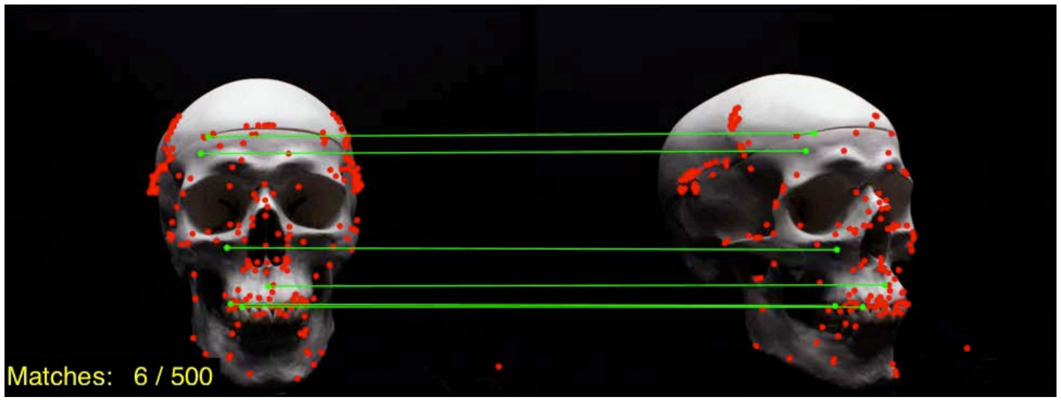
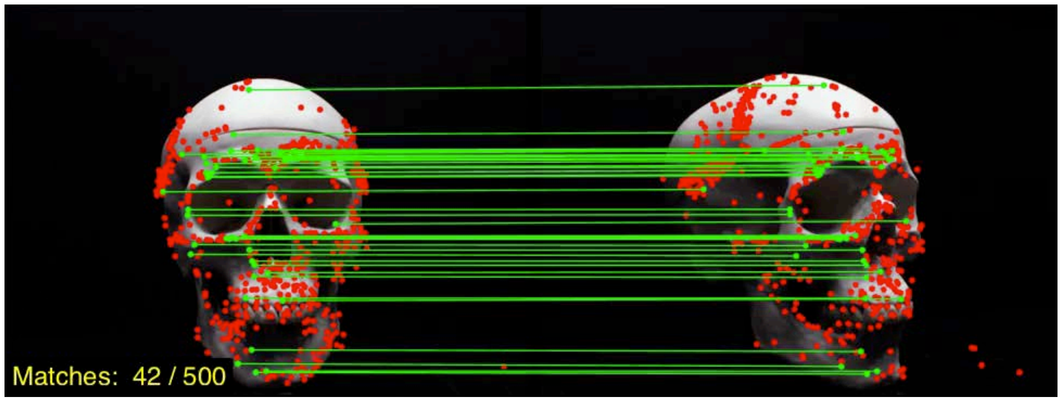
From Invariant Descriptors to Deep Pose Estimation

