SymCity: Feature Selection by Symmetry for Large Scale Image Retrieval

Giorgos Tolias, Yannis Kalantidis and Yannis Avrithis

Image and Video Analysis Team
National Technical University of Athens

ACM Multimedia 2012, 29 Oct. – 2 Nov. 2012, Nara, Japan

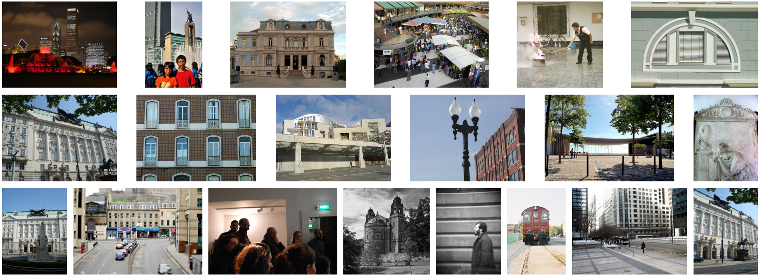
Outline

Outline

Specific object retrieval

Problem

- Search in a large corpus of images
- Robust matching against viewpoint change, photometric variations, occlusion and background clutter



Challenge

- Reduce memory requirements
- Leave performance unaffected or even improve

Specific object retrieval

Problem

- Search in a large corpus of images
- Robust matching against viewpoint change, photometric variations, occlusion and background clutter

Challenge

- Reduce memory requirements
- Leave performance unaffected or even improve

Index space bottleneck

Bag-of-Words (BoW)

- Good performance at low cost
- Index each local feature separately

Geometry verification

- Constantly better performance than BoW
- Space requirements slightly increased

Compact representations

- Lower space requirements, e.g. Fisher vectors [Perronnin *et al.* 2010]
- Not compatible with geometry verification

Feature selection

- Currently only from multiple views

Index space bottleneck

Bag-of-Words (BoW)

- Good performance at low cost
- Index each local feature separately

Geometry verification

- Constantly better performance than BoW
- Space requirements slightly increased

Compact representations

- Lower space requirements, e.g. Fisher vectors [Perronnin *et al.* 2010]
- Not compatible with geometry verification

Feature selection

- Currently only from multiple views

Index space bottleneck

Bag-of-Words (BoW)

- Good performance at low cost
- Index each local feature separately

Geometry verification

- Constantly better performance than BoW
- Space requirements slightly increased

Compact representations

- Lower space requirements, e.g. Fisher vectors [Perronnin *et al.* 2010]
- Not compatible with geometry verification

Feature selection

- Currently only from multiple views

Feature selection – related work

Corpus-wide, supervised (by geotag)

- Informative features [Schindler *et al.* 2007, Li & Kosecka 2006]

Corpus-wide, unsupervised

- Sparse PCA on vocabulary [Naikal *et al.* 2011]

Per image, supervised (by geotag)

- Foreground object detection [Gammeter *et al.* 2009]
- Scene map construction [Avrithis *et al.* 2010]

Per image, unsupervised

- Spatial verification of multiple views [Turcot & Lowe 2009]

Feature selection – related work

Corpus-wide, supervised (by geotag)

- Informative features [Schindler *et al.* 2007, Li & Kosecka 2006]

Corpus-wide, unsupervised

- Sparse PCA on vocabulary [Naikal *et al.* 2011]

Per image, supervised (by geotag)

- Foreground object detection [Gammeter *et al.* 2009]
- Scene map construction [Avrithis *et al.* 2010]

Per image, unsupervised

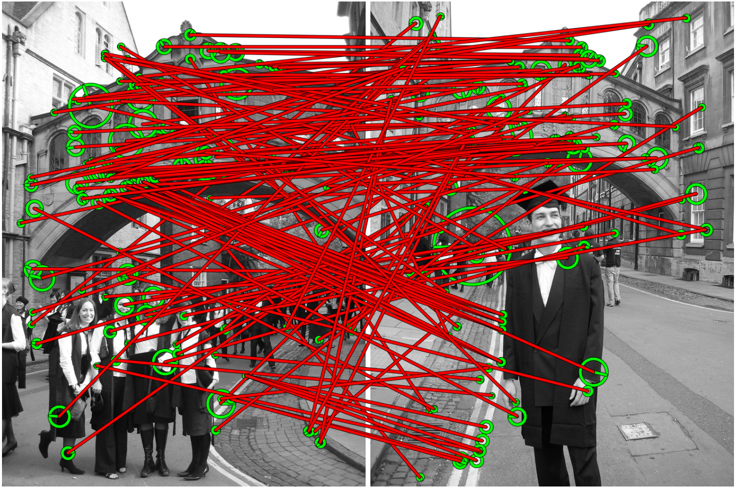
- Spatial verification of multiple views [Turcot & Lowe 2009]

Multiple view feature selection

[Turcot & Lowe 2009]

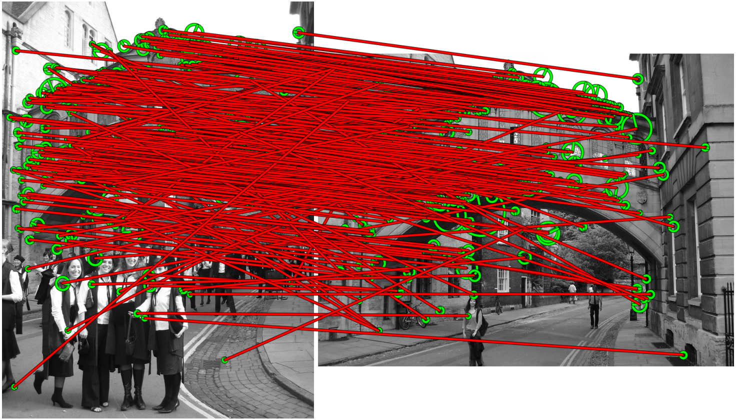
- BoW-based retrieval system
- Spatial verification by RANSAC
- Query with all images
- Keep spatially verified features

