On the Art of Establishing Correspondence

Jiri Matas

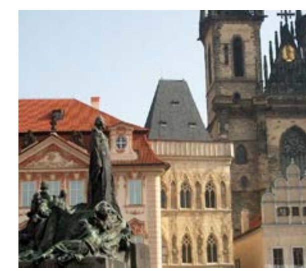
Presentation prepared with
Dmytro Mishkin

Visual Recognition Group

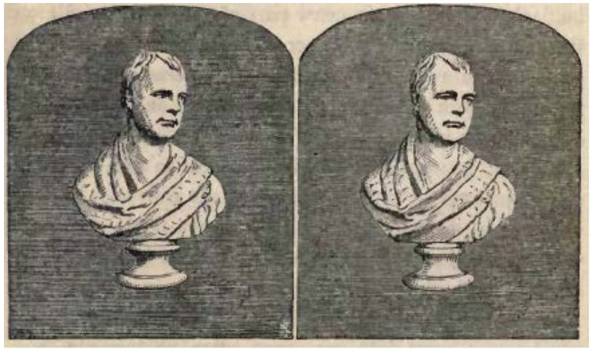
Center for Machine Perception

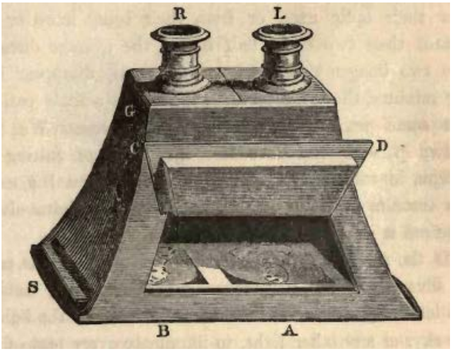
Czech Technical University in Prague

http://cmp.felk.cvut.cz



Correspondence in Stereoscopic Images

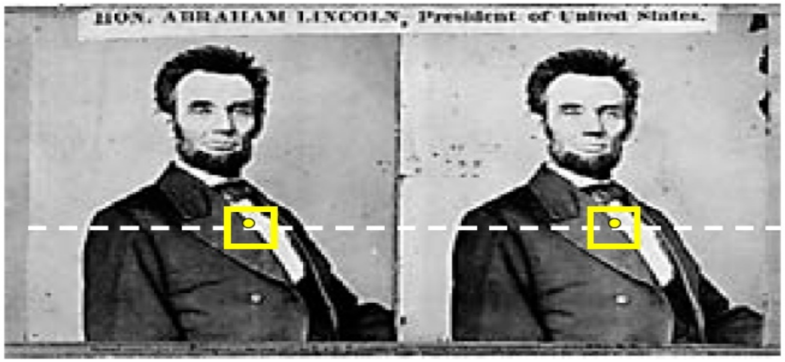




Brewster Stereoscope, 1856

A "photo" for each eyes

Correspondence established by the human visual system.



Correspondence by classical narrow-baseline stereo methods, e.g. Cox 1996

Correspondence Problems

Given images A and B, find a geometric model linking them and a set of features consistent with the model.

Correspondence Problems

Given images A and B, find a geometric model linking them.

Given images A and B, and a geometric model linking them (F, E, H), estimate reliably the confidence that the model is correct.

If images A and B are geometrically unrelated, establish fast and with high confidence this fact.

Given a set of n images A_i, select a subset of pairs that are geometrically related much faster then in time proportional to n^2.

(Registration) Given images A and B and an approximation of the geometric model linking them (F, E, H), find the highest precision model.

Wide Baseline Stereo (WBS), circa 2000

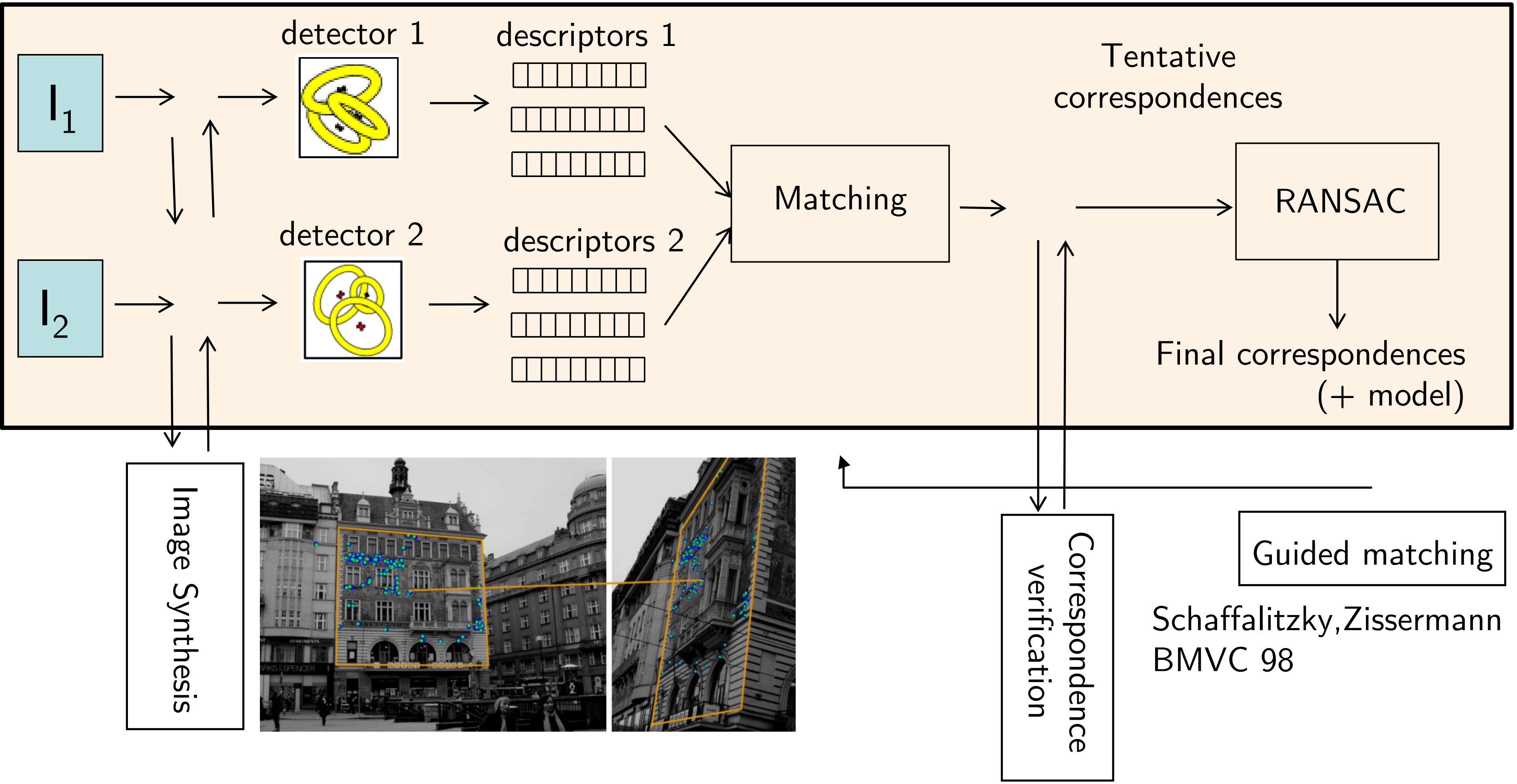
Widening of the baseline, zooming in/out, rotation

Standard approach:

D. Lowe, 2000, SIFT

Also:
Mikolajczyk & Schmid,
Tuytelaars & van Gool,
Matas et al. and many
other

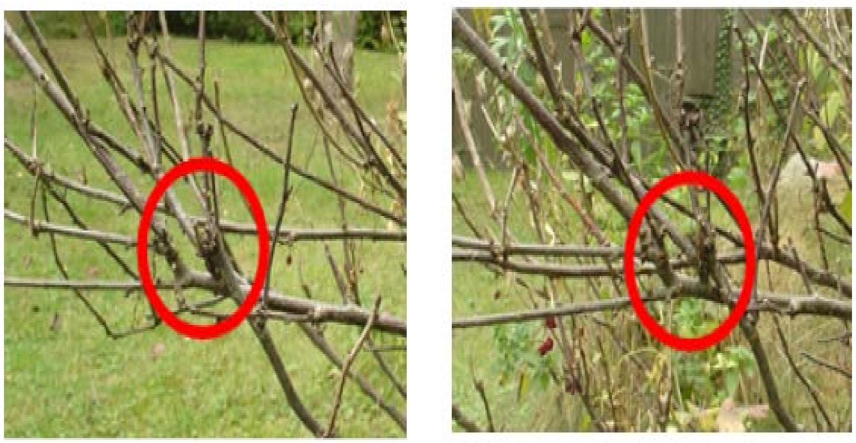
Classical Two-view Correspondence Pipeline



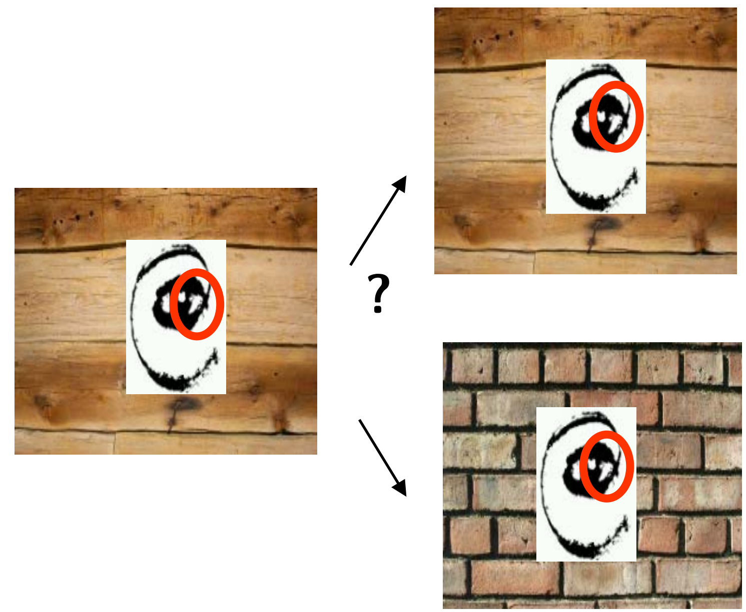
Morel, Yu: ASIFT: A New Framework for Fully Affine Invariant Image Comparison. SIAM JIS 2009
Mishkin, MODS: Fast and robust method for two-view matching. CVIU 2015

