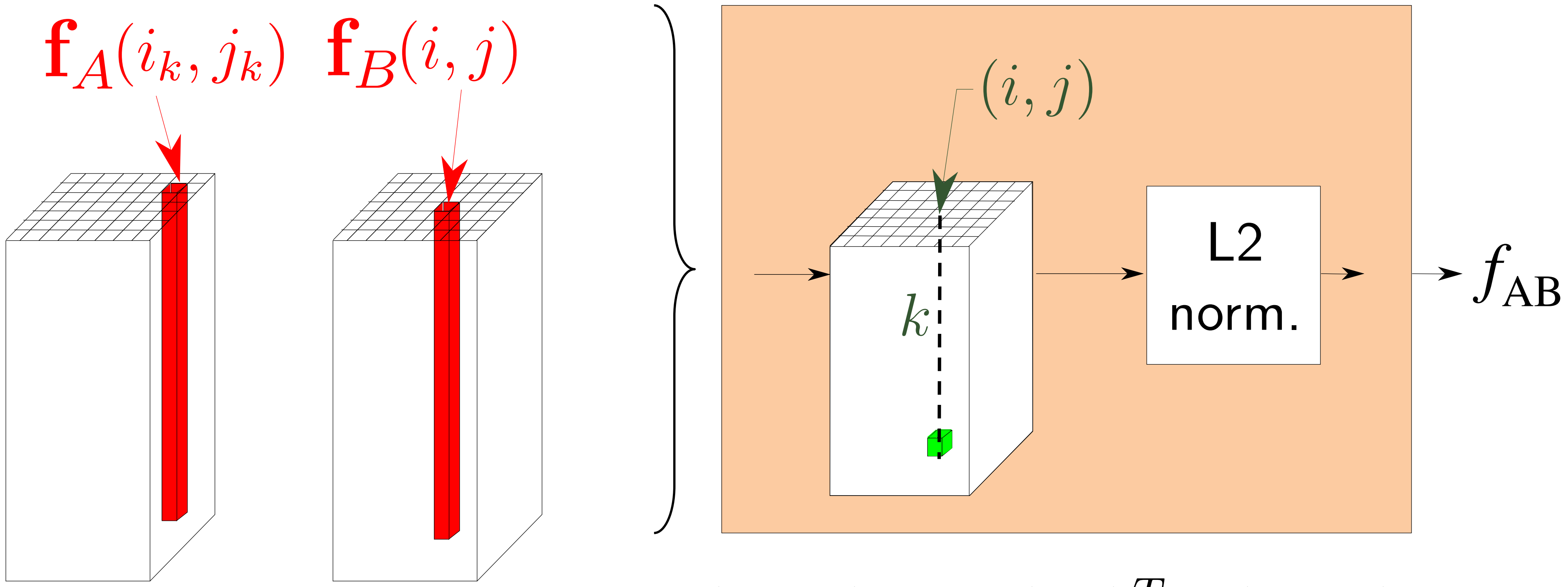

$$c_{AB}(i, j, k) = \mathbf{f}_B(i, j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



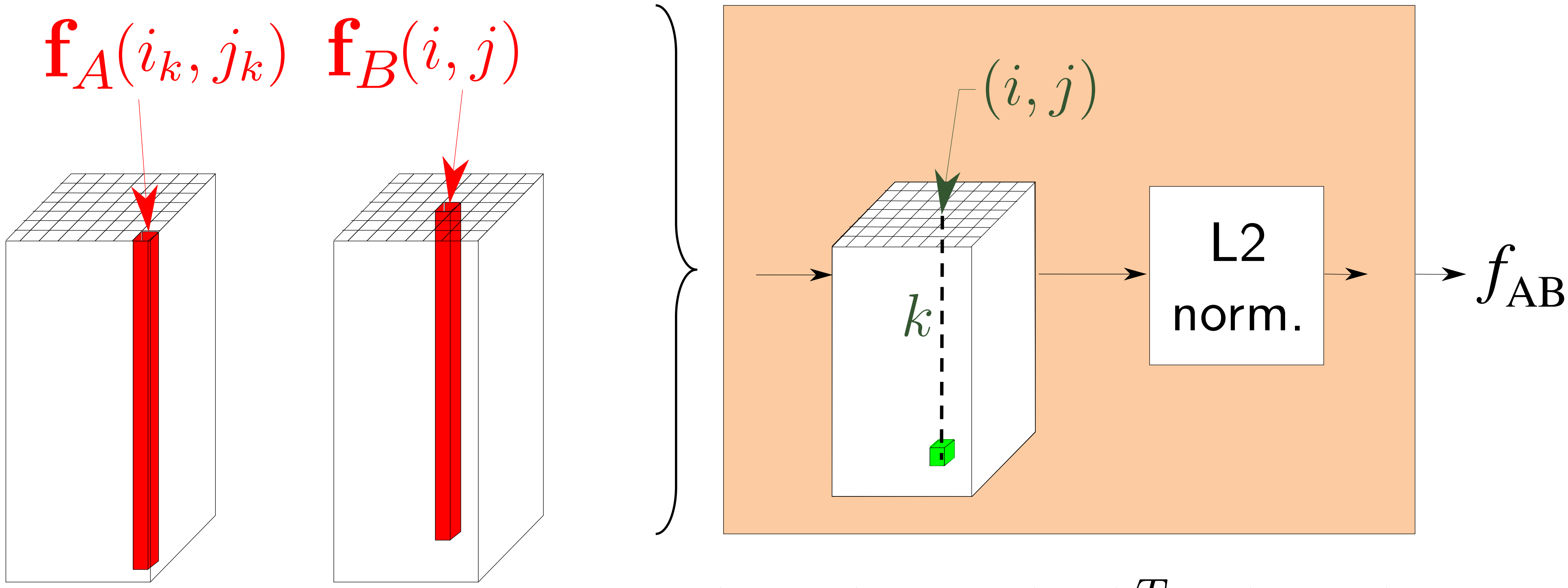

$$c_{AB}(i, j, k) = \mathbf{f}_B(i, j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



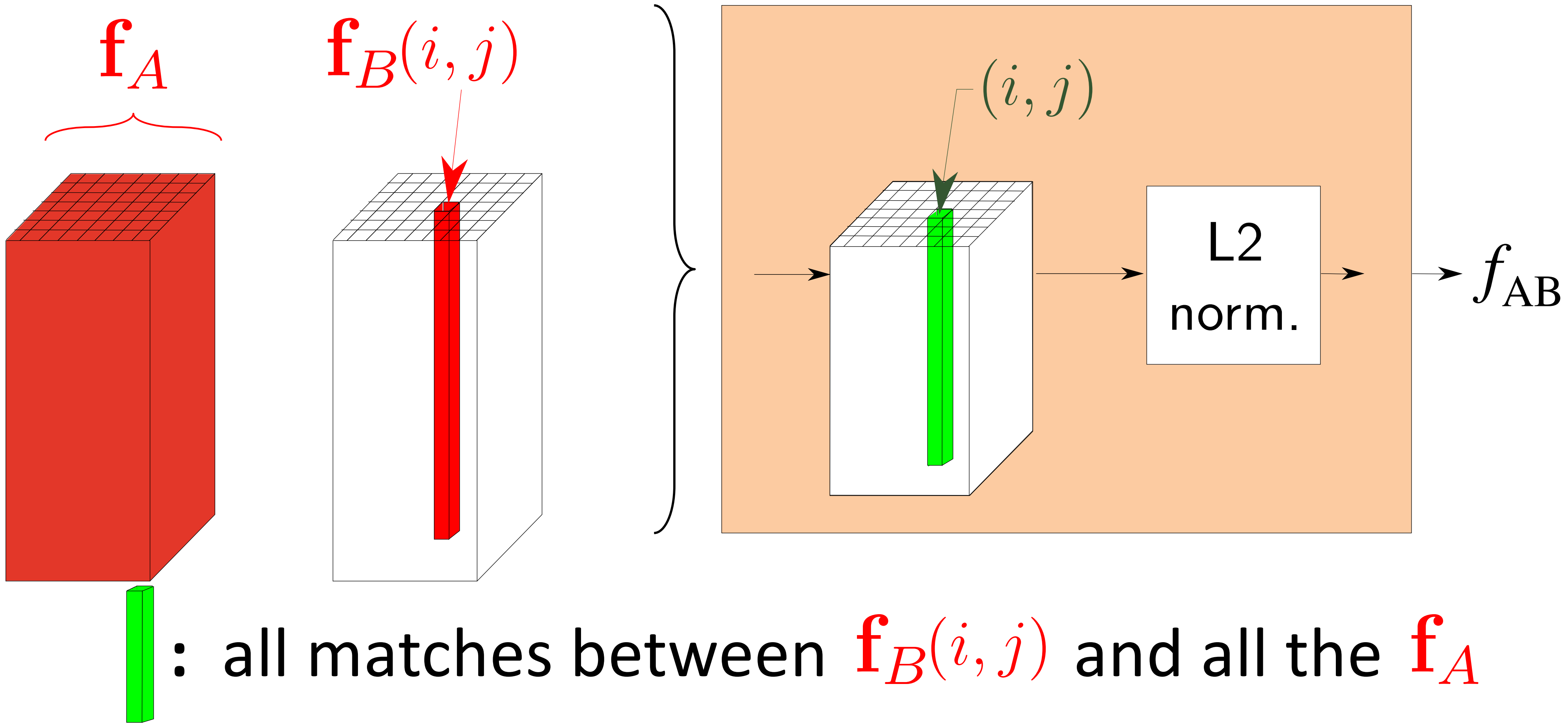

$$c_{AB}(i, j, k) = \mathbf{f}_B(i, j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