Multiple view selection - spatial verification

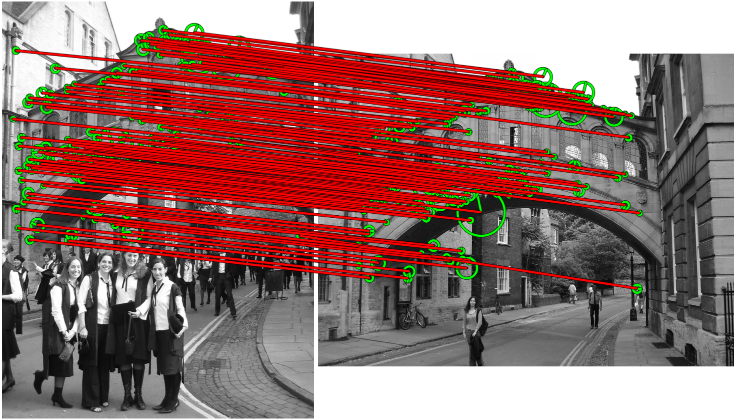


Multiple view selection - spatial verification

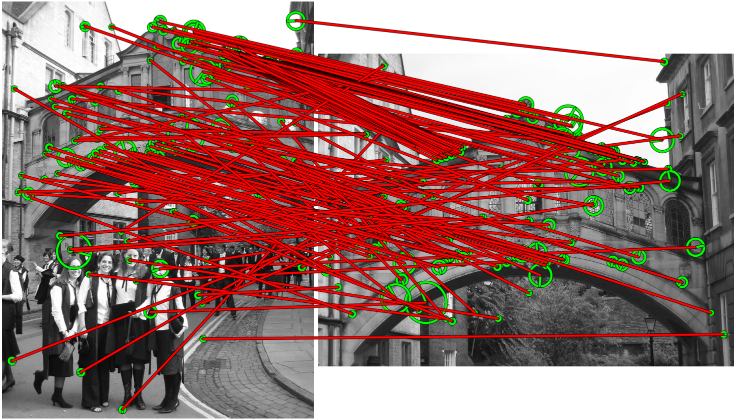
Multiple view selection - spatial verification



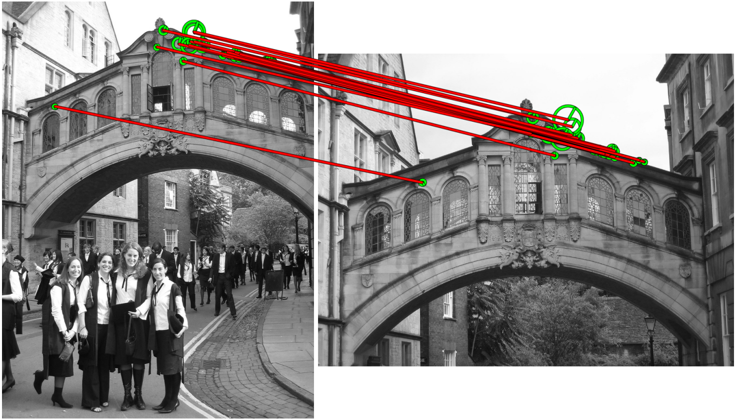
Multiple view selection - spatial verification



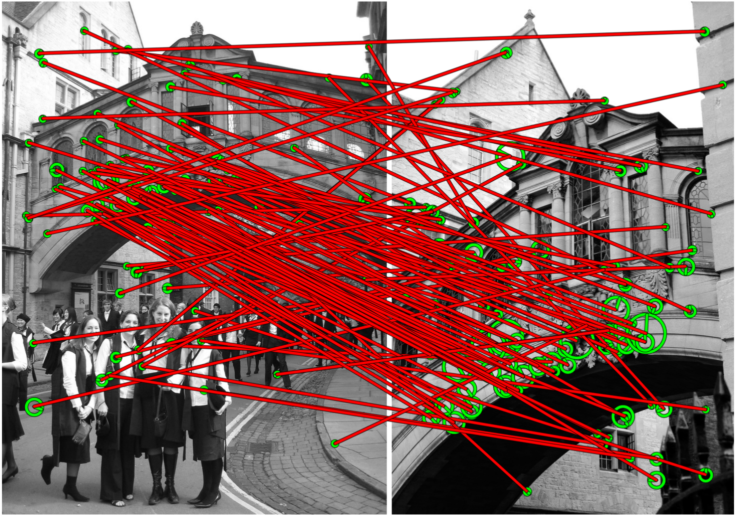
Multiple view selection - spatial verification



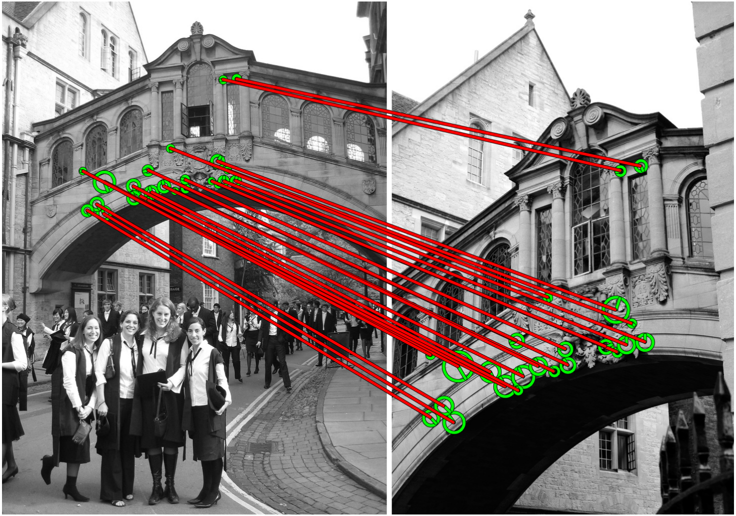
Multiple view selection - spatial verification



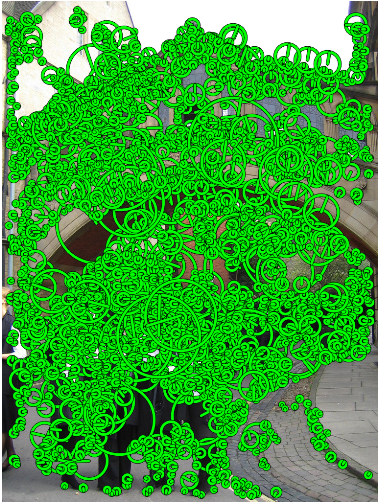
Multiple view selection - spatial verification



Multiple view selection - spatial verification



Multiple view selection - verified features



But how about single views?