K. Yi, E. Trulls, V. Lepetit, and P. Fua

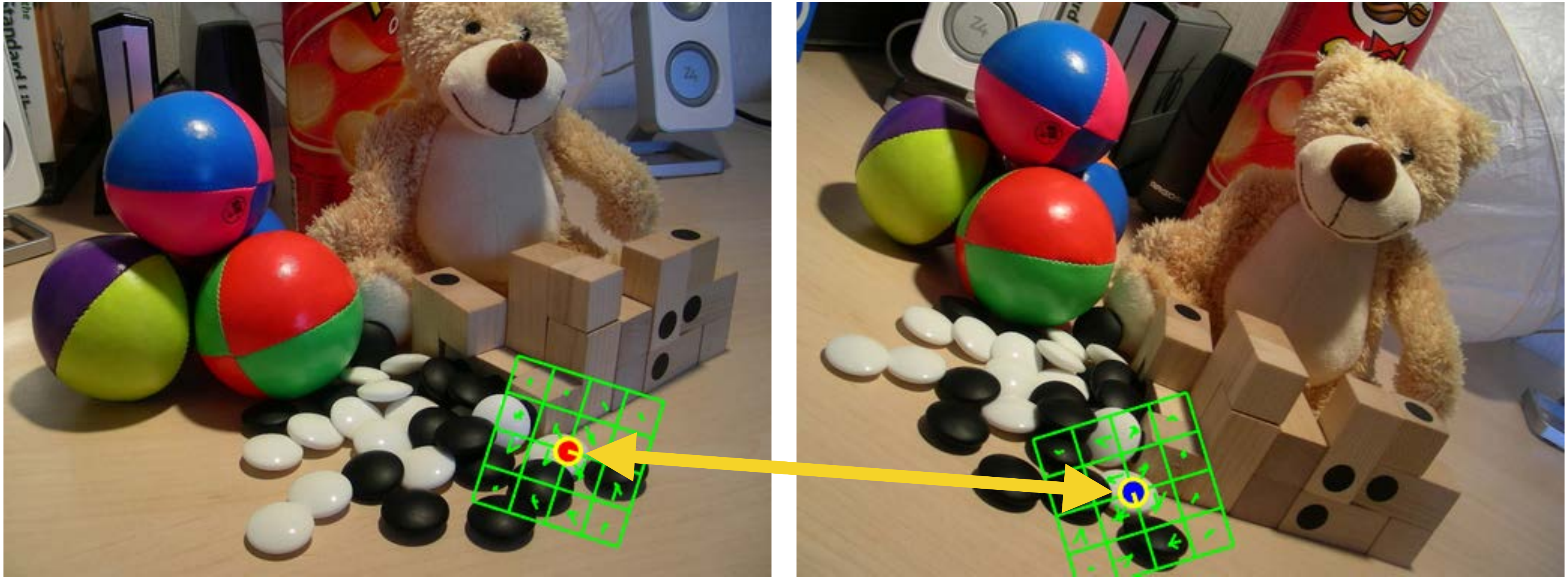


SIFT



LIFT

Feature Points

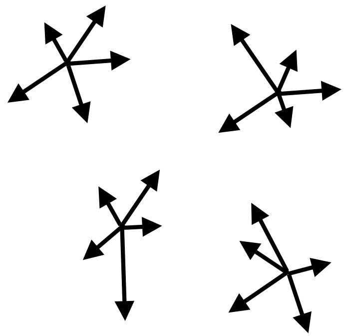


Outstanding tool for matching points across images.

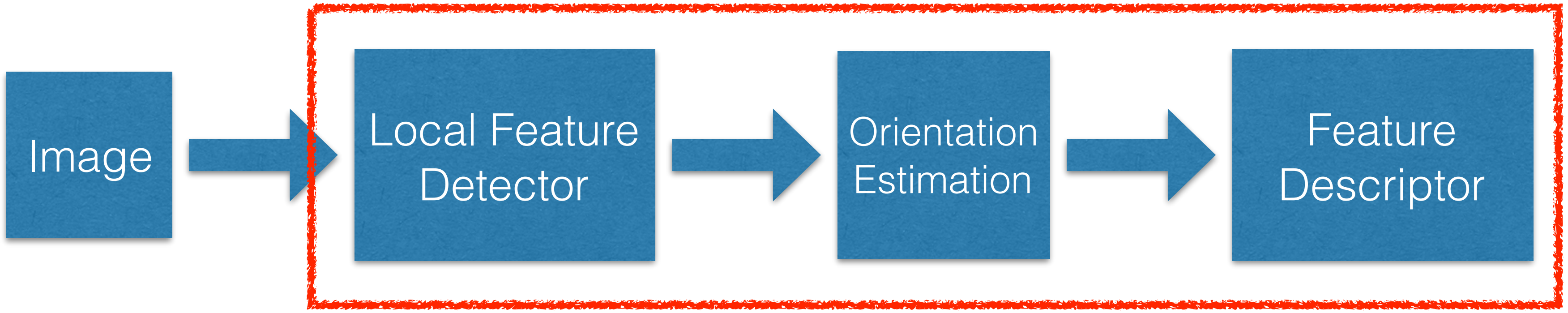
SIFT (Lowe, ICCV'99) started the trend: ~48k citations.