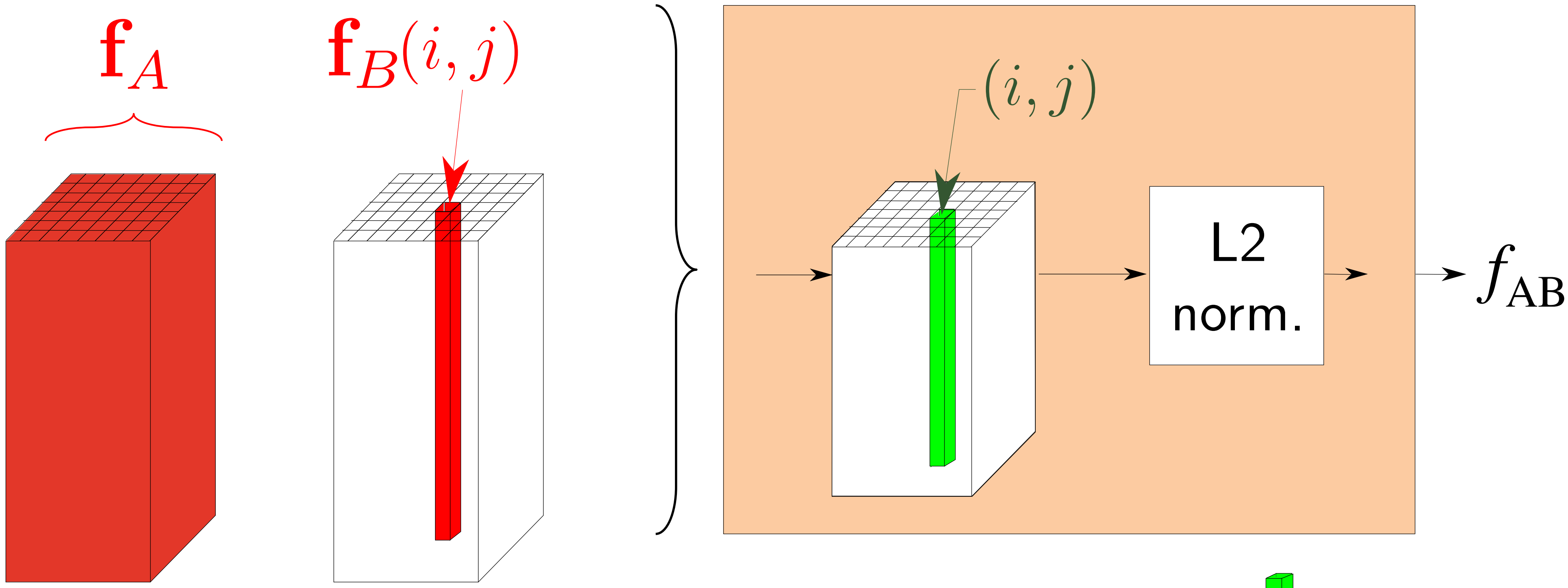

$$c_{AB}(i, j, k) = \mathbf{f}_B(i, j)^T \mathbf{f}_A(i_k, j_k)$$

# Proposed approach



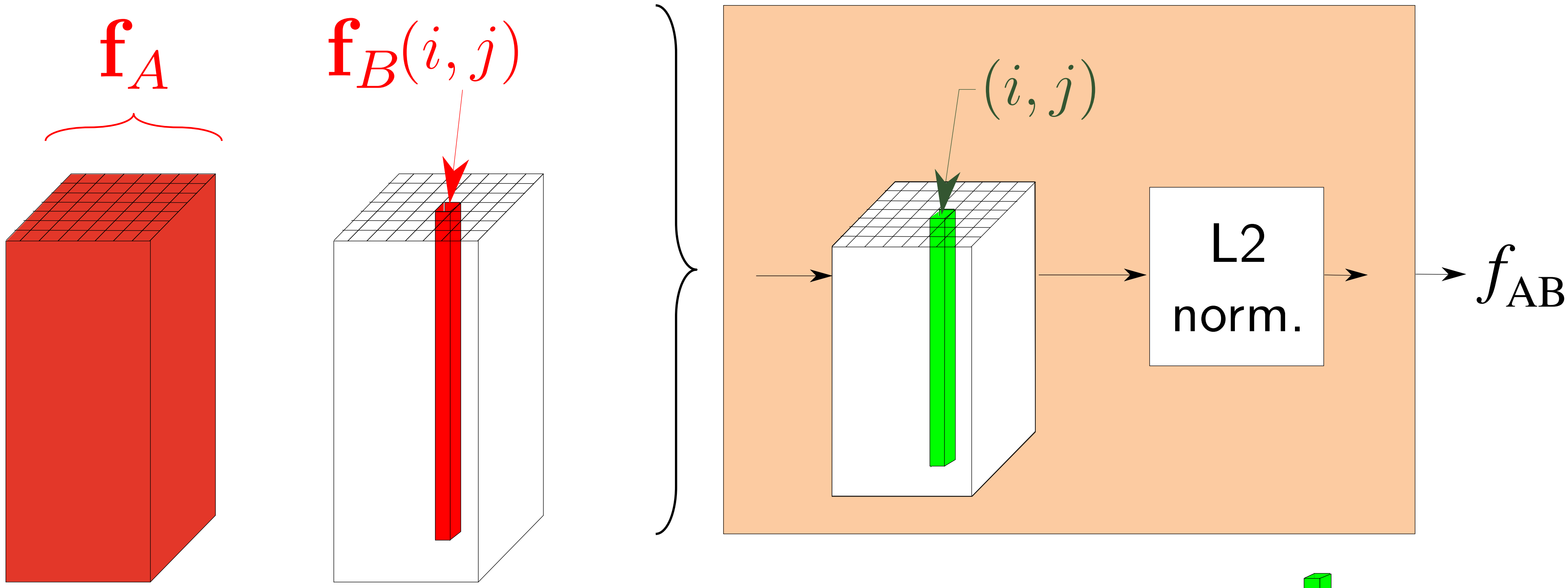

: all matches between $\mathbf{f}_B(i,j)$ and all the $\mathbf{f}_A$

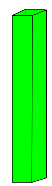