Feature selection from single, unique views

multiple views

single view [this work]

Selection from single, unique views

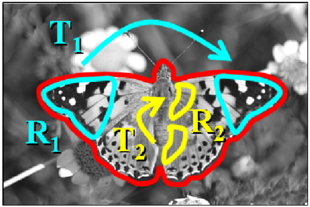
- Detect symmetries
- Detect repeating patterns
- Select all features participating in such patterns

Why symmetries?

Why self-similarities?

Rationale: local self-similarities are everywhere!

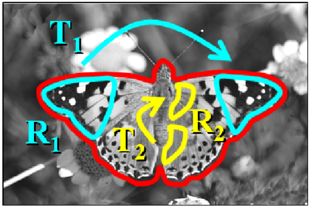
- Segmentation by composition [Bagon *et al.* 2008]



- Self-similarity descriptor [Shechtman & Irani 2007]

Rationale: local self-similarities are everywhere!

- Segmentation by composition [Bagon *et al.* 2008]



- Self-similarity descriptor [Shechtman & Irani 2007]

Proposed method

Spatial matching

- Image with itself
- Image with mirrored counterpart

Similar to geometry verification, but

- Descriptors, not visual words
- No one-to-one correspondence constraint
- No single transformation model

We propose two methods

- Spatial self-matching (SSM)
- Hough pyramid self-matching (HPSM)

Proposed method

Spatial matching

- Image with itself
- Image with mirrored counterpart

Similar to geometry verification, but

- Descriptors, not visual words
- No one-to-one correspondence constraint
- No single transformation model

We propose two methods

- Spatial self-matching (SSM)
- Hough pyramid self-matching (HPSM)

Proposed method

Spatial matching

- Image with itself
- Image with mirrored counterpart

Similar to geometry verification, but

- Descriptors, not visual words
- No one-to-one correspondence constraint
- No single transformation model

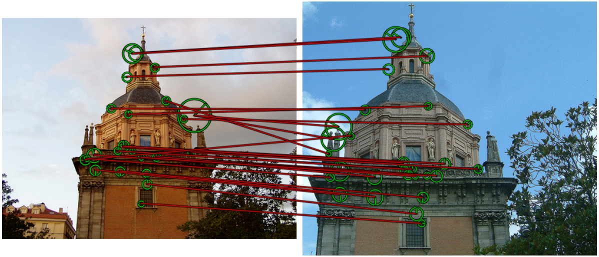
We propose two methods

- Spatial self-matching (SSM)
- Hough pyramid self-matching (HPSM)

Does feature selection affect performance?

Does it?

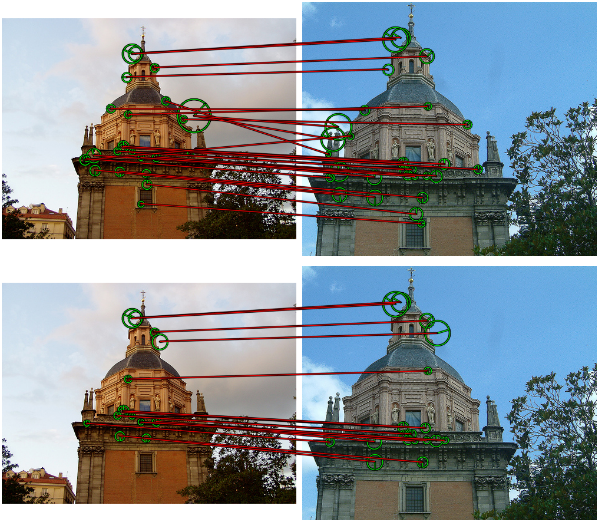
BoW tentative correspondences



Full feature set

Does it?

BoW tentative correspondences

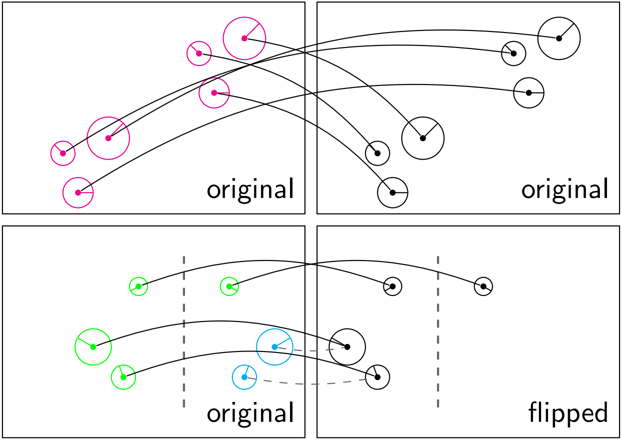


Full feature set

15% selected

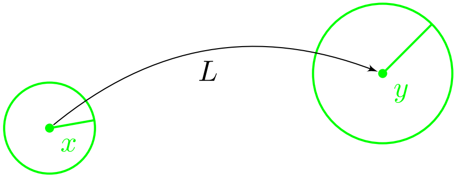
Outline

Idea



- Self matching: direct transformations
- Flipped matching: opposite transformations