Local Feature Pipeline

Image → Local Feature Detector → Orientation Estimation → Feature Descriptor

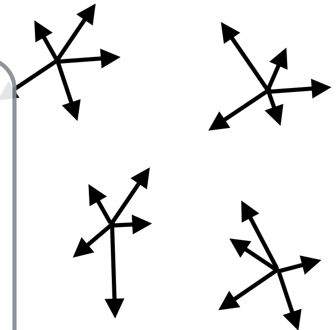


Local Feature Pipeline



- Lowe, D., "Distinctive Image Features from Scale-Invariant Keypoints," IJCV, 2004
- Bay, H., Ess, A., Tuytelaars, T., Van Gool, L, "SURF: Speeded Up Robust Features," CVIU, 2008
- Rublee, E., Rabaud, V., Konolidge, K., Bradski, G, "ORB: An Efficient Alternative to SIFT or SURF," ICCV, 2011
- Alcantarilla, P.F., Neuvo, J., Bartoli, A., "Fast explicit diffusion for accelerated features in nonlinear scale spaces," TPAMI, 2011
- Leutenegger, S., Chli, M., Siegwart, R., "BRISK: Binary Robust Invariant Scalable Keypoints", ICCV, 2011
- Alcantarilla, P.F., Bartoli, A. Davison, A.J., "KAZE features," ECCV, 2012

⋮



Local Feature Pipeline

Image → Local Feature Detector → Orientation Estimation → Feature Descriptor

Local Feature Detector

- Harris, C., Stephens, M., "A Combined Corner and Edge Detector,"AVC, 1988
- Mikolajczyk, K., Schmid, C., "Scale and Affine Invariant Interest Point Detectors," IJCV, 2004
- Förstner, W., Dickscheid, T., Schindler, F., "Detecting Interpretable and Accurate Scale-Invariant Keypoints," ICCV, 2009
- Rosten, E., Porter, R., Drummond, T., "Faster and Better: A Machine Learning Approach to Corner Detection," TPAMI, 2010
- Zitnick, C., Ramnath, K., "Edge Foci Interest Points", ICCV, 2011
- Mainali, P., Lafruit, G., Tack, K., Van Gool, L., Lauwereins, R., "Derivative-Based Scale Invariant Image Feature Detector with Error Resilience", TPAMI, 2014

⋮

Orientation Estimation

- Gauglitz, S., Turk, M., Höllerer, T., "Improving Keypoint Orientation Assignment,", BMVC, 2011

Heuristics…

- Dominant Gradient Orientations (SIFT, SURF,…)
- Center of Mass (ORB,…)

Feature Descriptor

- Winder, S., Brown, M., "Learning Local Image Descriptors," CVPR, 2007
- Tola, E., Lepetit, V., Fua, P., "A Fast Local Descriptor for Dense Matching," CVPR, 2008
- Fan, B., Wu, F., Hu., Z., "Aggregating Gradient Distributions into Intensity Orders: A Novel Local Image Descriptor," CVPR, 2011
- Alahi, A., Ortiz, R., Vandergheynst, P., "FREAK: Fast Retina Keypoint," CVPR, 2012
- Simonyan, K., Vedaldi, A., Zisserman, A., "Learning Local Feature Descriptors Using Convex Optimisation," TPAMI, 2014
- Zagoruyko, S., Komodakis, N., "Learning to Compare Image Patches via Convolutional Neural Networks,", CVPR, 2015