# Proposed approach



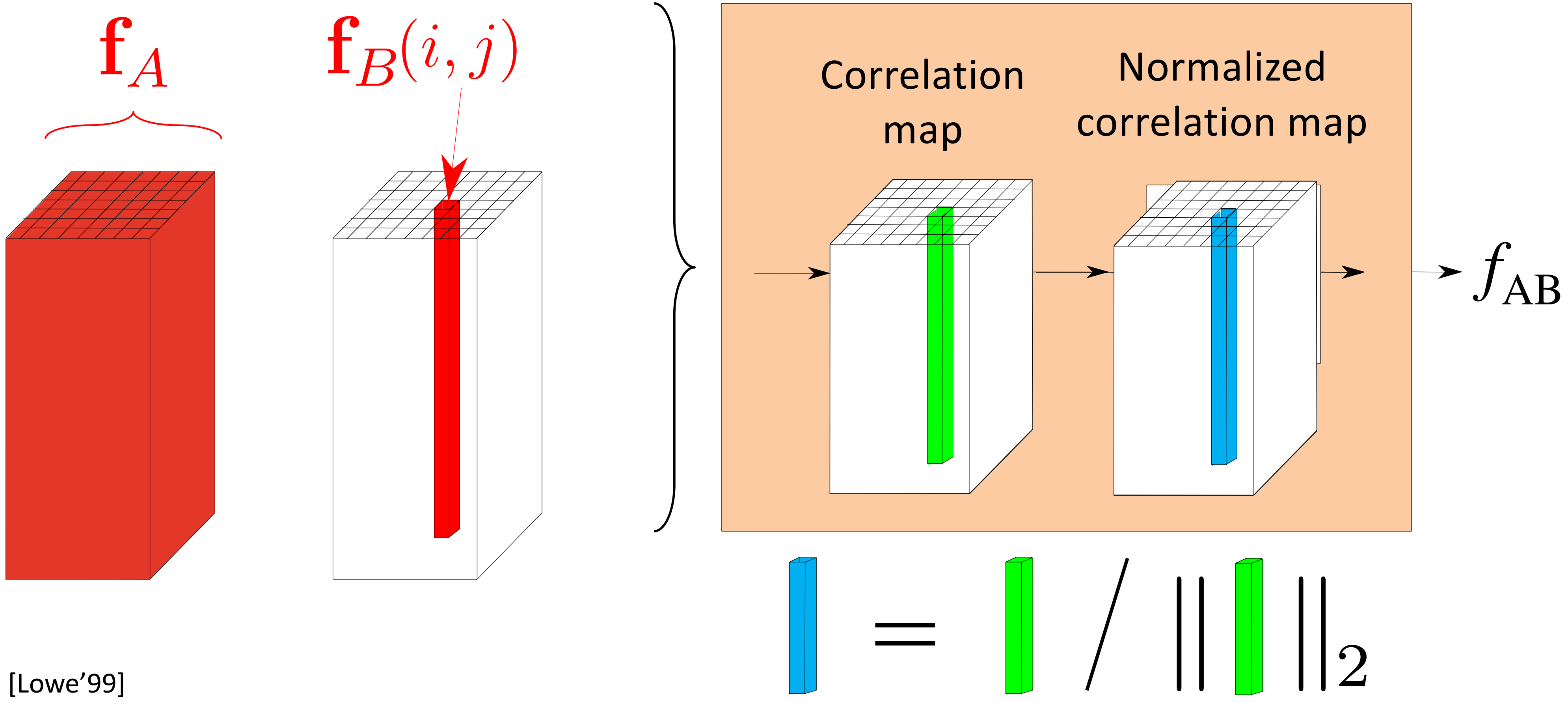

**Ideally:** a single good match along

# Proposed approach



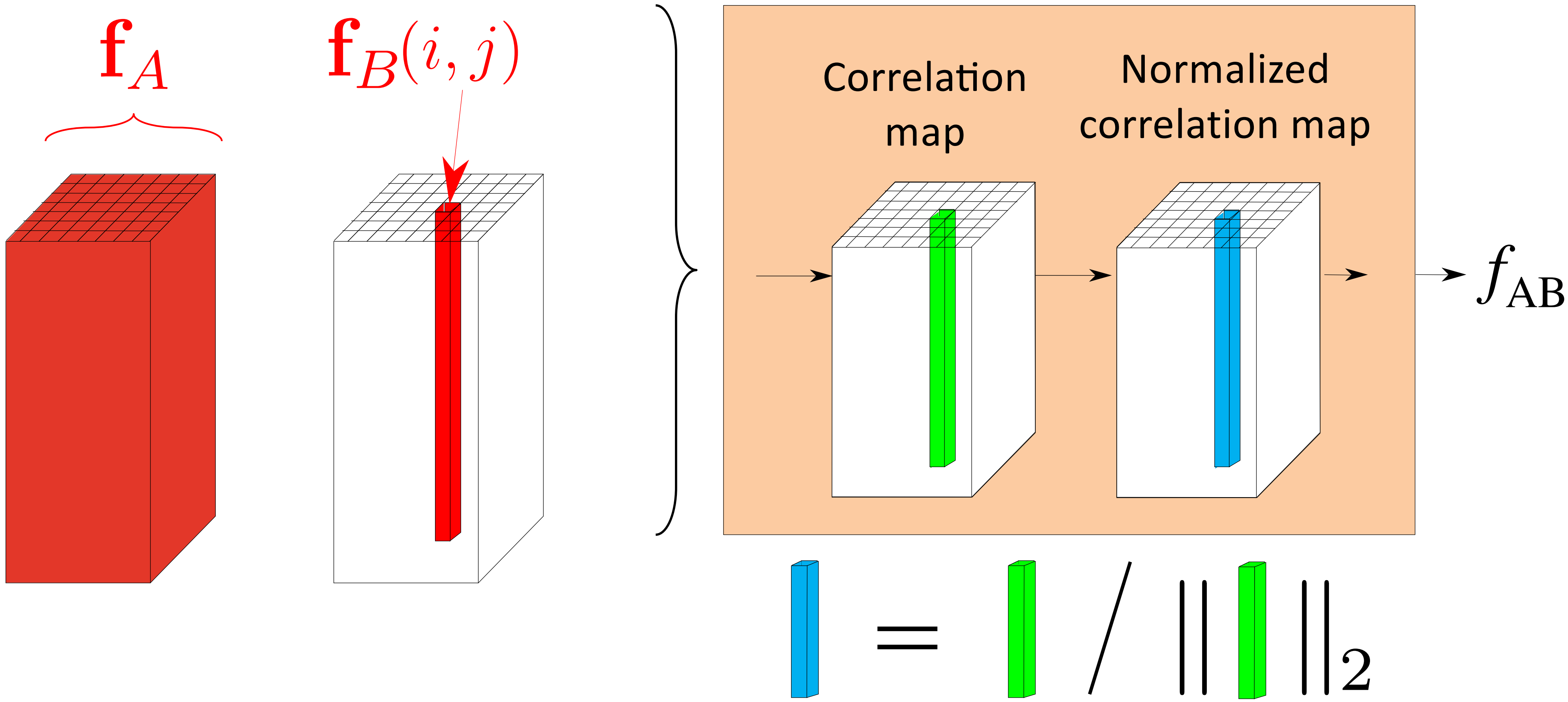

**In practice:** ambiguous matches along 

# Proposed approach

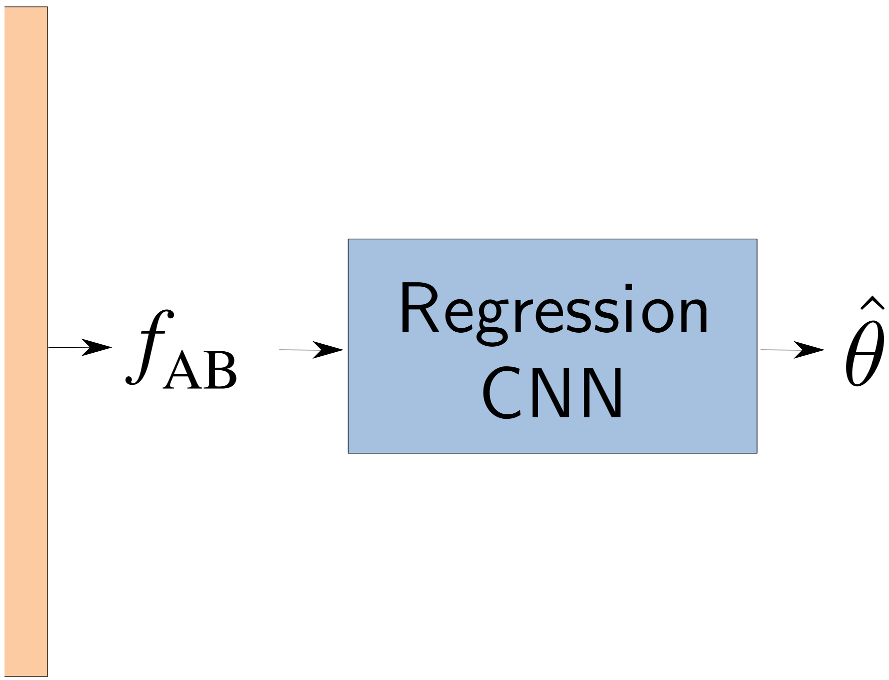



[Lowe'99]