Single correspondence hypothesis

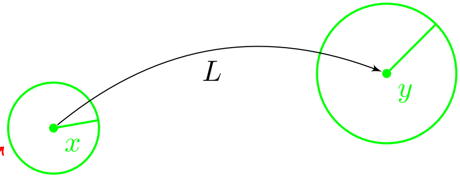


$$g(x, y) = g(c) = [p(c)^{\mathrm{T}} \ \sigma(c) \ \theta(c)]^{\mathrm{T}}$$

$$g(x) = [p(x)^{\mathrm{T}} \ \sigma(x) \ \theta(x)]^{\mathrm{T}}$$

$$g(y) = [p(y)^{\mathrm{T}} \ \sigma(y) \ \theta(y)]^{\mathrm{T}}$$

Single correspondence hypothesis

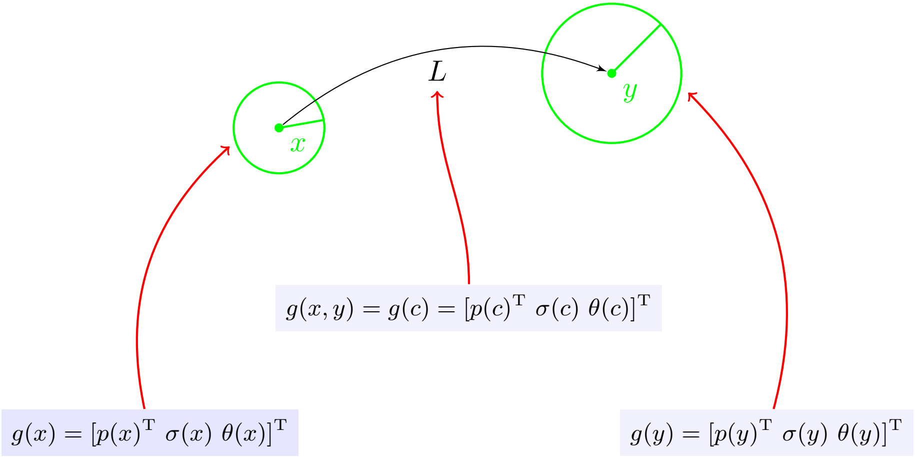


$$g(x, y) = g(c) = [p(c)^{\mathrm{T}}\ \sigma(c)\ \theta(c)]^{\mathrm{T}}$$

$$g(x) = [p(x)^{\mathrm{T}}\ \sigma(x)\ \theta(x)]^{\mathrm{T}}$$

$$g(y) = [p(y)^{\mathrm{T}}\ \sigma(y)\ \theta(y)]^{\mathrm{T}}$$

Single correspondence hypothesis



Self-matching: tentative correspondences

Valid pairs

$$C_v(X) = \{(x, y) \in X^2 : \; v(x, y) \; \}$$

valid iff $\|g(x, y)\| \geq \rho$

Descriptor nearest neighbors

k-nearest neighbors

descriptor distance

$$N(x) = \{y \in X : y \in \; \mathcal{N}_X^k(x) \; \wedge \; d(x, y) \; \leq \delta\}$$

$$C_d(X) = \{(x, y) \in X^2 : y \in N(x)\}$$

Tentative correspondences

$$C_t(X) = C_d(X) \cap C_v(X)$$

Self-matching: tentative correspondences

Valid pairs

$$C_v(X) = \{(x, y) \in X^2 : \ v(x, y) \ \}$$

valid iff $\|g(x, y)\| \geq \rho$

Descriptor nearest neighbors

k-nearest neighbors

descriptor distance

$$N(x) = \{y \in X : y \in \ \mathcal{N}_X^k(x) \ \wedge \ d(x, y) \ \leq \delta\}$$

$$C_d(X) = \{(x, y) \in X^2 : y \in N(x)\}$$

Tentative correspondences

$$C_t(X) = C_d(X) \cap C_v(X)$$

Self-matching: tentative correspondences

Valid pairs

$$C_v(X) = \{(x, y) \in X^2 : \; v(x, y) \; \}$$

valid iff $\|g(x, y)\| \geq \rho$

Descriptor nearest neighbors

k-nearest neighbors

descriptor distance

$$N(x) = \{y \in X : y \in \; \mathcal{N}_X^k(x) \; \wedge \; d(x, y) \; \leq \delta\}$$

$$C_d(X) = \{(x, y) \in X^2 : y \in N(x)\}$$

Tentative correspondences

$$C_t(X) = C_d(X) \cap C_v(X)$$

Self-matching: tentative correspondences

Valid pairs

$$C_v(X) = \{(x, y) \in X^2 : \; v(x, y) \; \}$$

valid iff $\|g(x, y)\| \geq \rho$

Descriptor nearest neighbors

k-nearest neighbors

descriptor distance

$$N(x) = \{y \in X : y \in \; \mathcal{N}_X^k(x) \; \wedge \; d(x, y) \; \leq \delta\}$$

$$C_d(X) = \{(x, y) \in X^2 : y \in N(x)\}$$

Tentative correspondences

$$C_t(X) = C_d(X) \cap C_v(X)$$

Spatial self-matching (SSM)

- Inspired by fast spatial matching (FSM) [Philbin *et al.* 2007]
- Each correspondence $c \in C$ gives rise to a hypothesis $h = t(c)$
- Hypothesis inliers: $I_C(h) = \{(x, y) \in C : \|\mathbf{p}(y) - h\mathbf{p}(x)\| < \epsilon\}$
- FSM seeks best hypothesis overall, $\max_h \{|I_C(h)| : h \in H_C(c)\}$
- We find hypotheses per correspondence $c = (x, y)$

$$H_C(x, y) = \{h \in t(C) : \|\mathbf{p}(y) - h\mathbf{p}(x)\| < \epsilon\}$$

  and seek the best to define the correspondence strength:

$$\alpha_C(c) = \max\{|I_C(h)| : h \in H_C(c)\}$$

- Verified correspondences: $\alpha(C) = \{c \in C : \alpha_C(c) \geq \tau_\alpha\}$
- Select features of verified correspondences

Spatial self-matching (SSM)