⋮

Deep Learning Revolution

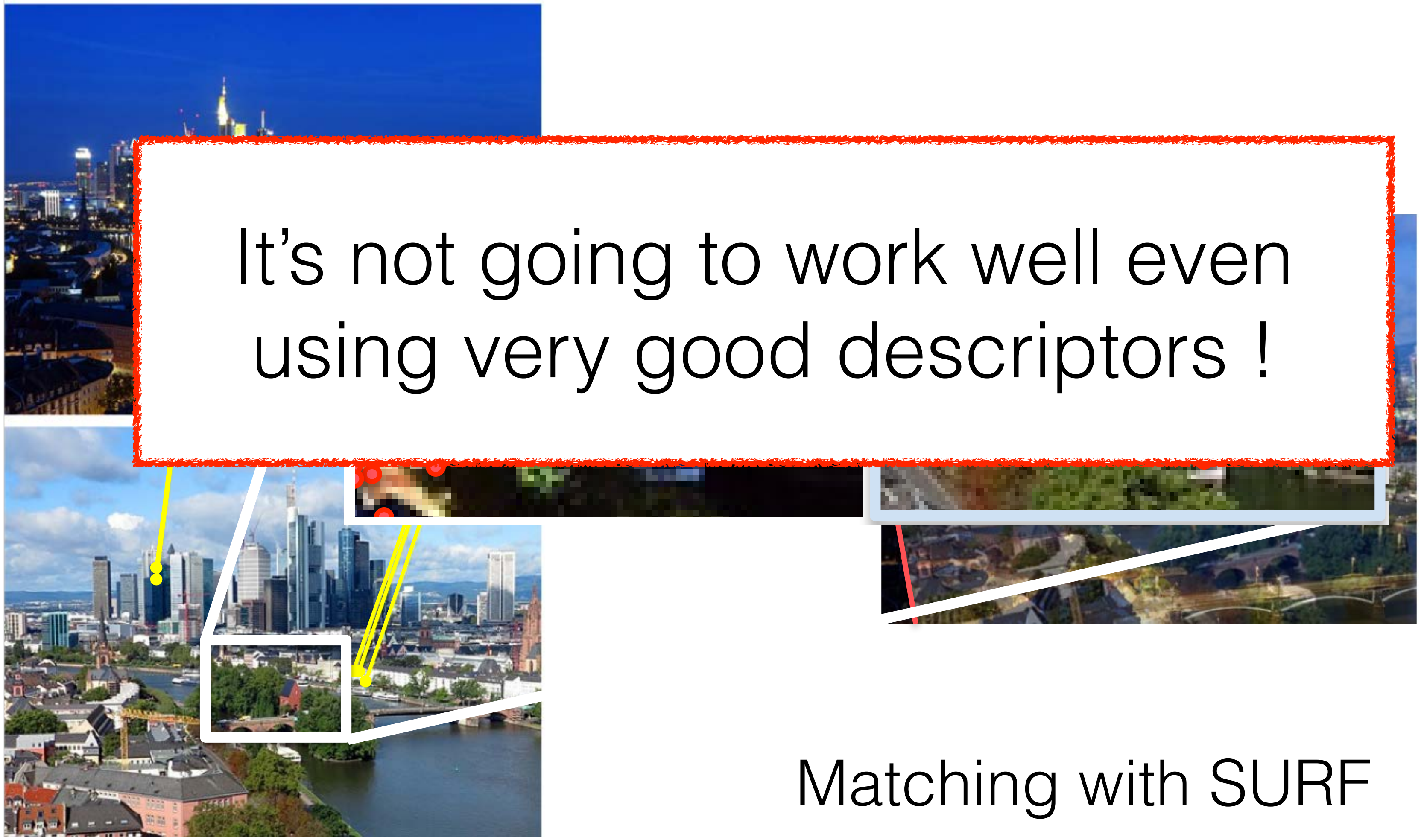
An opportunity to revisit and improve the pipeline:

- Reformulate its different components in terms of CNNs.
- Integrate them into a fully differentiable pipeline.
- Optimize them jointly.