Correspondence Verification

- Difficult matching problems:
  - Rich 3D structure with many occlusions
  - Small overlap
  - Image quality and noise
  - (Repetitive patterns)



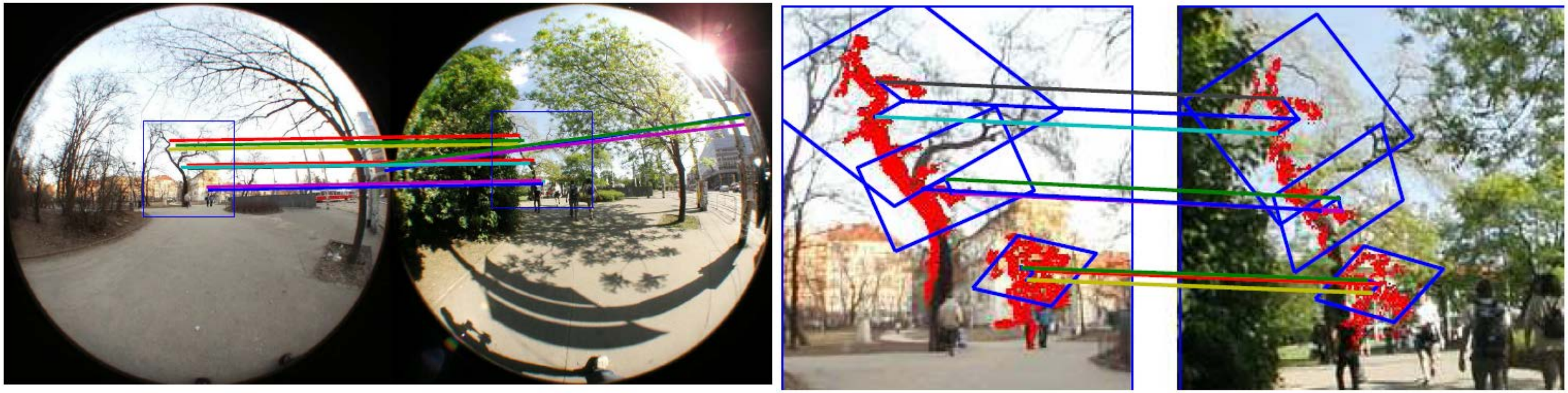
measurement region too large



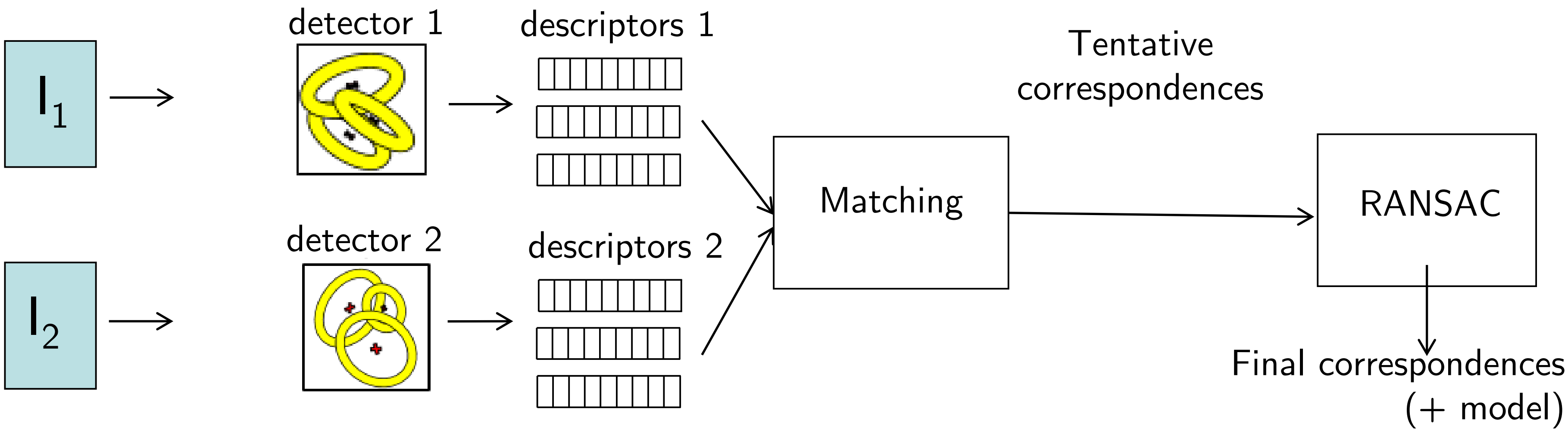
measurement region too small

Correspondence Verification by Co-segmentation

- high discriminability
  - significantly outperforms a standard selection process based SIFT-ratio
- very fast (0.5 sec / 1000 correspondences)
- always applicable before RANSAC
- the process generating tentative correspondences can be much more permissive
  - 99% of outliers not a problem, correct correspondences recovered
  - higher number of correct correspondences



Classical Two-view Correspondence Pipeline



DoG	SIFT	1st to 2nd distance ratio	RANSAC
Hessian	RootSIFT	**1st geometric inconsistent**	**RANSAC++**

1st Geometricly Inconsistent Constraint [Mishkin et al., Two-View Matching with View Synthesis Revisited. IVCNZ 2013] (rediscovered: in [Sarlin et.al, CVPR 2019)

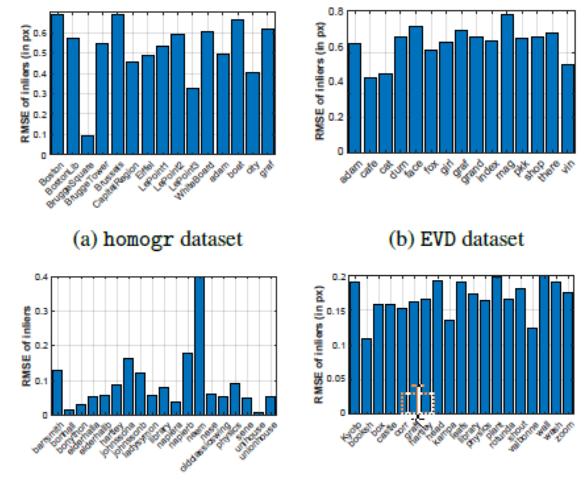
similar constraints used for training descriptors: SuperPoint (CVPRW 2018), D2Net (CVPR 2019), RFNet (arXiv 2019, called "neighbor mask")

MAGSAC (Barath et al., CVPR 2019, Poster 3-1.158)

Idea: do not require the user to provide the scale.
The optimal one is different for every problem.

Marginalize: the result is a weighted average over a range of σ, weighted by the log-likelihood for the mode.

			[1] a	+ σ	[1] b	+ σ	[2]	[3]	**MAGSAC**
H – homography, **F** – fundamental m., **E** – essential m.	(1) **F**	RMSE (in px)	0.56	0.52	0.58	0.50	1.01	0.63	**0.38**
	(2) **F**		0.28	**0.27**	0.31	0.31	0.33	0.46	0.30
	(4) **F**		0.53	0.52	0.50	0.50	0.58	0.72	**0.47**
	(5) **H**		3.39	2.13	3.53	2.19	2.95	1.83	**1.37**
	(6) **H**		5.42	4.07	4.78	3.55	4.55	5.05	**1.76**
	(3) **E**		9.61	9.48	10.62	10.23	10.17	15.56	**6.51**
	(all)		3.30	2.83	3.39	2.88	3.27	4.04	**1.80**
	(1) **F**	time (in msecs)	25	25	**17**	**17**	**17**	55	31
	(2) **F**		393	394	**380**	**380**	**380**	447	939
	(3) **F**		132	140	119	128	126	**46**	467
	(4) **H**		71	72	64	65	65	**37**	131
	(5) **H**		367	369	353	356	355	291	**162**
	(6) **E**		2 548	2 549	2 535	2 537	2 536	4 637	**2 398**
	(all)		589	592	**578**	581	580	921	688
	(1) **F**	% fails	0.06	0.06	0.06	0.06	0.06	0.06	**0.00**
	(2) **F**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	(3) **F**		**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**
	(4) **H**		0.12	0.12	0.12	0.12	0.12	**0.00**	0.06
	(5) **H**		0.57	0.50	0.57	0.43	0.53	0.33	**0.29**
	(6) **E**		0.27	0.22	0.26	0.22	0.24	0.23	**0.00**
	(all)		0.18	0.15	0.16	0.14	0.16	0.10	**0.03**



(a) homogr dataset (b) EVD dataset (c) AdelaideRMF dataset (d) kusvod2 dataset

[1]a – LO-RANSAC

[1]b – LO-MSAC

[2] – LO-RANSAAC

[3] – AC-RANSAC

Data interpretation and model likelihood

Distribution assumptions:

- Outliers are uniformly distributed $(\sim\mathcal{U}(0,l))$

Typically, the inlier residuals are calculated as the Eucledian-distance from the model in a ρ-dimensional space. Thus,

- the inliers residuals have chi-square distribution

Distance function

Likelihood of model θ given σ:

$$L(\theta \mid \sigma) = \underbrace{\frac{1}{l^{|\mathcal{X}|-|I(\sigma)|}}}_{\text{Comes from the outlier distribution}} \prod_{x\in I(\sigma)} \underbrace{\left[2C(p)\sigma^{-p}D^{p-1}(\theta,x)\exp\frac{-D^{2}(\theta,x)}{2\sigma^{2}}\right]}_{\text{Comes from the inliers' distribution}}$$

Set of inliers which σ implies

GC-RANSAC [Barath and Matas, CVPR 2018]