- Inspired by fast spatial matching (FSM) [Philbin *et al.* 2007]
- Each correspondence $c \in C$ gives rise to a hypothesis $h = t(c)$
- Hypothesis inliers: $I_C(h) = \{(x, y) \in C : \|\mathbf{p}(y) - h\mathbf{p}(x)\| < \epsilon\}$
- FSM seeks best hypothesis overall, $\max_h\{|I_C(h)| : h \in H_C(c)\}$
- We find hypotheses per correspondence $c = (x, y)$

$$H_C(x, y) = \{h \in t(C) : \|\mathbf{p}(y) - h\mathbf{p}(x)\| < \epsilon\}$$

and seek the best to define the correspondence strength:

$$\alpha_C(c) = \max\{|I_C(h)| : h \in H_C(c)\}$$

- Verified correspondences: $\alpha(C) = \{c \in C : \alpha_C(c) \geq \tau_\alpha\}$
- Select features of verified correspondences

SSM algorithm

procedure $\alpha \leftarrow \mathrm{SSM}(C, t; \tau_\alpha)$

input : correspondences C, transformations t

parameter: inlier threshold τ_α

output : inlier strengths α

for $c \in C$ **do** ▷ initialize
- $\mathit{inlier}(c) \leftarrow$ FALSE ▷ mark as outlier
- $\alpha(c) \leftarrow 0$ ▷ zero strength

for $c \in C$ **do** ▷ for all hypotheses
- **if** $\mathit{inlier}(c)$ **then continue** ▷ skip hypothesis?
- $h \leftarrow t(c)$ ▷ current hypothesis
- $I \leftarrow I_C(h)$ ▷ current inliers (8)
- **if** $|I| < \tau_\alpha$ **then continue** ▷ verified hypothesis?
- **for** $c' \in I$ **do** ▷ for all inliers
  - $\mathit{inlier}(c') \leftarrow$ TRUE ▷ mark as inlier
  - $\alpha(c') \leftarrow \max(\alpha(c'), |I|)$ ▷ update strength

return α ▷ inlier strengths

Flipped matching

- Same matching algorithm
- Flip entire image, extract new set of features & descriptors Y
- y': back-projected counterpart of feature $y \in Y$
- Create correspondences in $X \times Y$

$$
\begin{aligned}
C_v(X,Y) &= \{(x,y) \in X \times Y : v(x,y')\} \\
C_d(X,Y) &= \{(x,y) \in X \times Y : y \in N(x)\} \\
C_t(X,Y) &= C_d(X,Y) \cap C_v(X,Y)
\end{aligned}
$$

- Validate against direct selection
- Select features
  - on original image X
  - back-projected from flipped image Y

Flipped matching

- Same matching algorithm
- Flip entire image, extract new set of features & descriptors Y
- y': back-projected counterpart of feature $y \in Y$
- Create correspondences in $X \times Y$

$$
\begin{aligned}
C_v(X,Y) &= \{(x,y) \in X \times Y : v(x,y')\} \\
C_d(X,Y) &= \{(x,y) \in X \times Y : y \in N(x)\} \\
C_t(X,Y) &= C_d(X,Y) \cap C_v(X,Y)
\end{aligned}
$$

- Validate against direct selection
- Select features
  - on original image X
  - back-projected from flipped image Y

Self-matching – example

Flipped matching – example

Selected features – example

- Selected by self-matching (magenta)
- Selected by flipped-matching
  - on original image (green)
  - on flipped image, back-projected (cyan)

Outline

Hough pyramid self-matching (HPSM)

- Based on Hough pyramid matching (HPM) [Tolias & Avrithis 2011]
- No inlier threshold ϵ
- No inlier counting or transformation estimation
- Supports multiple, even non-rigid transformations
- Correspondence strength: geometrical consistency to other correspondences
- Linear in the number of correspondences

However

- Strength is $\max$-normalized
- No one-to-one mapping as in original HPM

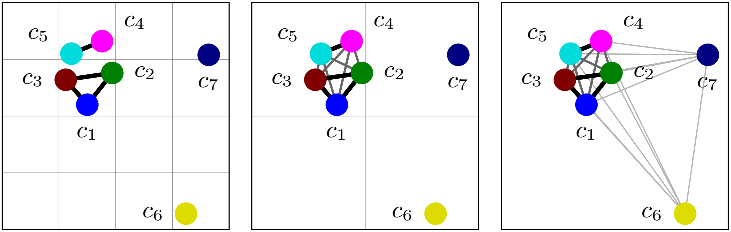
Hough pyramid self-matching (HPSM)

- Based on Hough pyramid matching (HPM) [Tolias & Avrithis 2011]
- No inlier threshold ϵ
- No inlier counting or transformation estimation
- Supports multiple, even non-rigid transformations
- Correspondence strength: geometrical consistency to other correspondences
- Linear in the number of correspondences

However

- Strength is $\max$-normalized
- No one-to-one mapping as in original HPM

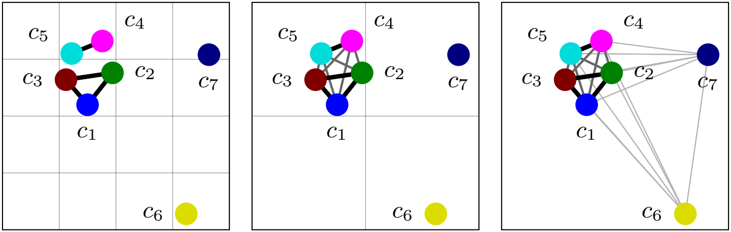
Hough pyramid self-matching



Level 0 Level 1 Level 2

	p	q	similarity score
c_1			$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_1)$
c_2			$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_2)$
c_3			$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_3)$
c_4			$(1 + \frac{1}{2}3 + \frac{1}{4}2)w(c_4)$
c_5			$(1 + \frac{1}{2}3 + \frac{1}{4}2)w(c_5)$
c_6			$\frac{1}{4}6w(c_8)$
c_7			$\frac{1}{4}6w(c_9)$