1. Detecting Keypoints

TILDE: a Temporally Invariant Learned DEtector
(CVPR 2015)

Hand-Designed Features under Severe Illumination Changes

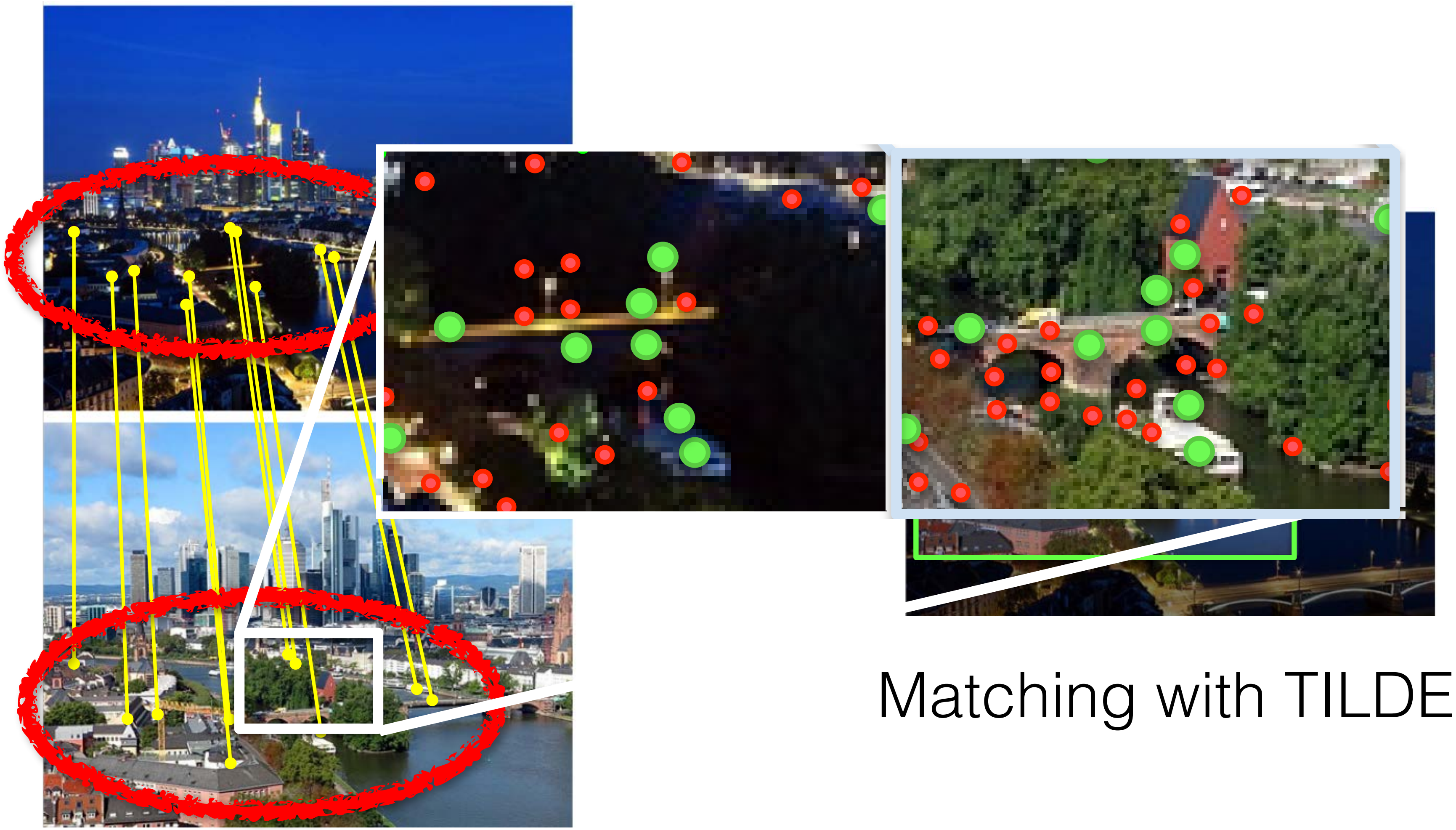


It's not going to work well even using very good descriptors !