# Proposed approach



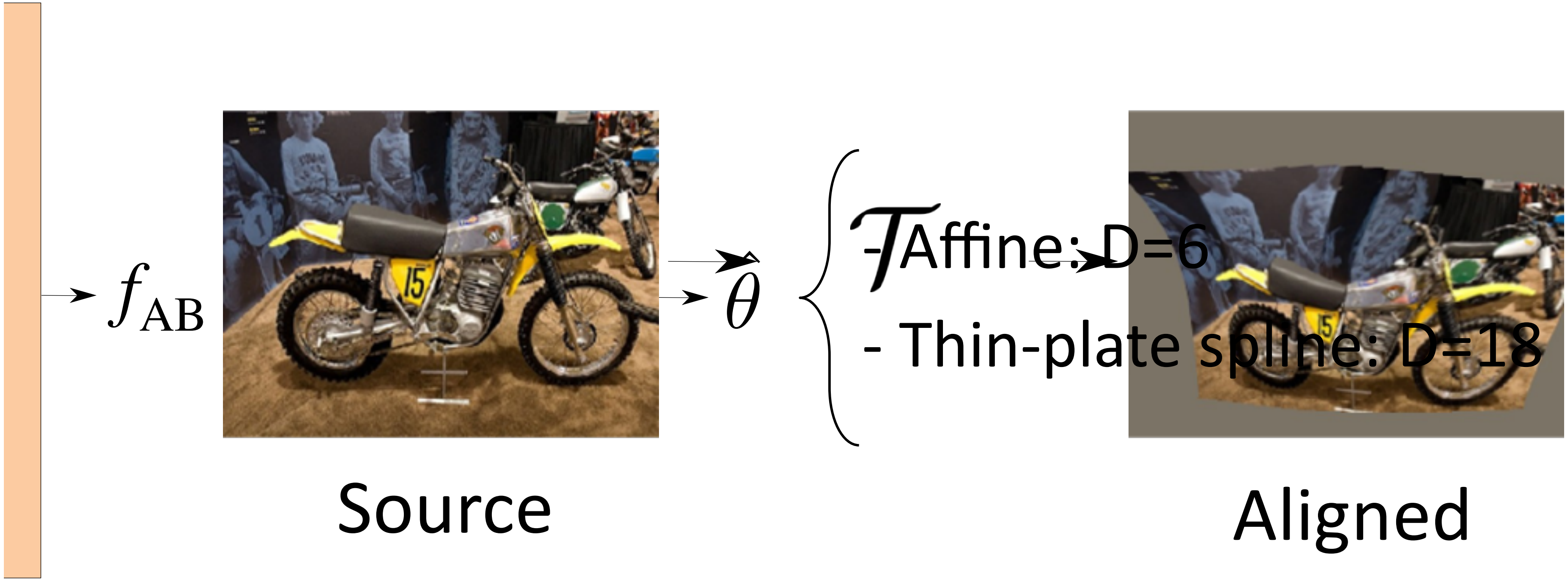
# Proposed approach



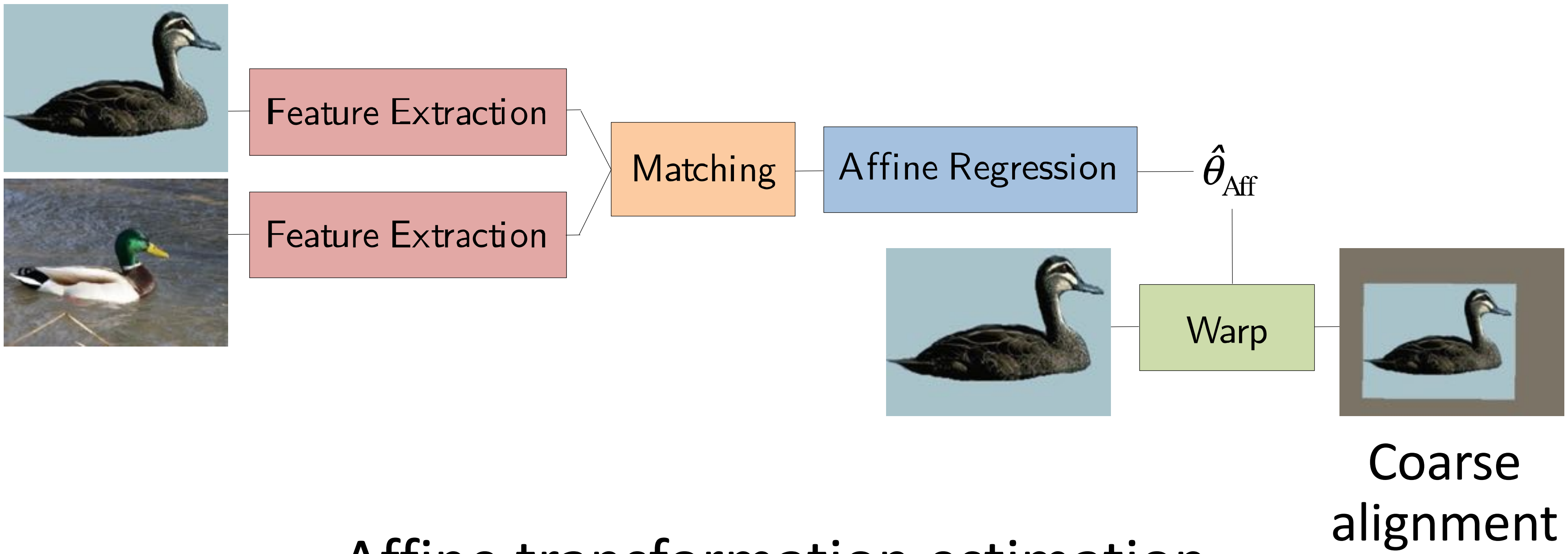

- Affine: D=6
- Thin-plate spline: D=18

$f_{\mathrm{AB}}$ : Scores for all possible feature pairs

# Proposed approach

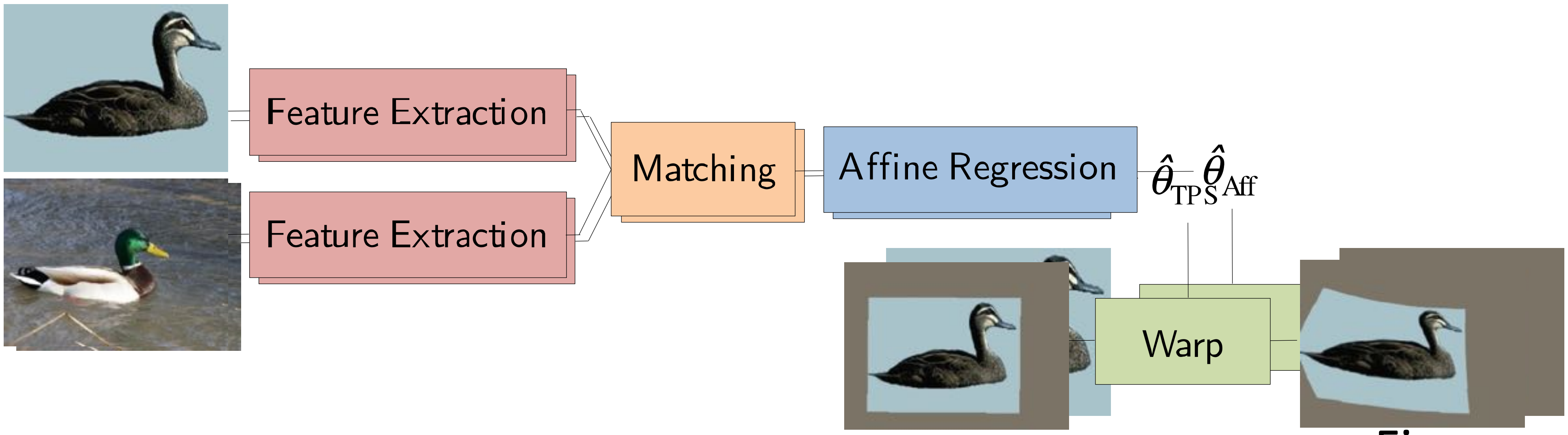


# Coarse to fine architecture




Affine transformation estimation

# Coarse to fine architecture



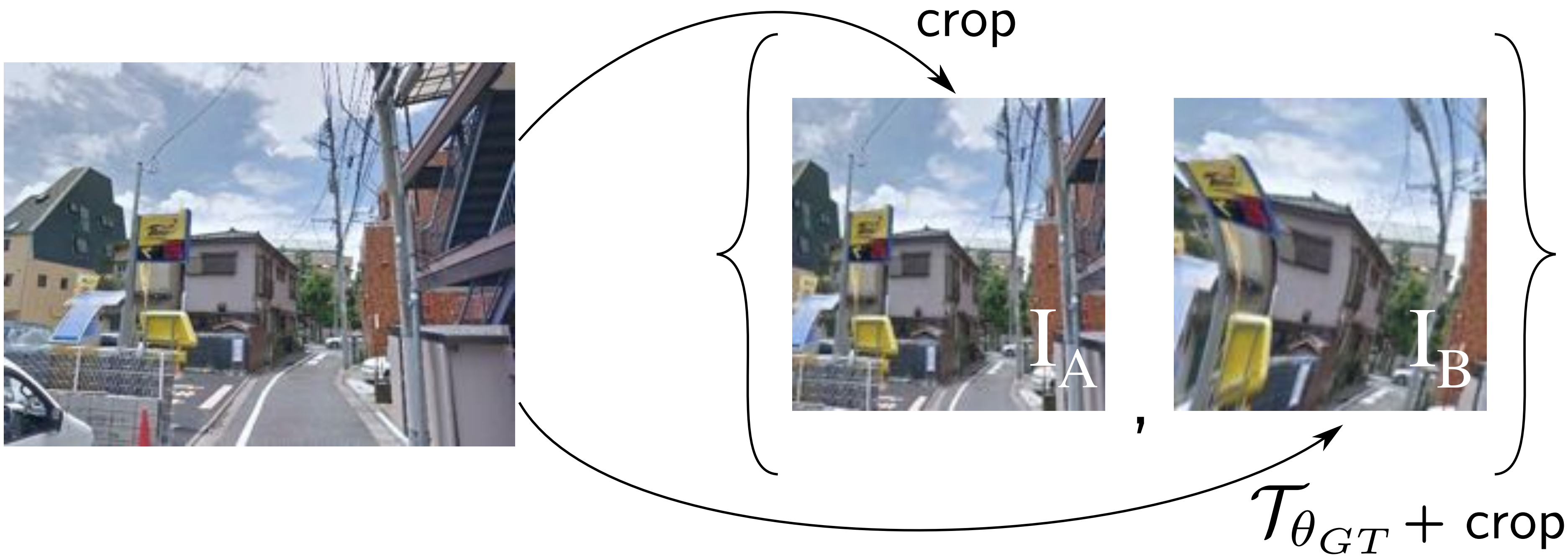

Fine alignment

Thin-plate spline transformation estimation

# Training


Annotating correspondences at a large scale is difficult

# Training

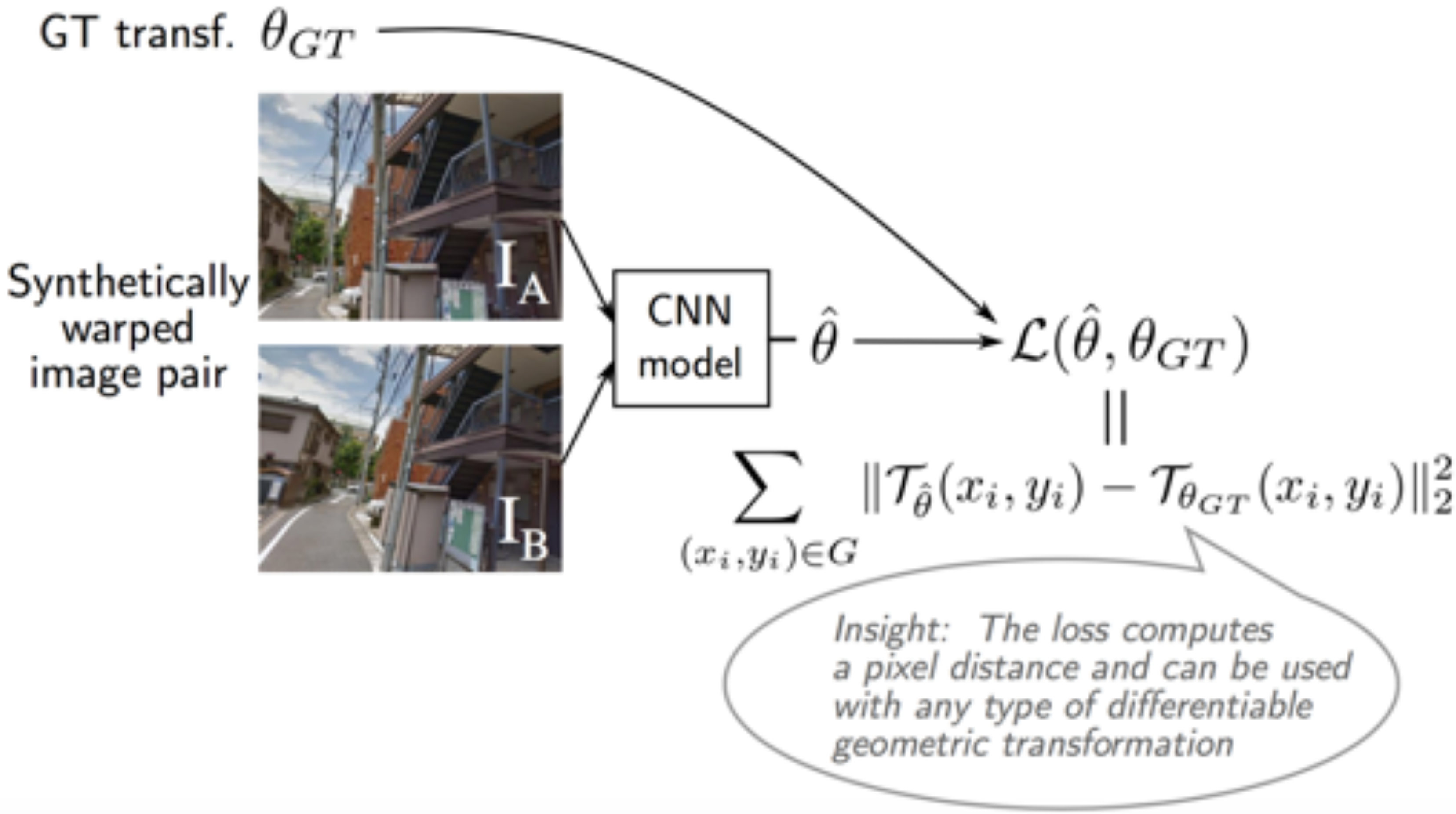



Synthetically generated pairs

Model generalizes to other image content

Tokyo StreetView images from [Arandjelovic et al. '15]