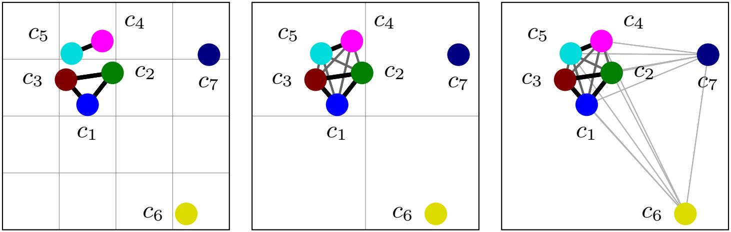
Hough pyramid self-matching



Level 0 Level 1 Level 2

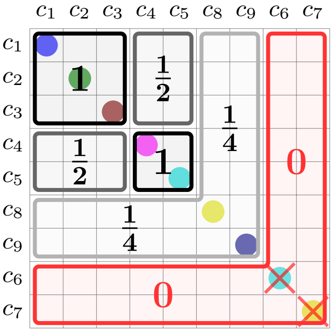
	p q	similarity score
c_1		$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_1)$
c_2		$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_2)$
c_3		$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_3)$
c_4		$(1 + \frac{1}{2}3 + \frac{1}{4}2)w(c_4)$
c_5		$(1 + \frac{1}{2}3 + \frac{1}{4}2)w(c_5)$
c_6		$\frac{1}{4}6w(c_8)$
c_7		$\frac{1}{4}6w(c_9)$

Hough pyramid self-matching



Level 0 Level 1 Level 2

	p q	similarity score
c_1		$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_1)$
c_2		$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_2)$
c_3		$(2 + \frac{1}{2}2 + \frac{1}{4}2)w(c_3)$
c_4		$(1 + \frac{1}{2}3 + \frac{1}{4}2)w(c_4)$
c_5		$(1 + \frac{1}{2}3 + \frac{1}{4}2)w(c_5)$
c_6		$\frac{1}{4}6w(c_8)$
c_7		$\frac{1}{4}6w(c_9)$

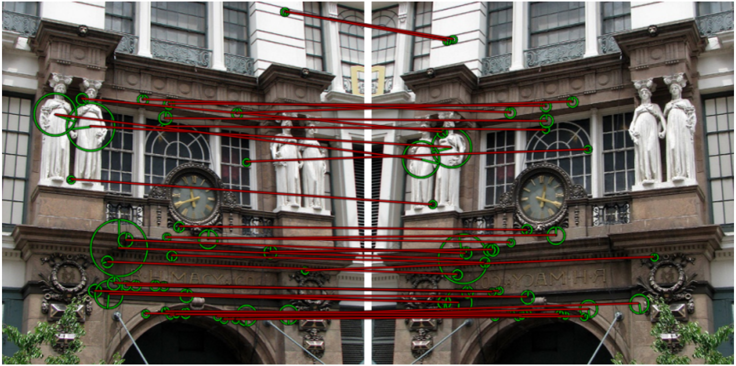


HPSM algorithm

procedure $\beta \leftarrow \mathrm{HPSM}(C, L)$
input : correspondences C, levels L
output: strengths β
begin
  $B \leftarrow \textsc{Partition}(L)$ ▷ partition space in L levels
  for $c \in C$ **do** $\beta(c) \leftarrow 0$ ▷ initialize strengths
  $\mathrm{HPSM\text{-}\textsc{rec}}(\beta, C, L-1, B)$ ▷ recurse at top
  return $\beta / \max(\beta)$ ▷ normalize

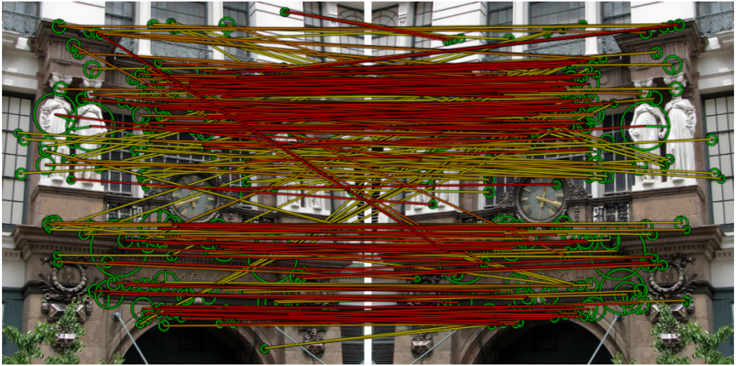
procedure $\mathrm{HPSM\text{-}\textsc{rec}}(\beta, C, \ell, B)$
in/out : strengths β
input : correspondences C, level ℓ, partition map B
begin
  if $\ell < 0$ **then return**
  for $b \in B_\ell$ **do** $F(b) \leftarrow \emptyset$ ▷ initialize histogram
  for $c \in C$ **do** ▷ populate histogram
    $F(q_\ell(c)) \leftarrow F(q_\ell(c)) \cup c$ ▷ . . . by quantizing
  for $b \in B_\ell$ **do**
    $F \leftarrow F(b)$ ▷ correspondences in b
    if $|F| < 2$ **then continue** ▷ exclude singles
    $\mathrm{HPSM\text{-}\textsc{rec}}(\beta, F, \ell-1, B)$ ▷ recurse down
    if $\ell = L-1$ **then** $m \leftarrow 2$ **else** $m \leftarrow 1$
    for $c \in F$ **do** ▷ update strengths in b
      $\beta(c) \leftarrow \beta(c) + 2^{-\ell} m |F|$

Hough pyramid self-matching – example



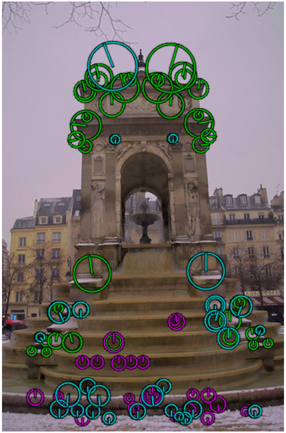
correspondences in a single bin at level 0