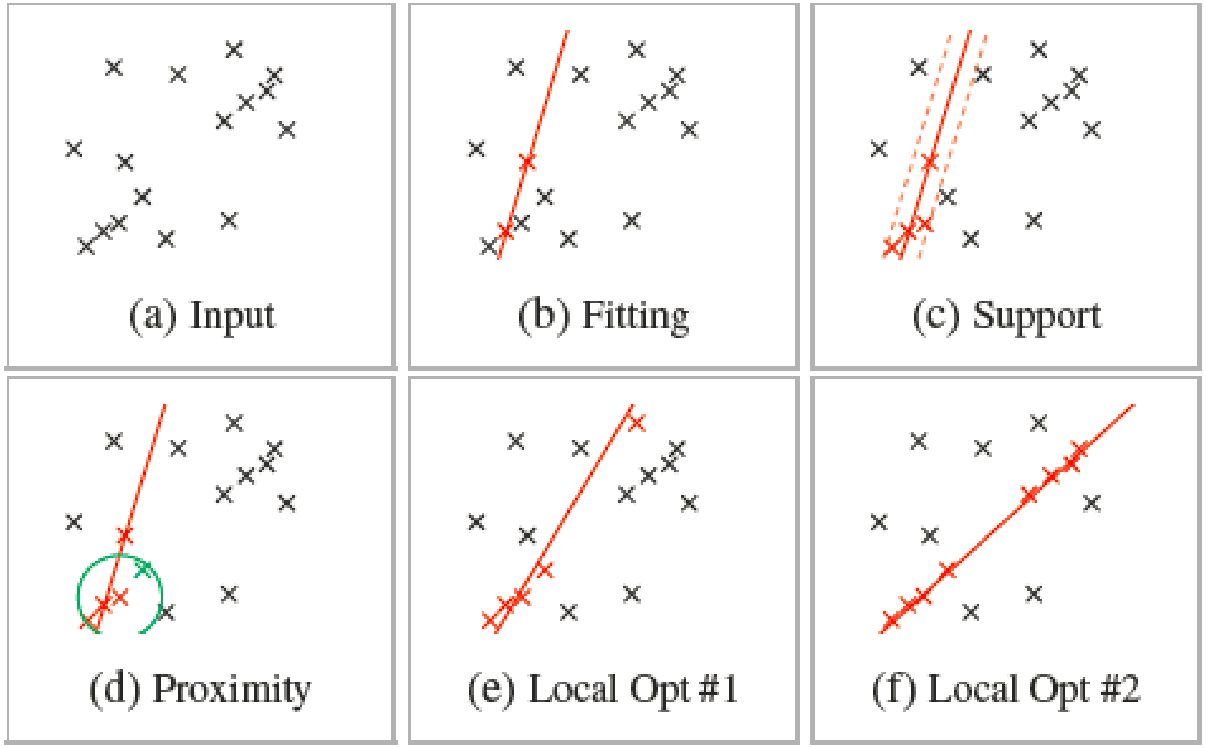
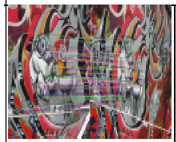
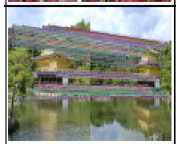
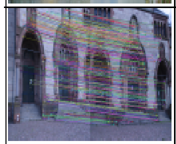


Figure 1: The proposed graph-cut based local optimization converging from a "not-all-inlier" sample, i.e. it is contaminated by an outlier, to the desired model. (a) The input data points, (b) RANSAC-like sampling and model fitting, (c) computation of model support, e.g. counting the inliers, (d) considering spatial proximity by graph-cut, (e-f) iterated local optimization using least-squares fitting and graph-cut.

GC RANSAC - Performance

		Confidence 95%						
		RSC	PSC	PSCd	FLO	SPRT	GC	P-NSC
	$\overline{\mathcal{E}}$	1.3 ± 0.0	1.4 ± 0.4	1.6 ± 0.2	0.88 ± 0.3	1.2 ± 0.3	$\mathbf{0.8} \pm 0.4$	0.8 ± 0.2
	$\overline{\mathcal{T}}$	0.9 ± 0.2	$\mathbf{0.5} \pm 0.1$	0.8 ± 0.2	1.6 ± 0.5	1.1 ± 0.15	2.8 ± 0.7	3.1 ± 0.3
	$\overline{\mathcal{S}}$	34.2 ± 1.0	$\mathbf{14.8} \pm 1.1$	23.2 ± 6.2	30.3 ± 5.2	69.9 ± 5.7	29.9 ± 9.5	19.1 ± 6.1
	$\overline{\mathcal{E}}$	2.5 ± 2.1	7.9 ± 5.1	9.8 ± 6.7	1.8 ± 1.7	6.5 ± 6.3	$\mathbf{0.7} \pm 0.2$	4.2 ± 2.3
	$\overline{\mathcal{T}}$	7.1 ± 3.9	$\mathbf{0.8} \pm 0.2$	0.9 ± 0.4	6.2 ± 1.6	5.1 ± 1.7	7.8 ± 1.7	8.1 ± 3.3
	$\overline{\mathcal{S}}$	113.7 ± 65.2	5.6 ± 2.7	$\mathbf{4.6} \pm 2.9$	59.6 ± 46.3	229 ± 123	37.8 ± 26.8	69.7 ± 39.6
	$\overline{\mathcal{E}}$	2.3 ± 0.6	4.3 ± 0.9	3.1 ± 0.9	1.3 ± 1.1	2.4 ± 0.6	$\mathbf{0.35} \pm 0$	2.4 ± 0.6
	$\overline{\mathcal{T}}$	17.8 ± 7.9	4.9 ± 2.1	$\mathbf{4.6} \pm 1.1$	16.7 ± 5.9	13.2 ± 5.2	18.3 ± 5.5	13.7 ± 5.2
	$\overline{\mathcal{S}}$	39.7 ± 17.2	12.1 ± 5.33	$\mathbf{10} \pm 2.2$	33.5 ± 13	36.6 ± 19.8	33.3 ± 29.5	26.4 ± 13.5

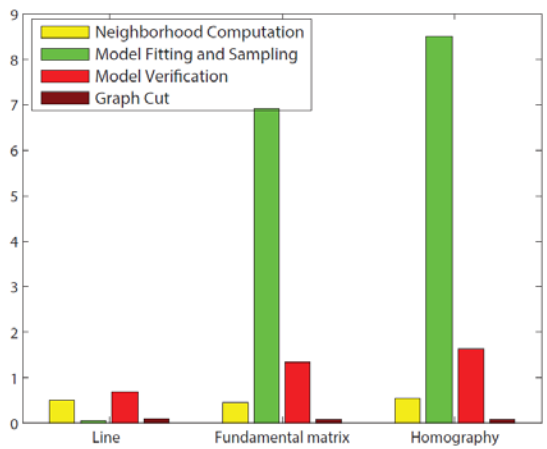
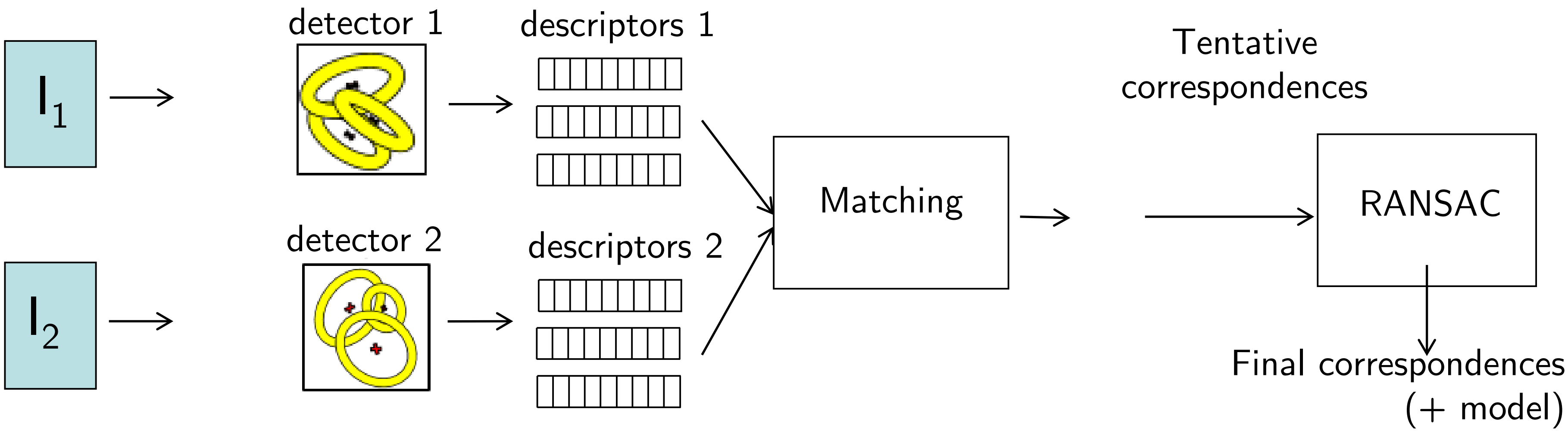


Figure 7: The breakdown of the processing times in milliseconds. Computed as the mean of all tests. *Best viewed in color.*

Is Classical Two-view Pipeline Dead? Dying?



DoG	SIFT	1^{st} to 2^{nd} distance ratio	RANSAC
Hessian	RootSIFT	**1^{st} geometric inconsistent**	**RANSAC++**

- Learnt descriptors superior: HardNet, ContextDesc; but that does not change the pipeline

- Detection and description learnt together, possibly also the metric for matching: SuperPoint, D2Net have superior results

- RANSAC-like differential methods for end-to-end pipelines:
  - Ranftl and Koltun, Deep Fundamental Matrix Estimation, ECCV 2018
  - Brachmann, PhD thesis, 2018

Classical Two-view Pipeline Dying?

D. DeTone, T. Malisiewicz, A. Rabinovich:
SuperPoint: Self-Supervised Interest Point Detection and Description. CoRR abs/1712.07629 (2017):

Convolutional neural networks have been shown to be superior to hand-engineered representations on almost all tasks requiring images as input.

The Classical Pipeline: what is the verdict of the Image Matching: Local Features & Beyond CVPR 2019 Workshop Challenge?

We appreciate the collaboration of the organizers.
Big thank you goes to:
Eduard Trulls trulls@google.com
Kwang Moo Yi kyi@uvic.ca

Thanks to the authors of:

- COLMAP who made this type of challenge possible
  - Johannes Schönberger, Jan-Michael Frahm
- Challenge Contributors that provided their results to us
  - Mihai Dusmanu (D2Net)
  - Zixin LUO (ContextDesc)
  - Daniel DeTone (SuperPoint)

Stereo best mAP15: 8%

SfM best mAP15: 73%

Why? Seems that something is wrong? Plus SfM seems simpler!

[P1] Phototourism dataset — Stereo task

Performance in stereo matching, averaged over all the test sequences.