# Training loss



# Results on PF

Source

Target

# Results on PF

Source

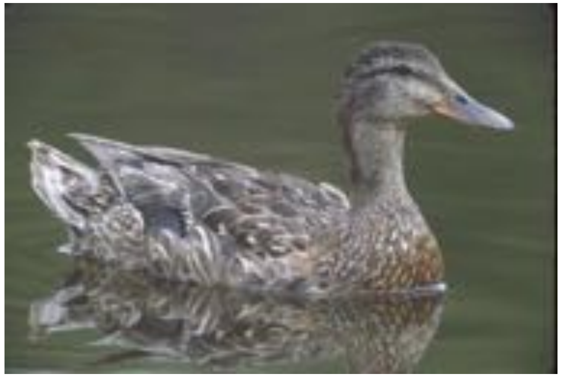
Target

# Results on PF

Source

Target

# Results on PF



Source

Target

# Results on PF

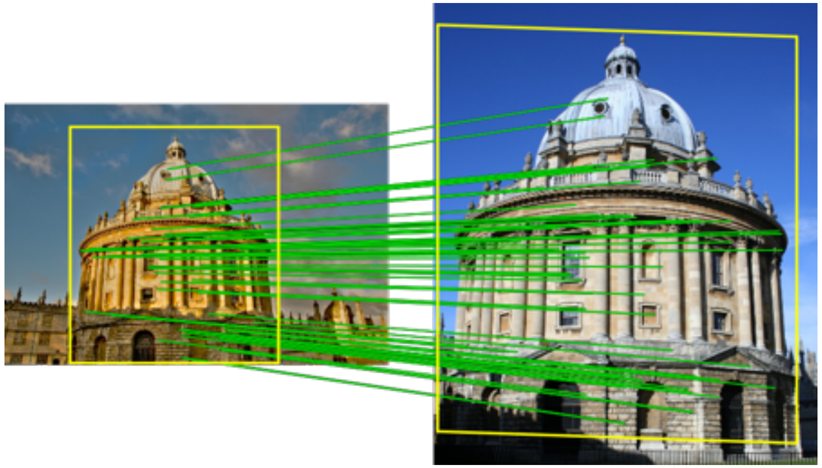
Source

Target

# Results on PF

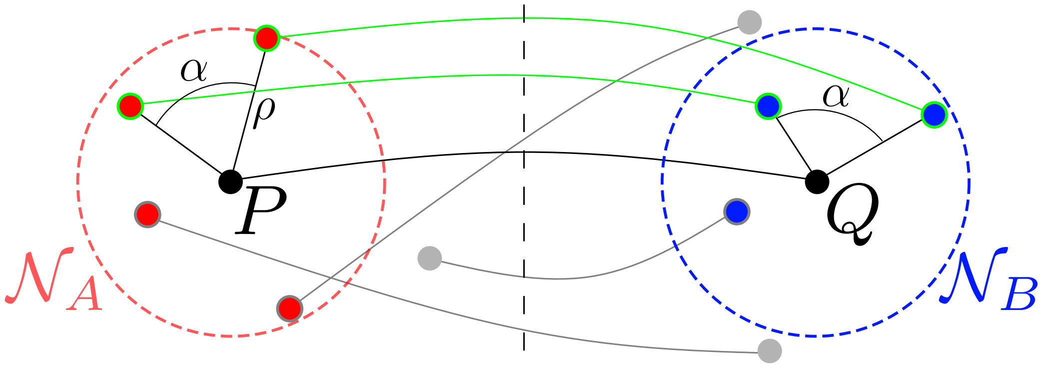
| Methods | PCK (%) |
|---|---|
| DeepFlow [43] | 20 |
| GMK [15] | 27 |
| SIFT Flow [37] | 38 |
| DSP [31] | 29 |
| Proposal Flow NAM [23] | 53 |
| Proposal Flow PHM [23] | 55 |
| Proposal Flow LOM [23] | 56 |
| RANSAC with our features (affine) | 47 |
| Ours (affine) | 49 |
| Ours (affine + thin-plate spline) | 56 |
| Ours (affine ensemble + thin-plate spline) | **57** |