Hough pyramid self-matching – example



all correspondences (red: strongest; yellow: weakest)

Selection examples



Selection examples

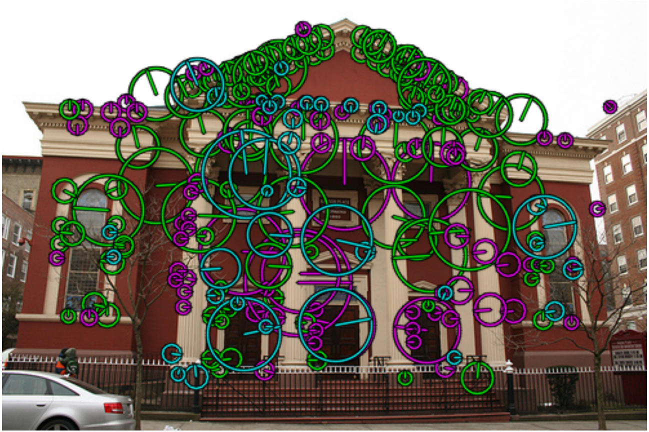
Selection examples

Selection examples

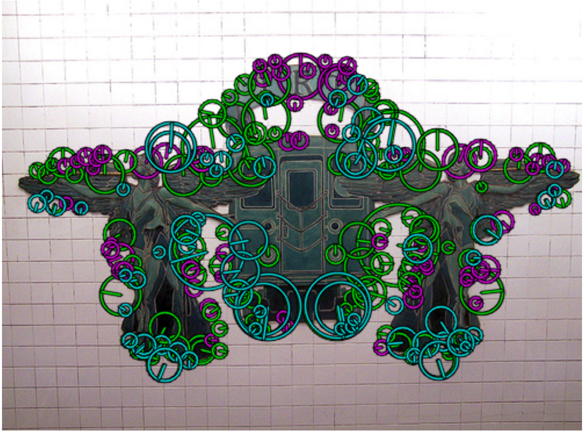
Selection examples

Selection examples

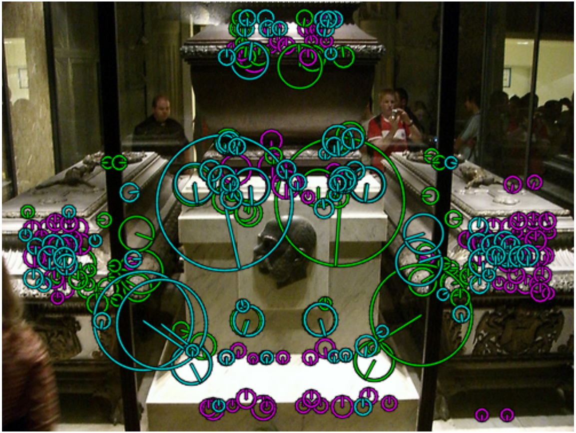
Selection examples



Selection examples



Selection examples



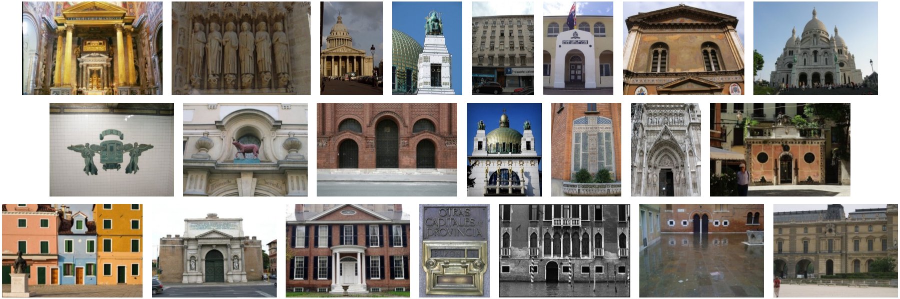
Selection examples

Selection examples

Selection examples

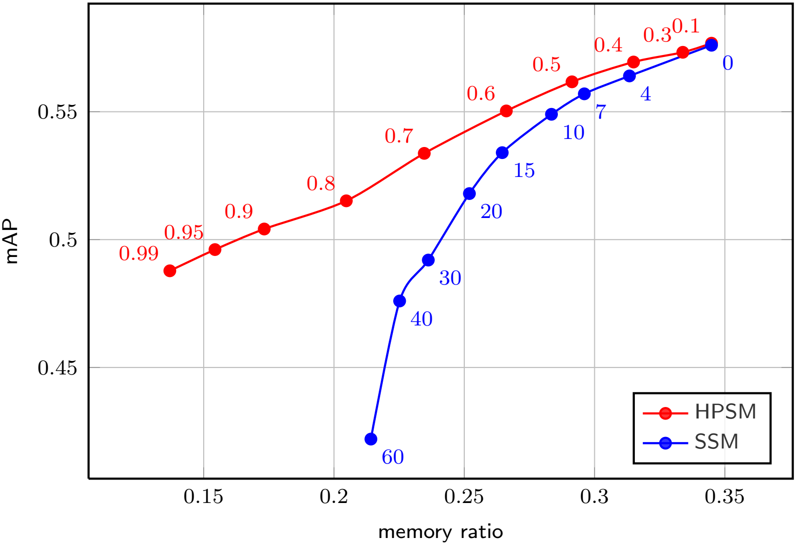
Outline

SymCity dataset

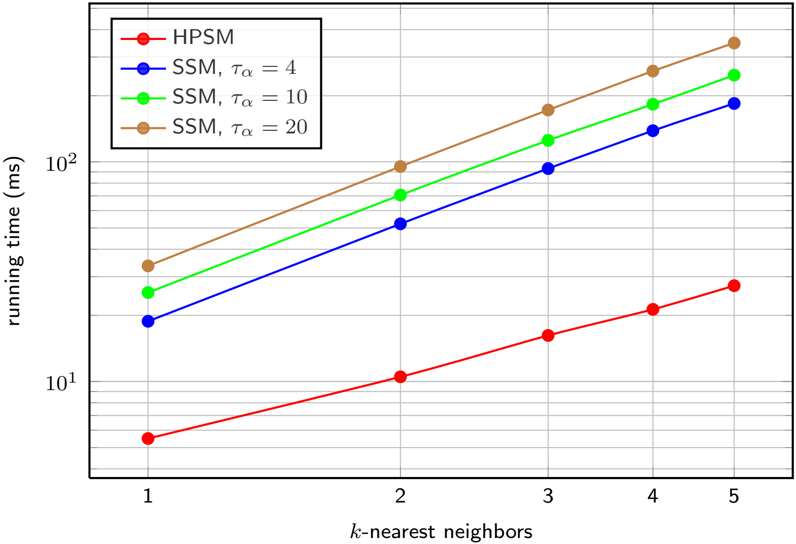


- 953 annotated photos, 299 groups
- Semi-automatic generation of image clusters of up to 4 images
- One single image from each group in the database
- Remaining 645 used as queries

SSM vs HPSM – 100K distractors used



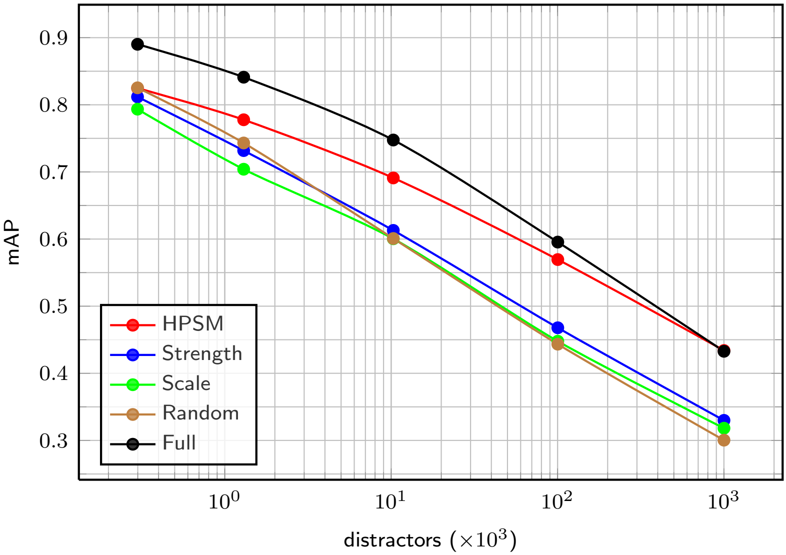
SSM vs HPSM – 100K distractors used