- Click here for a breakdown by sequence

Show 10 entries Search:

Stereo — averaged over all sequences

Method	Date	Type	#kp	MS	$mAP^{5°}$	$mAP^{10°}$	$mAP^{15°}$	$mAP^{20°}$	$mAP^{25°}$
SIFT + ContextDesc + Inlier Classification V2 kp:8000, match:custom	19-05-28	F/M	7515.2	0.3633	0.0016	0.0217	0.0823	0.1818	0.2963

[P2] Phototourism dataset — Multi-view task

Performance in SfM reconstruction, averaged over all the test sequences.

- Click here for a breakdown by sequence
- Click here for a breakdown by subset size

Show 10 entries Search:

MVS — averaged over all sequences

Method	Date	Type	Ims (%)	#Pts	SR	TL	$mAP^{5°}$	$mAP^{10°}$	$mAP^{15°}$	$mAP^{20°}$	$mAP^{25°}$	ATE
SIFT + ContextDesc + Inlier Classification V2 kp:8000, match:custom	19-05-28	F/M	98.6	6126.0	97.5	3.44	0.5755	0.6830	0.7389	0.7750	0.8006	—

Examples of image pairs – nothing super difficult

Q

map5

map10

map15

map 25

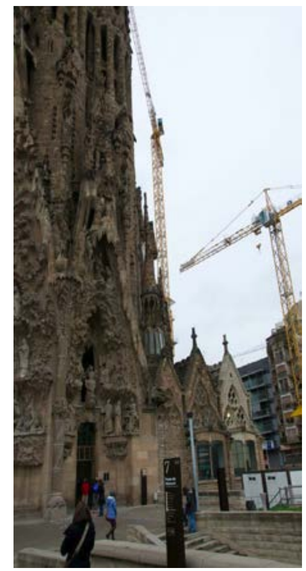
Examples of image pairs

Q

map5

map10

map15



map 25

Examples of image pairs

Q

map5

map10

map 35

What are the differences in Stereo vs. SfM evaluatin?

Stereo: features ⇨ matching ⇨ OpenCV RANSAC ⇨ pose estimation

SfM: features ⇨ matching ⇨ COLMAP RANSAC + bundle adjustment ⇨ pose estim.

Participants

Hidden, organizers

Seems that there is a problem with RANSAC or its parameters.

(not visible nor tunable by participants)

Our changes in camera pose estimation in evaluation

Before: normalize keypoints by K
and run
RansacE (threshold hard to interpret)

After: run RansacF (threshold in pixels)
get E from F by formula E = K' F K

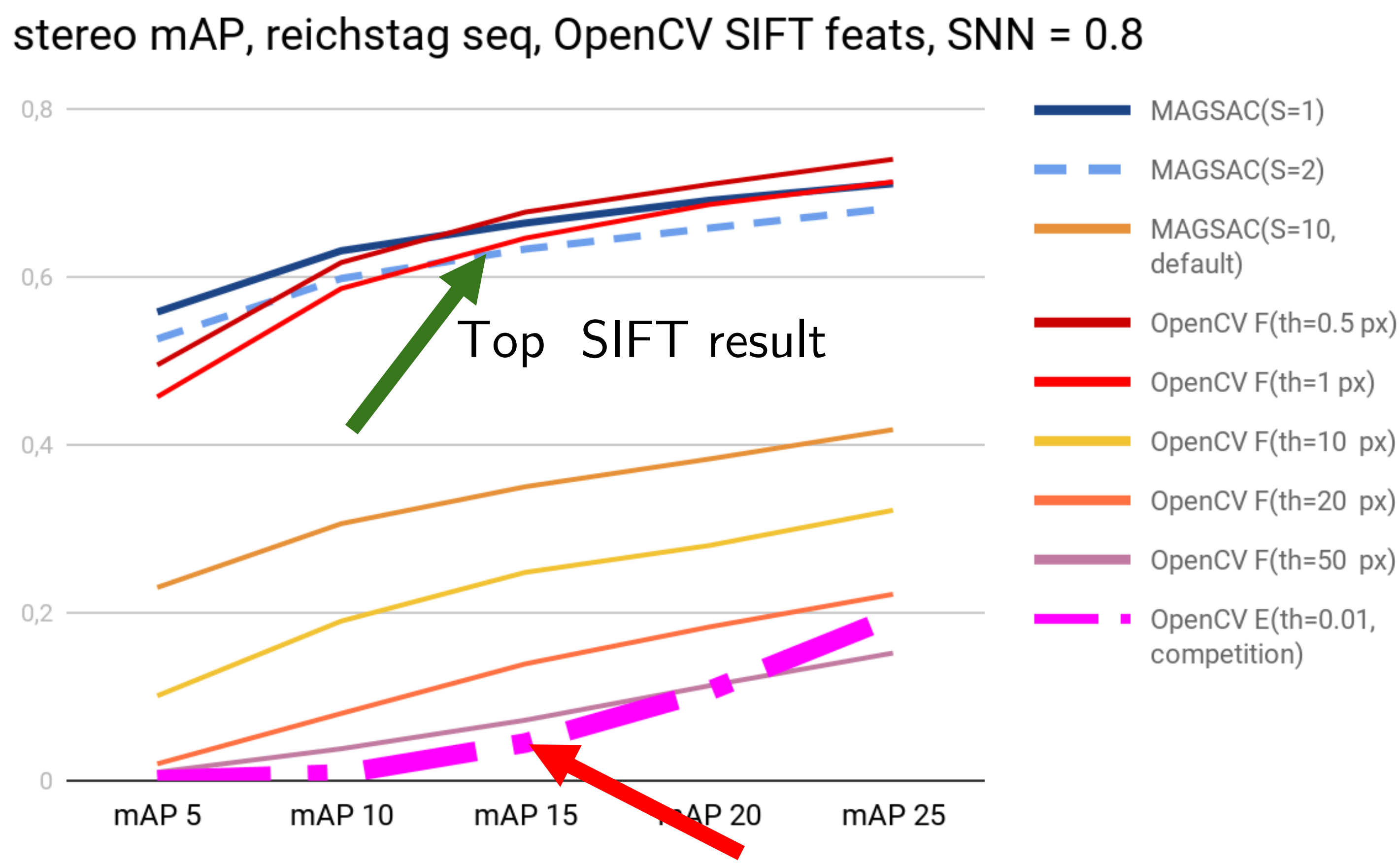
```
def normalize_keypoints(keypoints, image_shape, K):
    C_x = (image_shape[1] - 1.0) * 0.5
    C_y = (image_shape[0] - 1.0) * 0.5
    # Correct coordinates using K
    C_x += K[0, 2]
    C_y += K[1, 2]
    f_x = K[0, 0]
    f_y = K[1, 1]
    keypoints = (keypoints - np.array([[C_x, C_y]])) / np.array([[f_x, f_y]])

    return keypoints

def eval_decompose():
    ...
    kp1 = normalize_keypoints(kp1, img1_shp, calc1["K"])
    kp2 = normalize_keypoints(kp2, img2_shp, calc2["K"])
    E, mask_new = cv2.findEssentialMat(
        kp1, kp2, method=method, threshold=0.01)
    ...
```

```
def eval_decompose_F():
    ...
    F, mask_new = cv2.findFundamentalMat(
                kp1, kp2, method, 1.0,0.99)
    E = np.matmul(np.matmul(K2.T,F),K1)
```

K =[[866, 0 , 505.5],
[0 , 866 , 379],
[0 , 0 , 1]]