# Do we need global geometric model?



Global 2D affine transformation

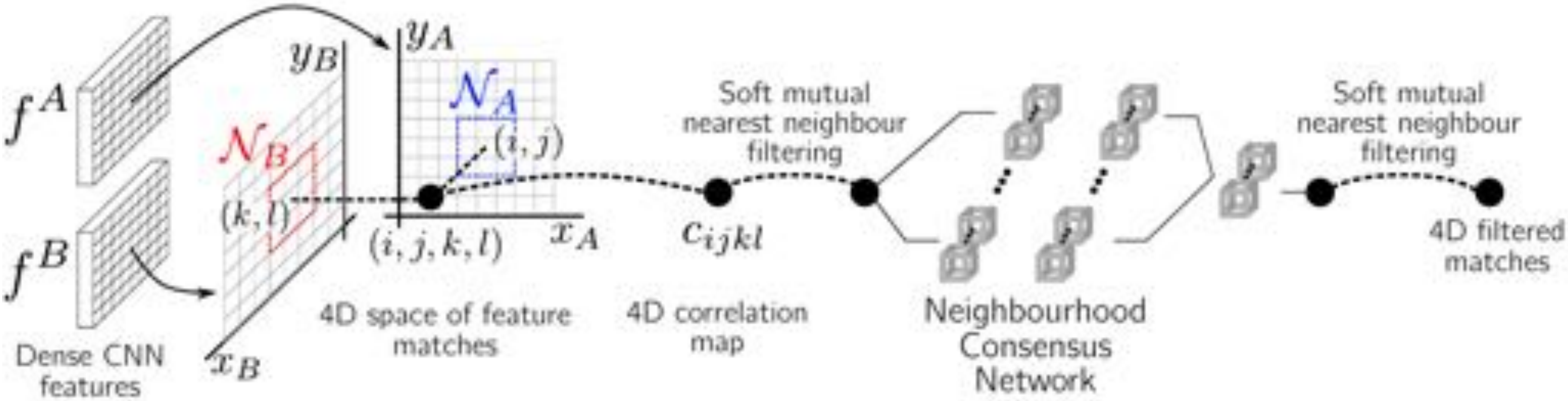
[Hartley&Zisserman'04, Lazebnik et al.03, Philbin et al.,'17, … ]




Semi-local constraints

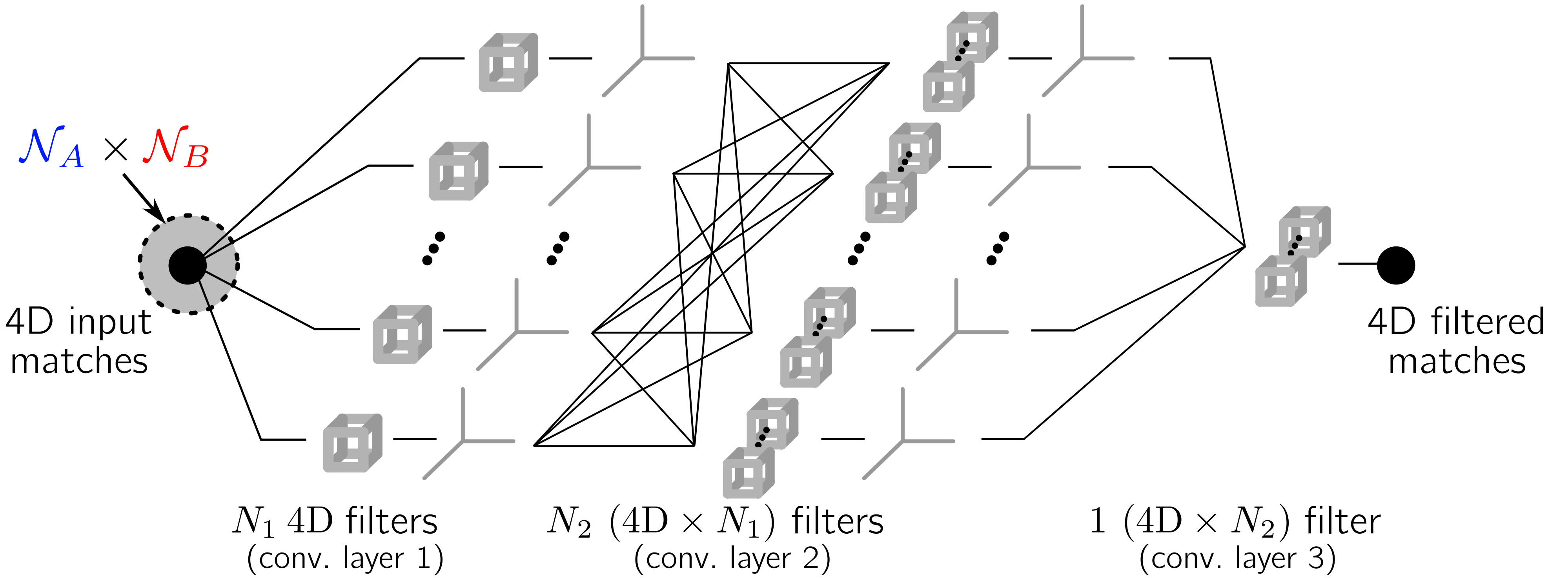
[Ferrari et al.'05, Schaffalitzky and Zisserman'02, Schmid and Mohr'97, Sivic and Zisserman'03, Zhang et al.'95, Bian et al'17, …]

# Neighborhood consensus networks



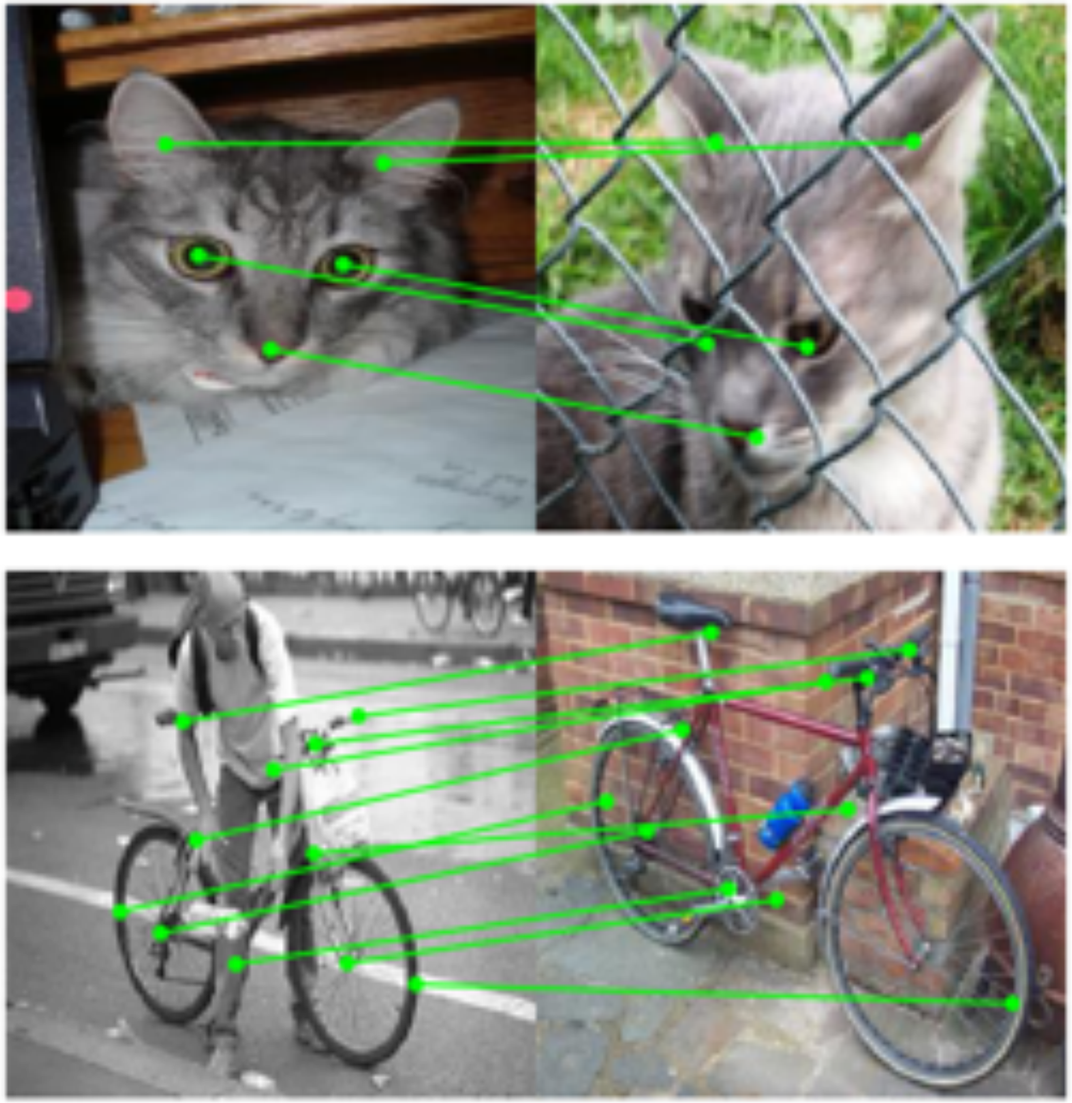

[Rocco et al., NIPS 2018]