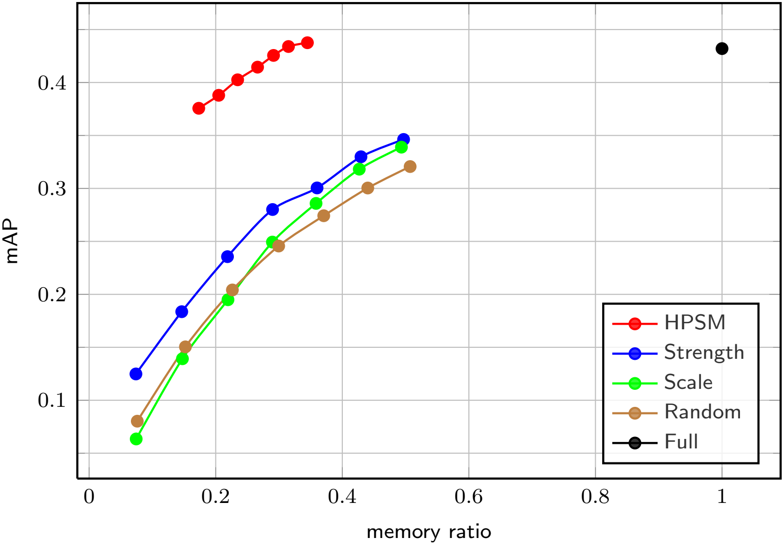
HPSM tuning experiment

k	1	2	**3**	4	5
$\tau_\beta = \mathbf{0.4}$	0.545	0.566	**0.569**	0.566	0.568
$\tau_\beta = 0.6$	0.522	0.538	0.550	0.551	0.547
$\tau_\beta = 0.8$	0.484	0.511	0.515	0.524	0.529

Large scale experiment – distractors



Large scale experiment – memory ratio



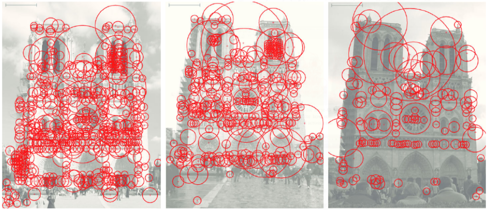
Outline

Future work

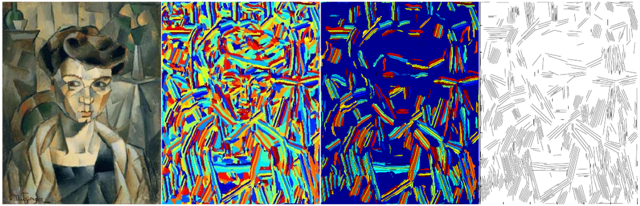
- Go at larger scale – outperform full set?
- Combine both multi-view selection and our method in a single retrieval system
- Use verified correspondences as feature tracks for vocabulary learning (visual synonyms)
- Use in other problems where symmetries and pattern mining are needed - HPSM runs at 16ms on average

Update

- Local symmetry feature detection [Hauagge & Snavely, CVPR 2012]



- Self-similar sketch [Vedaldi & Zisserman, ECCV 2012]



SymCity page:

http://image.ntua.gr/iva/research/symcity

Thank you!