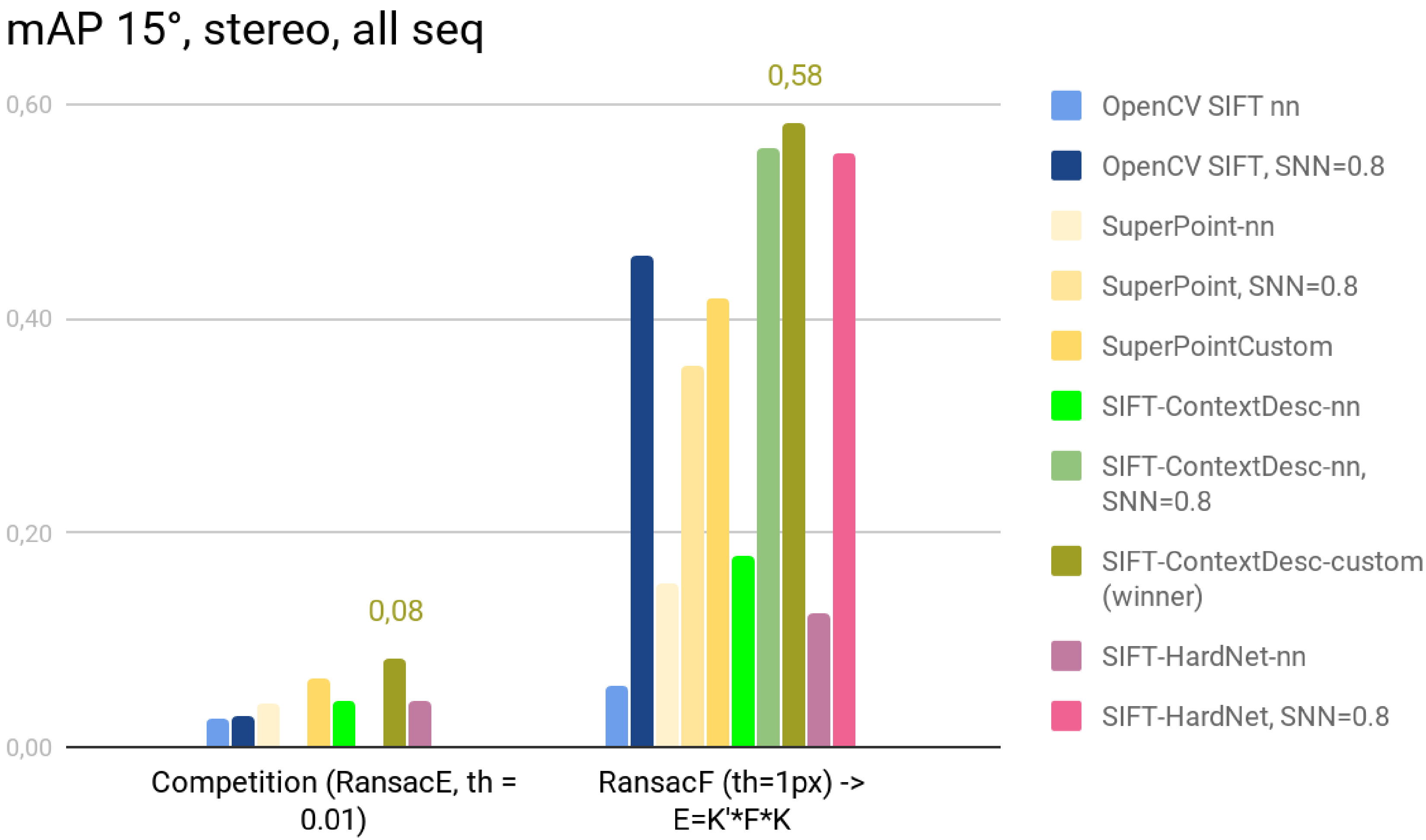
det(K)^(⅓.) = 58

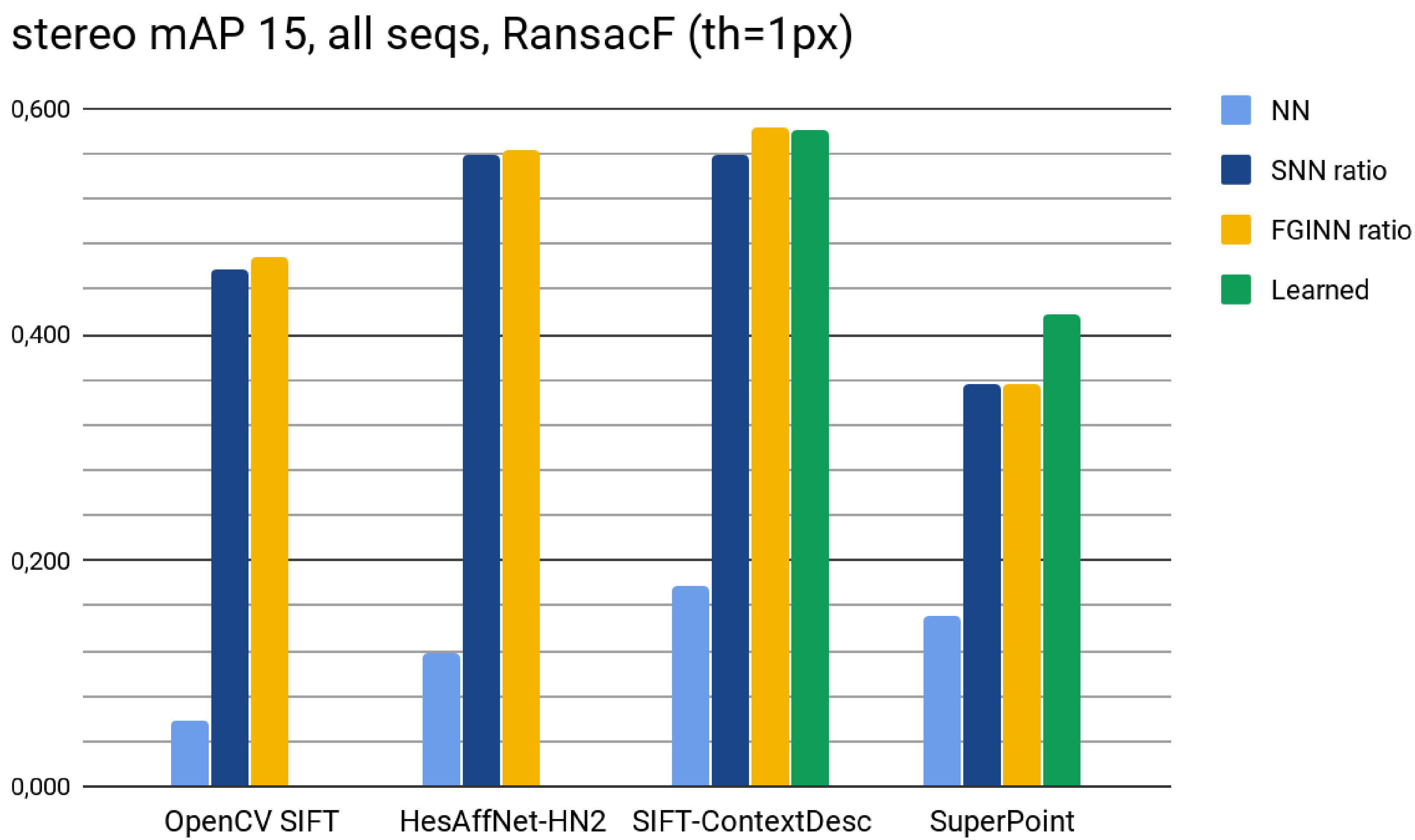
Pose precision, recovered by the competition procedure for SIFTs – The OpenCV detector and descriptor



Re-evaluated results: everyone benefits

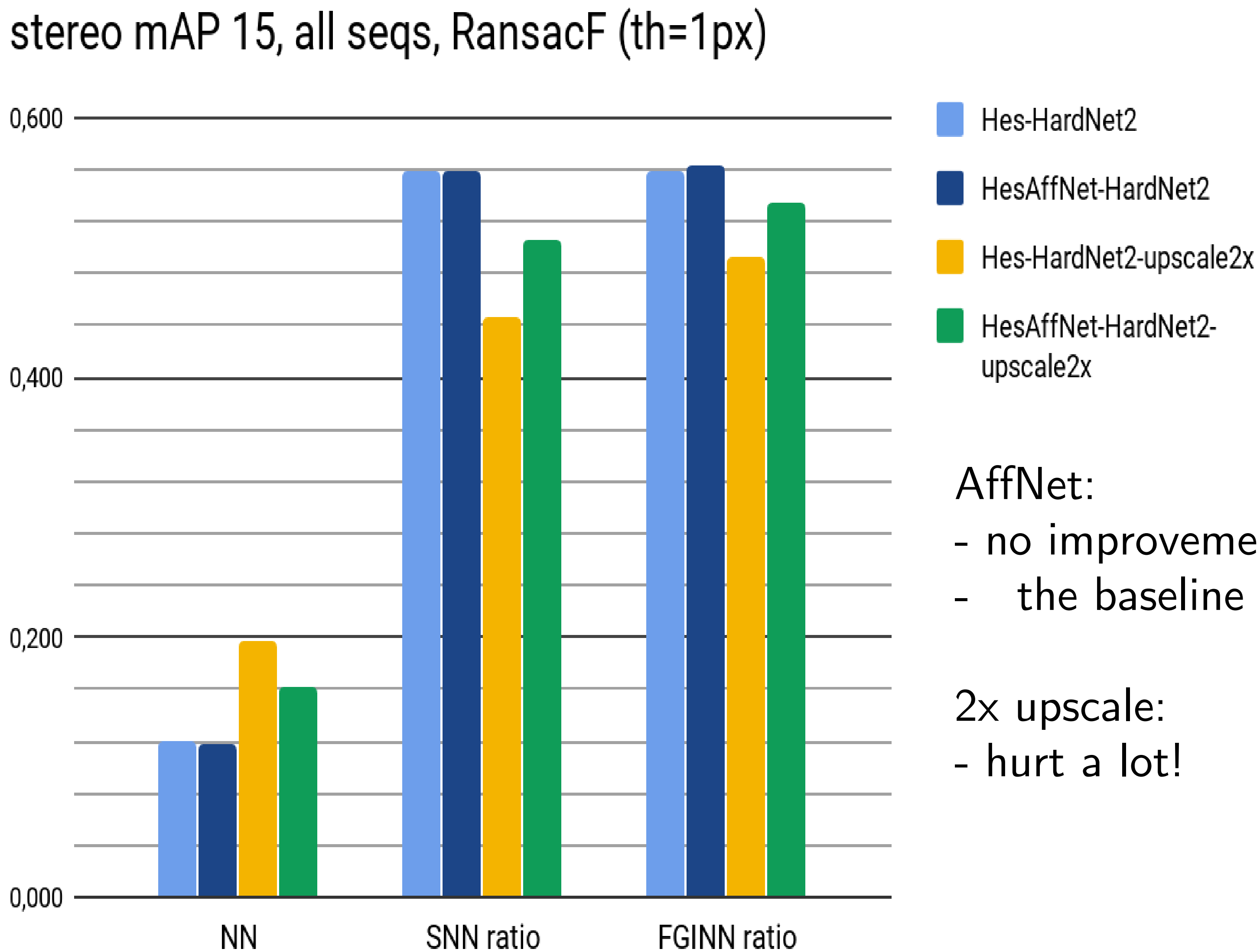
- Winner is the same,
- Ratio test is super important
- SIFT > SuperPoint now.
- HardNet is a strong baseline





Learned
Moo Yi, Trulls, Ono, Lepetit, Salzmann, Fua:
Learning to Find Good Correspondences, CVPR 2018