# Neighborhood consensus networks



# Results: PF-Pascal dataset

| Method | PCK ($\alpha = 0.1$) |
|---|---|
| HOG+PF-LOM [8] | 62.5 |
| SCNet-AG+ [9] | 72.2 |
| CNNGeo [20] | 71.9 |
| WeakAlign [21] | 75.8 |
| NC-Net | **78.9** |



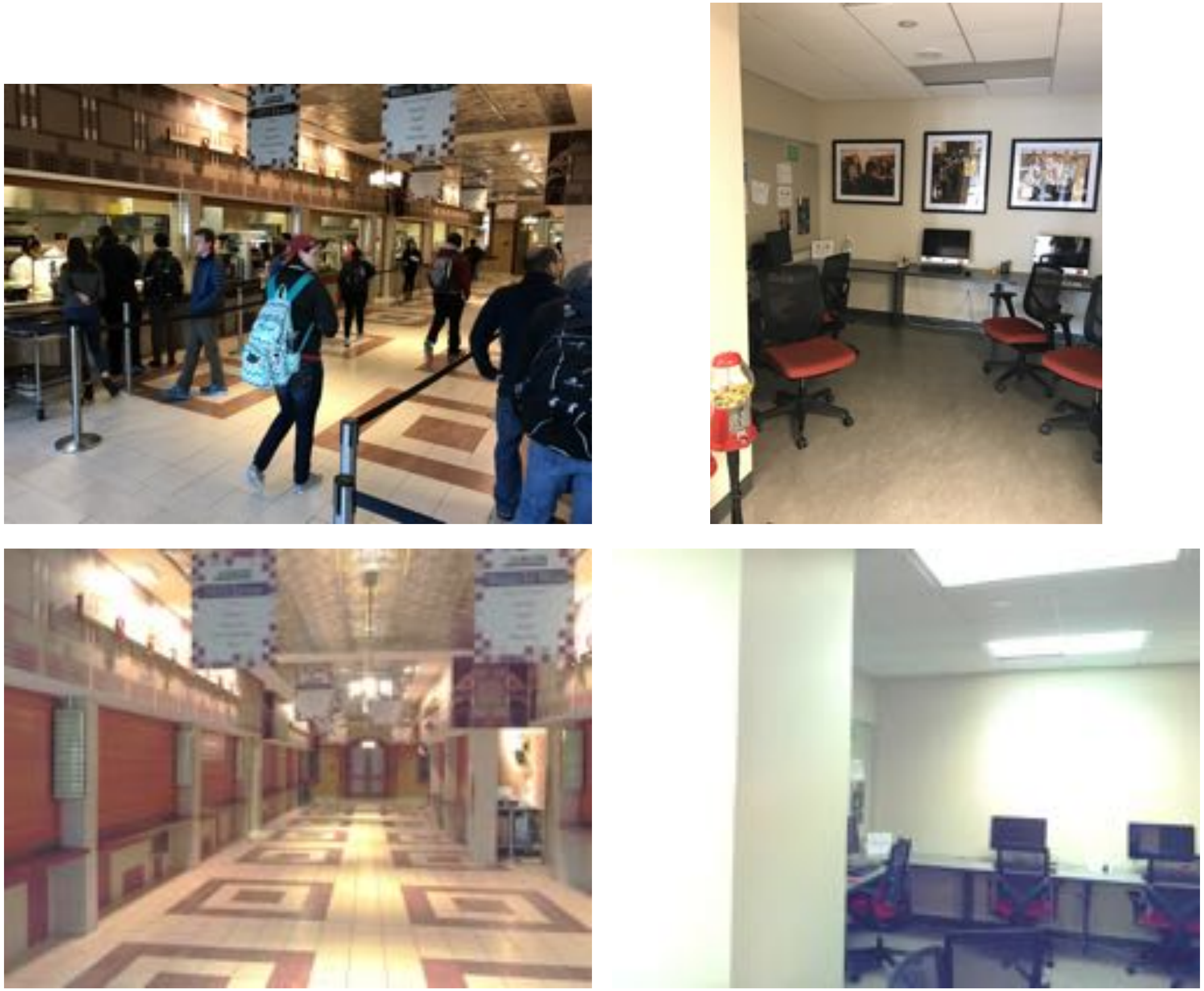
# Results: Indoor localization

Plug into localization pipeline of [Taira et al., CVPR'18]

| Distance (m) | SparsePE [31] | DensePE [31] | DensePE + **NC-Net** | InLoc [31] | InLoc + **NC-Net** |
|---|---|---|---|---|---|
| 0.25 | 21.3 | 35.3 | 34.7 | 38.9 | **41.0** |
| 0.50 | 30.7 | 47.4 | 50.8 | 56.5 | **59.0** |
| 1.00 | 42.6 | 57.1 | 60.2 | 69.9 | **71.4** |
| 2.00 | 47.1 | 61.1 | 64.7 | 74.2 | **77.8** |

# Visual localization indoors

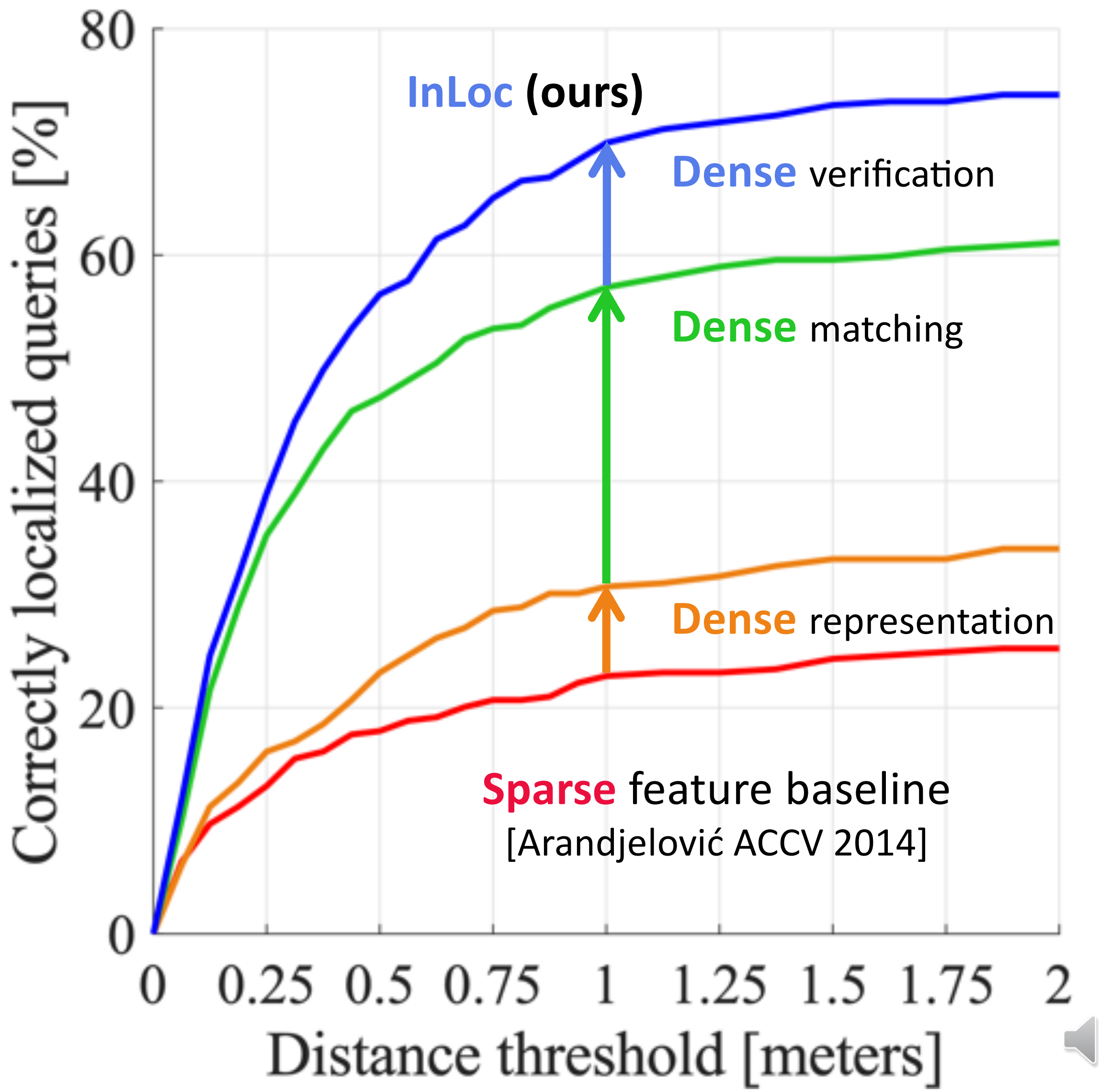
[Taira et al., CVPR 2018]