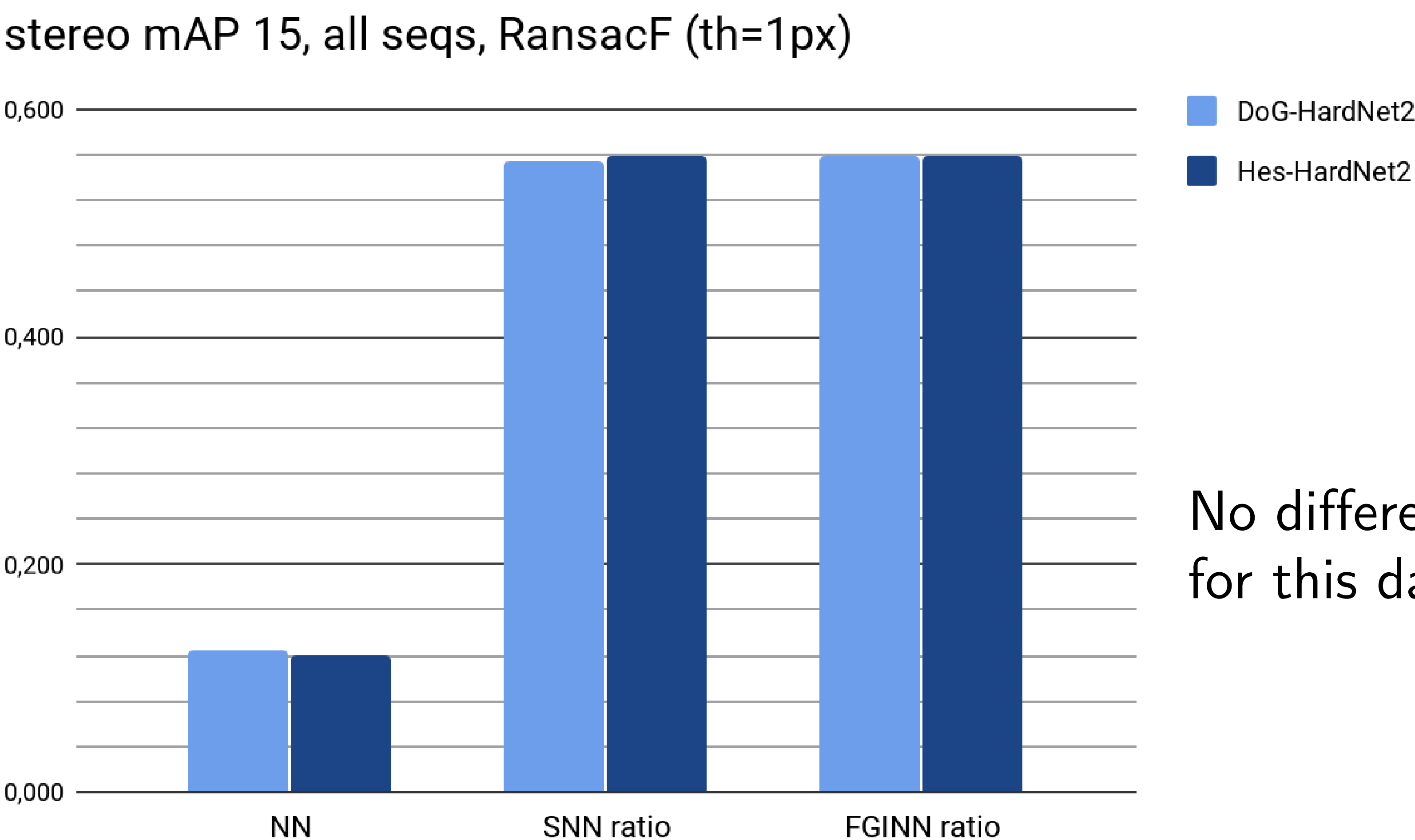
CMP Lessons: Does AffNet help?



AffNet:
- no improvement, no loss
- the baseline is narrow here

2x upscale:
- hurt a lot!

CMP Lessons: Does Hessian vs DoG (SIFT) help?



No difference
for this dataset

AffNet: learning measurement region

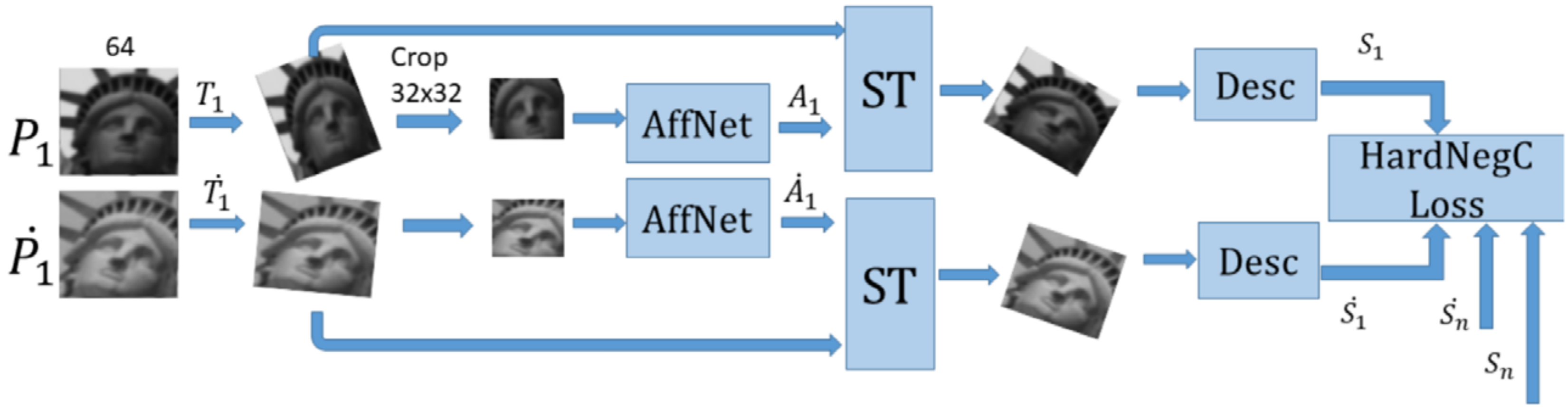
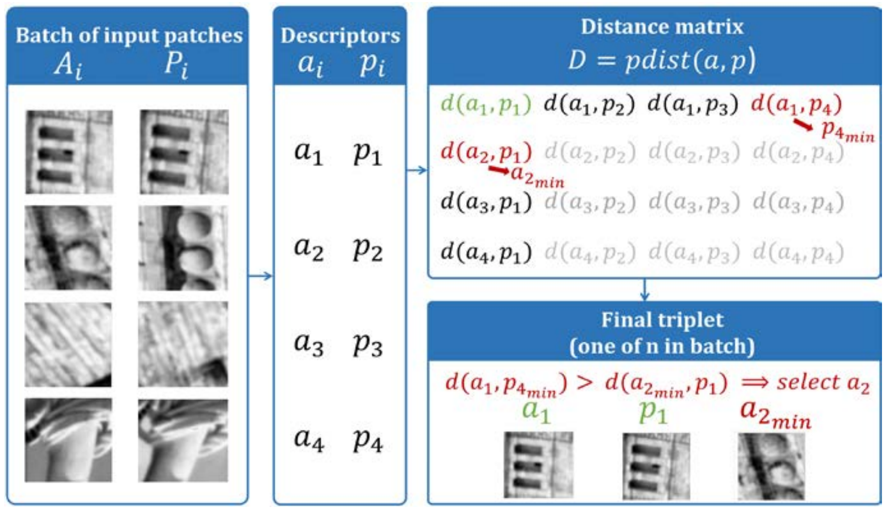


Fig. 5. AffNet training. Corresponding patches undergo random affine transformation $T_i, \dot{T}_i$, are cropped and fed into AffNet, which outputs affine transformation $A_i, \dot{A}_i$ to an unknown canonical shape. ST – the spatial transformer warps the patch into an estimated canonical shape. The patch is described by a differentiable CNN descriptor. $n \times n$ descriptor distance matrix is calculated and used to form triplets, according to the HardNegC loss.

$$L = \frac{1}{n} \sum_{i=1,n} \max\left(0, 1 + d(s_i, \dot{s}_i) - d(s_i, N)\right), \quad \frac{\partial L}{\partial N} := 0,$$

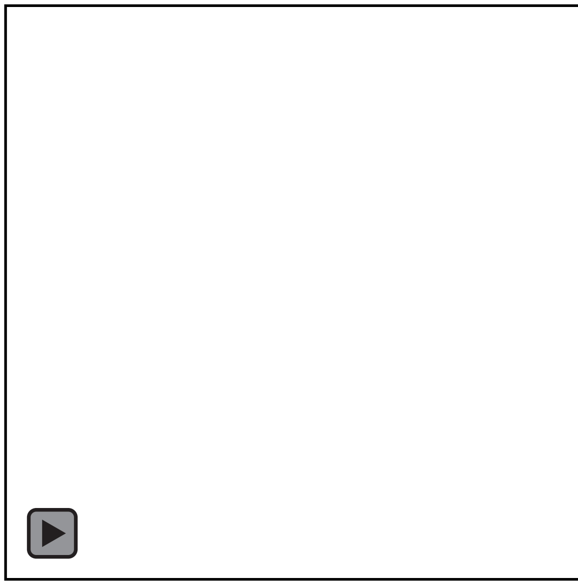
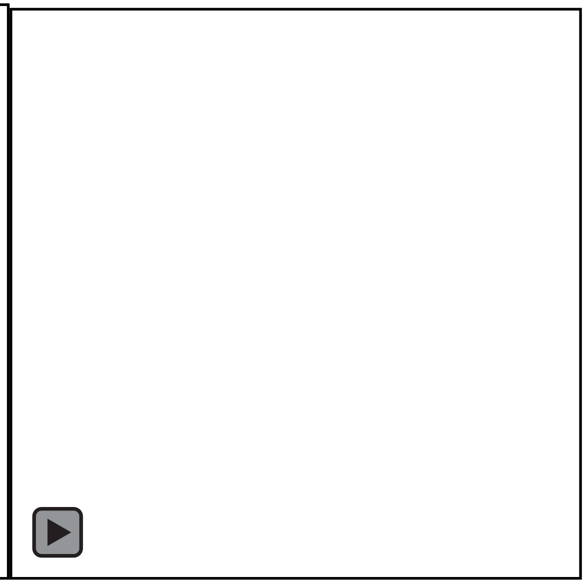
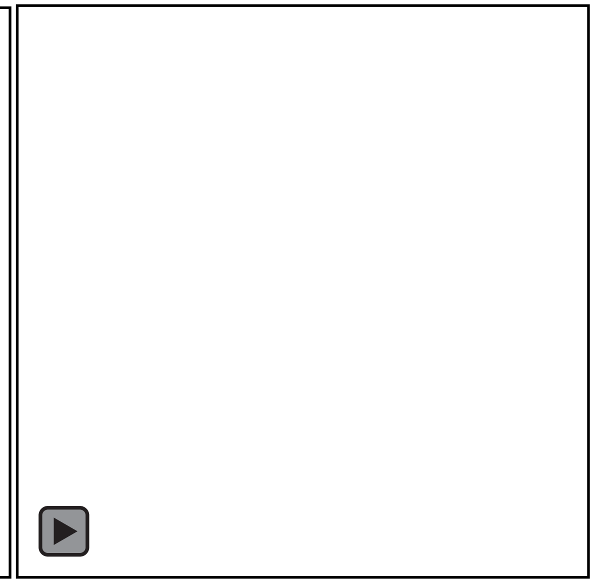
Mishkin et.al. Repeatability Is Not Enough: Learning Affine Regions via Discriminability. ECCV 2018

HardNegC loss: treat negative example as constant



$$L = \frac{1}{n} \sum_{i=1,n} \max\left(0, 1 + d(s_i, \dot{s}_i) - d(s_i, N)\right), \qquad \frac{\partial L}{\partial N} := 0,$$

Why HardNegC loss is needed?