# Evaluation

□ **InLoc dataset**

- **10K** DB images, **23,000m²**
- **329** test images with reference poses



# Example: Visual localization in changing conditions

[Sattler et al., CVPR 2018]

# Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions



**Torsten Sattler**

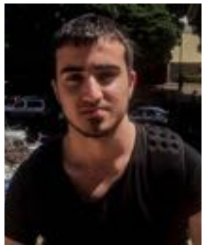

Will Maddern


Carl Toft


Akihiko Torii


Lars Hammarstrand


Erik Stenborg


Daniel Safari


Masatoshi Okutomi


Marc Pollefeys

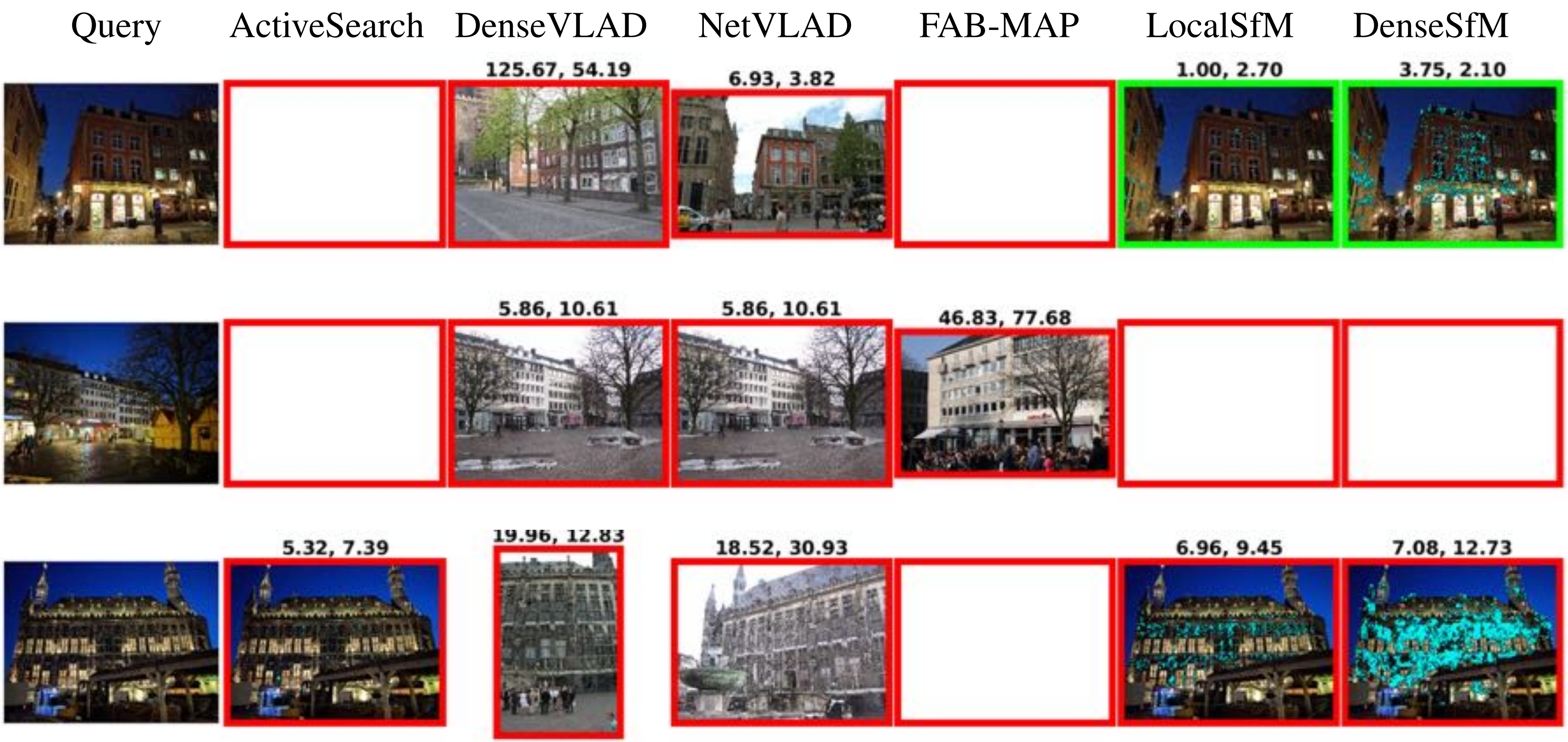

Josef Sivic


Fredrik Kahl


Tomas Pajdla

ETH zürich · UNIVERSITY OF OXFORD · CHALMERS UNIVERSITY OF TECHNOLOGY · Microsoft · DTU · Inria · CIIRC · LUND UNIVERSITY


2018 CVPR SALT LAKE CITY • JUNE 18-22




What is the right representation for **visual localization and navigation**?
- changing conditions, outdoor/indoor, generalization to new environments.

# Next challenge : Embodied computer vision

**Problems**:

1. Can we localize large-scale changing environments?

2. Can we learn to navigate in never seen before places?

3. How can we transfer these capabilities to a real robot?

4. How to learn to communicate with people about visually grounded concepts (spaces, directions, objects)?

5. Can we learn these capabilities without direction input/output supervision?

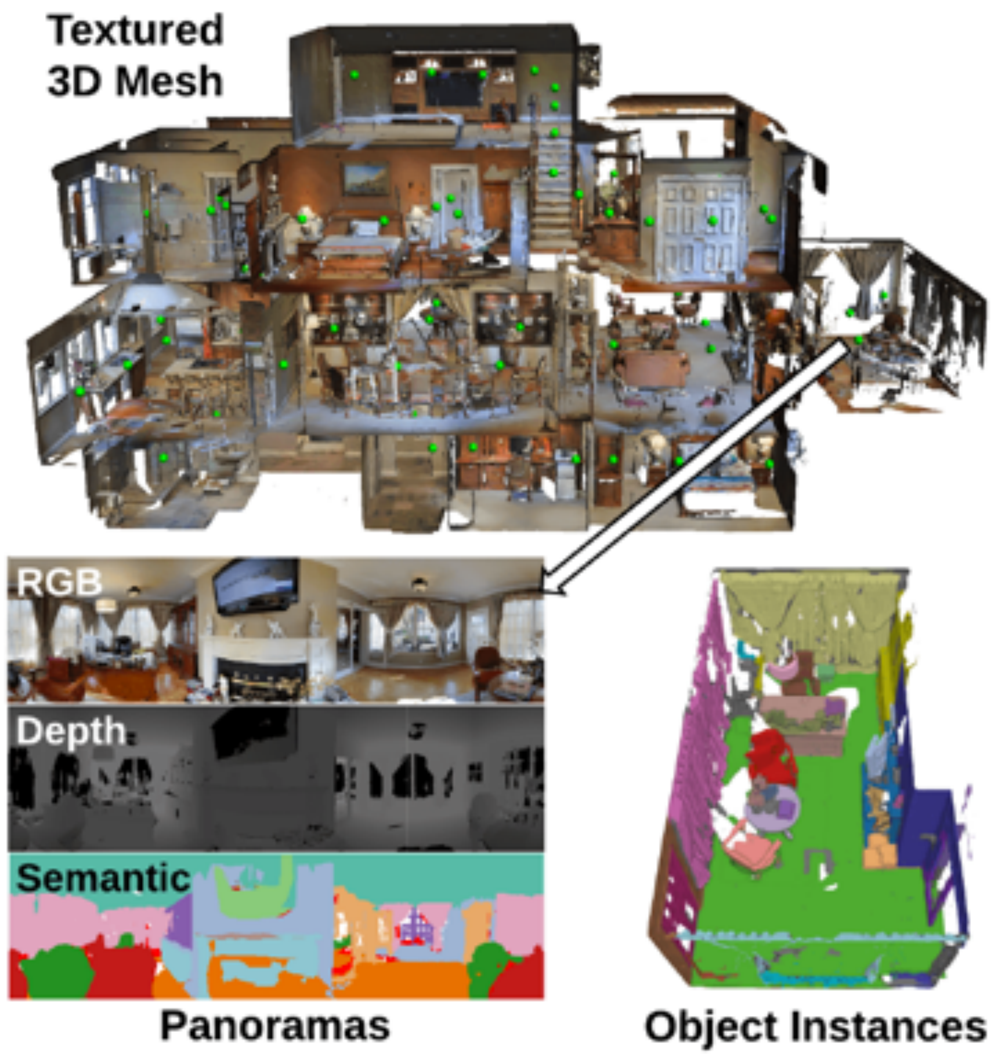



Image from: https://matterport.com/blog/2017/09/20/announcing-matterport3d-research-dataset/