Lessons Learned from the CMP IMW Submission:

- Good and properly set RANSAC is extremely important

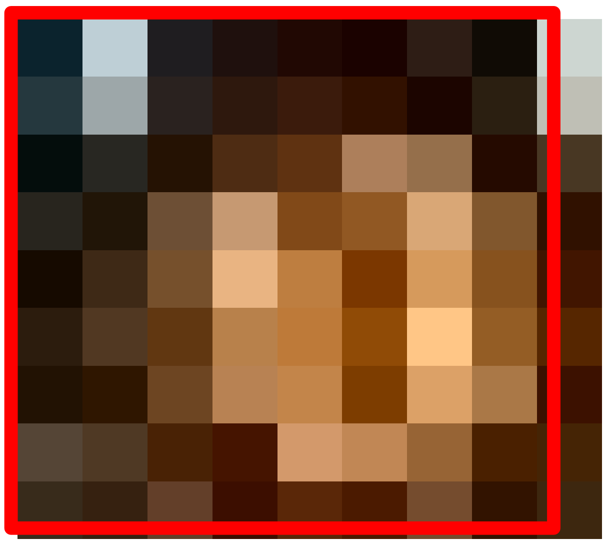
- Neither SNN ratio test, nor good RANSAC working on its own
- SNN + good RANSAC is a powerful combination

- FGINN > SNN, use it
- Learning to match gives a moderate boost over SNN
- DoG/Hessian + HardNet + FGINN is very competitive and simple baseline

- AffNet does't harm, potenitally helps for difficult to connect image

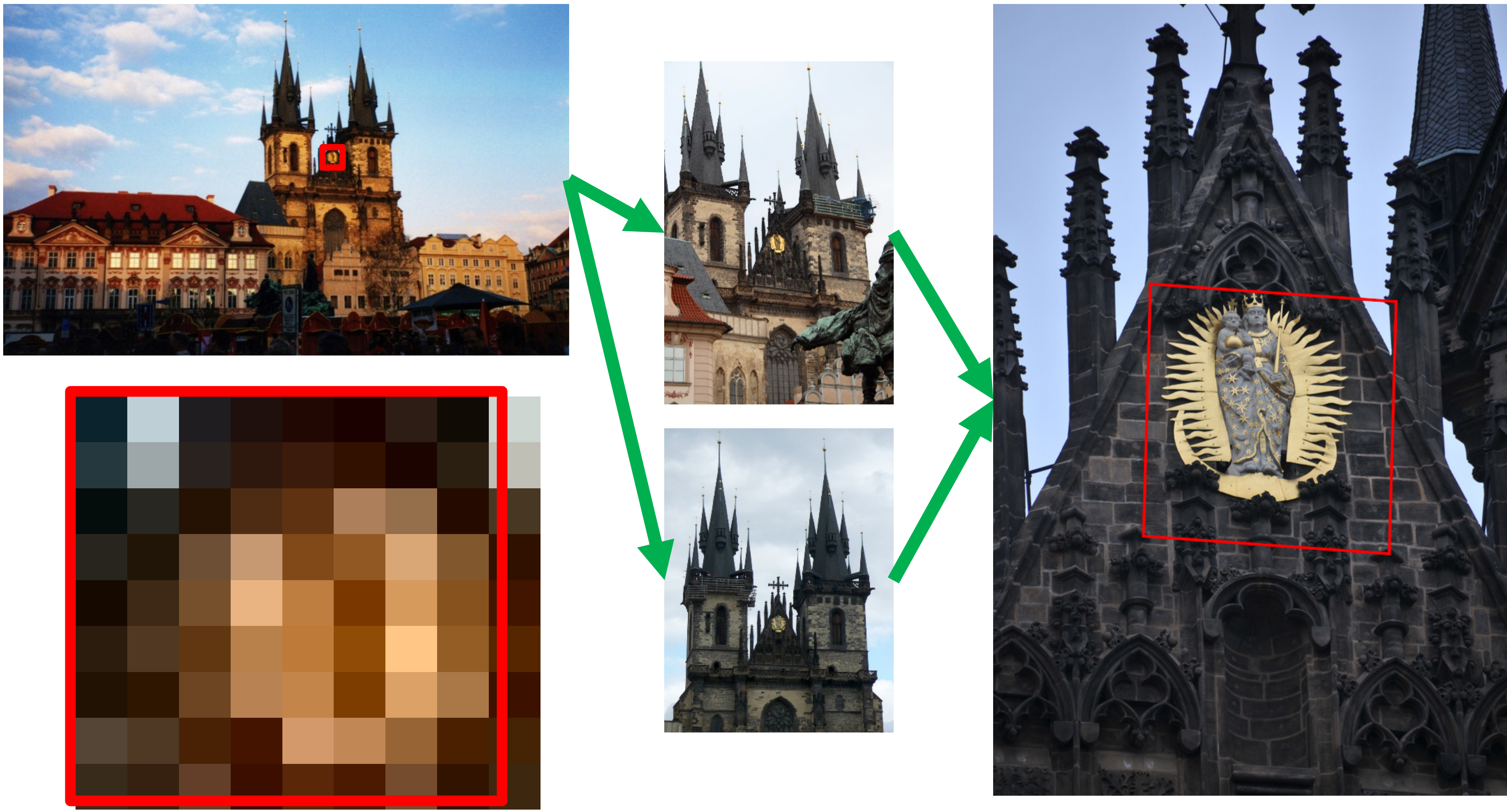
The Correspondence Problem - Challenges

Matching in the context of other images

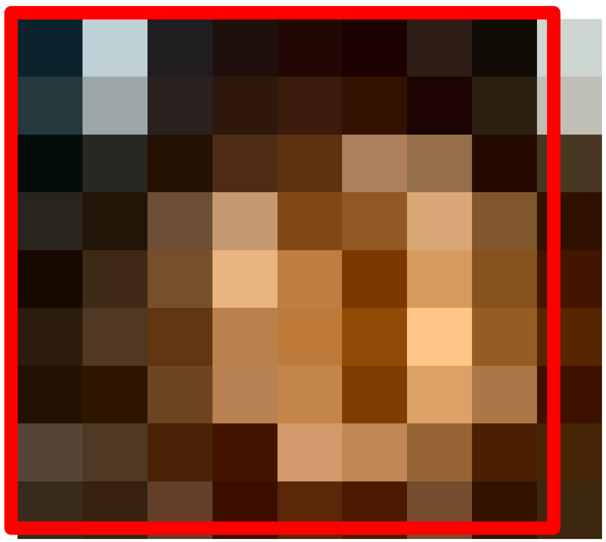


?

Matching in the context of other images



Matching in the context of other images



Finding correspondences

For a large viewpoint change (including scale)
=>
the wide-baseline stereo problem

Applications:

- pose estimation
- 3D reconstruction
- location recognition

Finding correspondences

for large viewpoint change
(including scale)
=>
the wide-baseline (WBS)
stereo problem

Finding correspondences

for large

illumination change

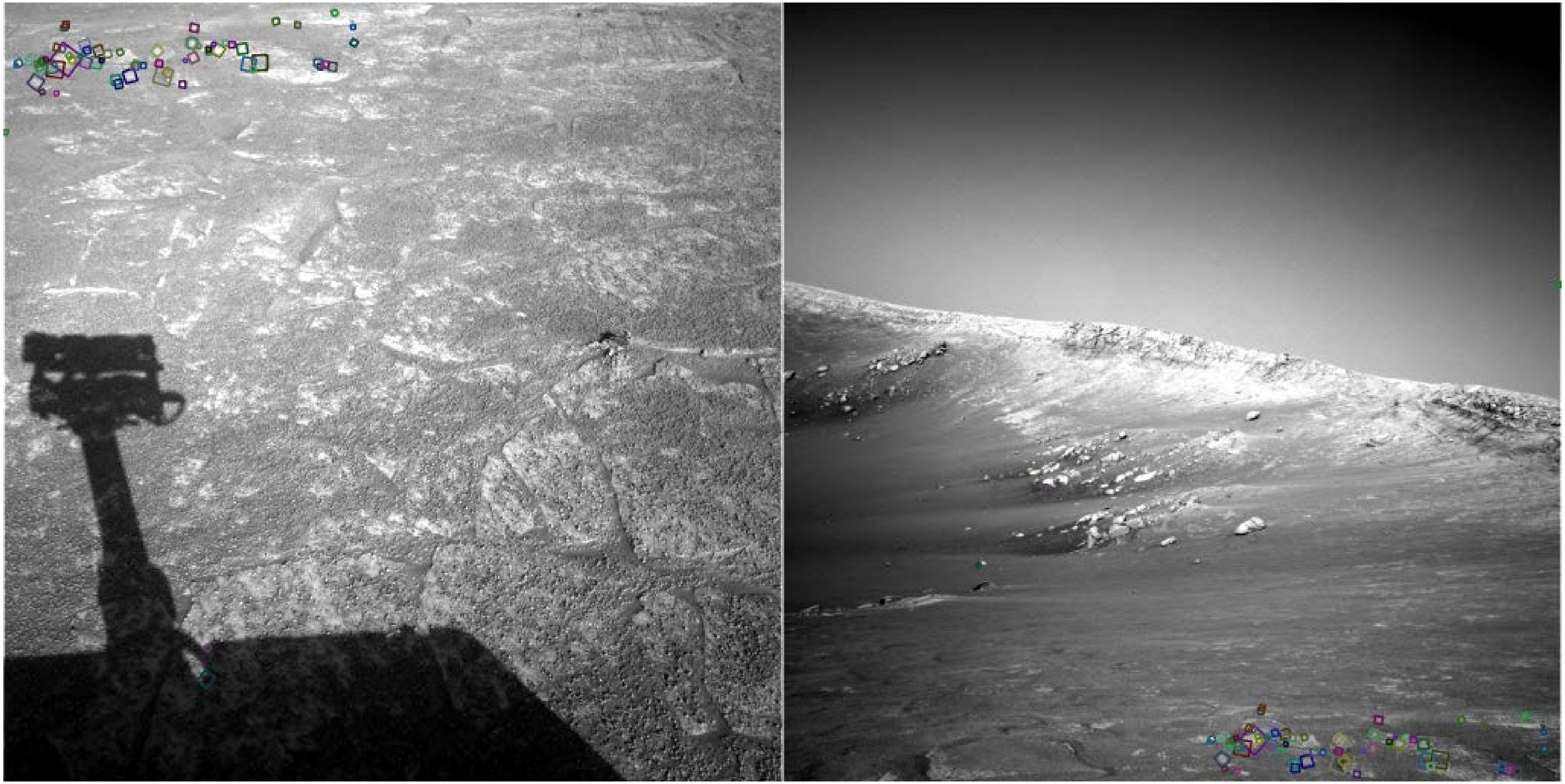
=>

wide “illumination-baseline” stereo problem

Applications:

- location recognition
- summarization of image collections

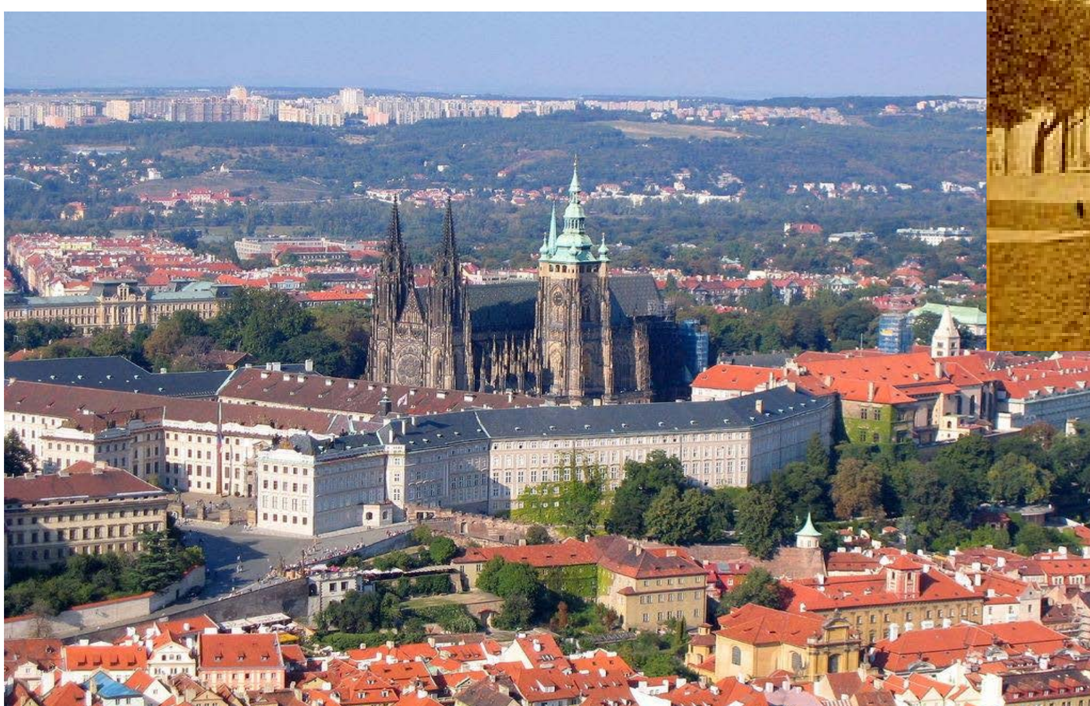
Find the matches (look for tiny colored squares...)



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

Finding correspondences

For large
time difference
=>
wide temporal-baseline
stereo problem



Applications:

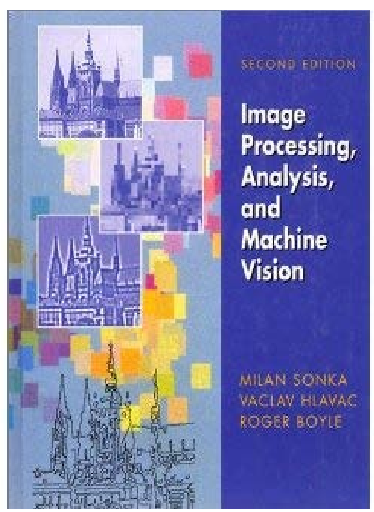
- historical reconstruction
- location recognition
- photographer recognition
- camera type recognition

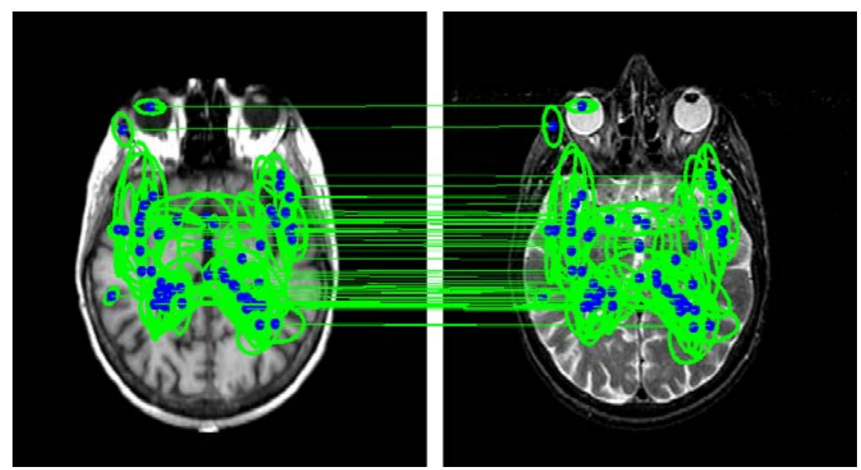
Finding Correspondences

change of modality

Applications:

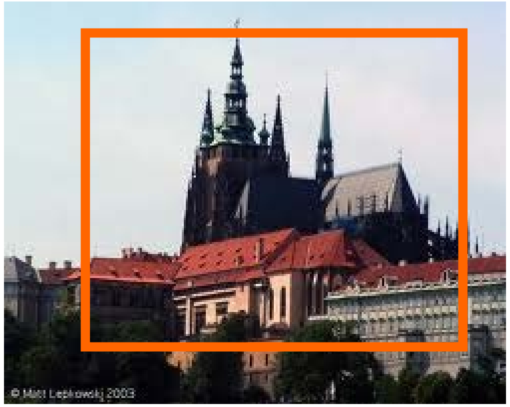
- medical imaging
- remote sensing



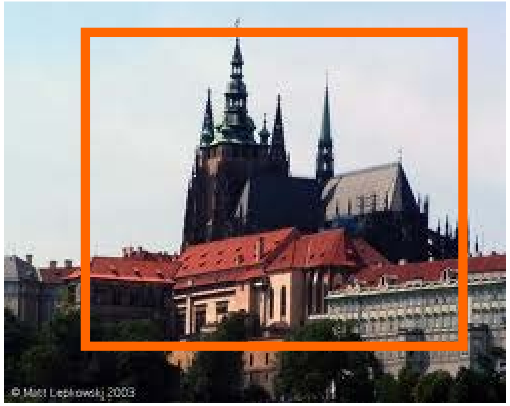


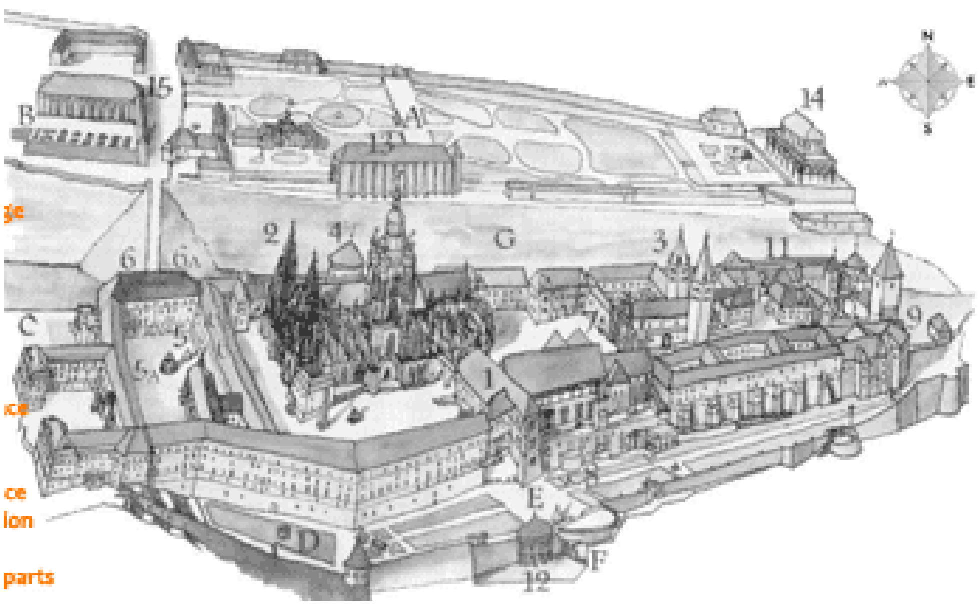
with occlusion “almost everywhere”

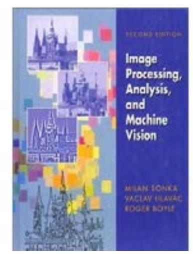
"Inprecise" Geometry ☺

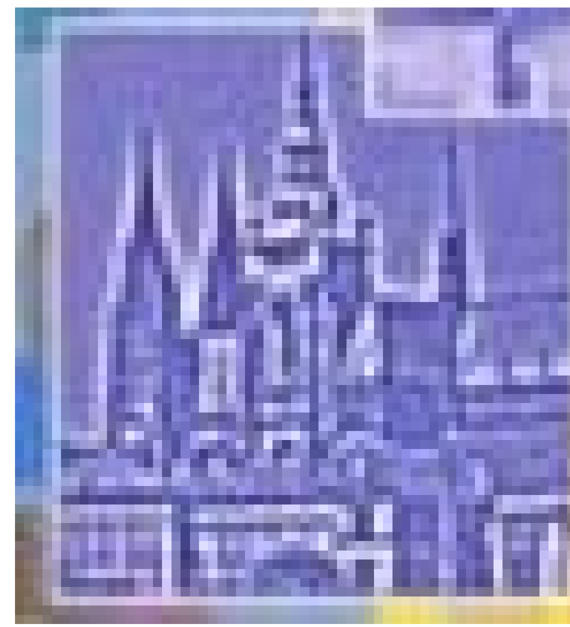


Retrieving different modalities









Thank you!