Matching with SURF

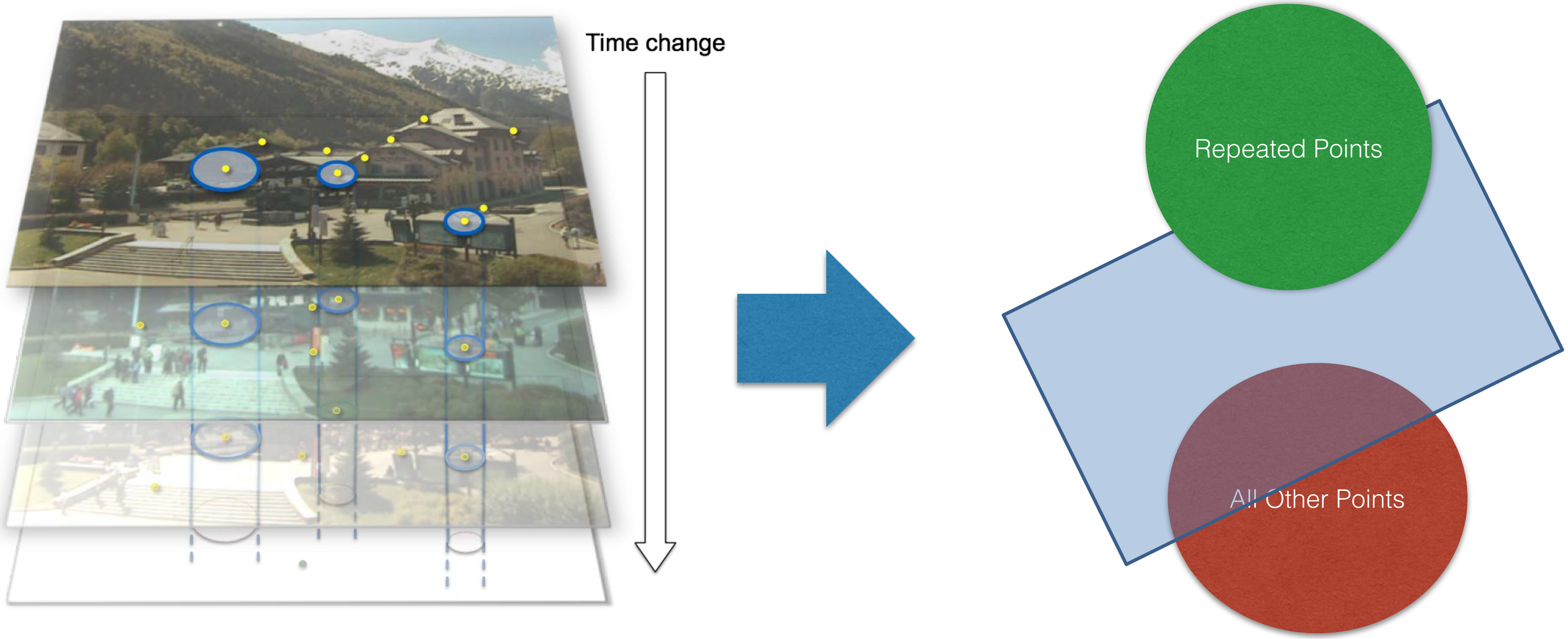
—> Poor repeatability.

Learning to find Keypoints that Are Robust to Illumination Changes.



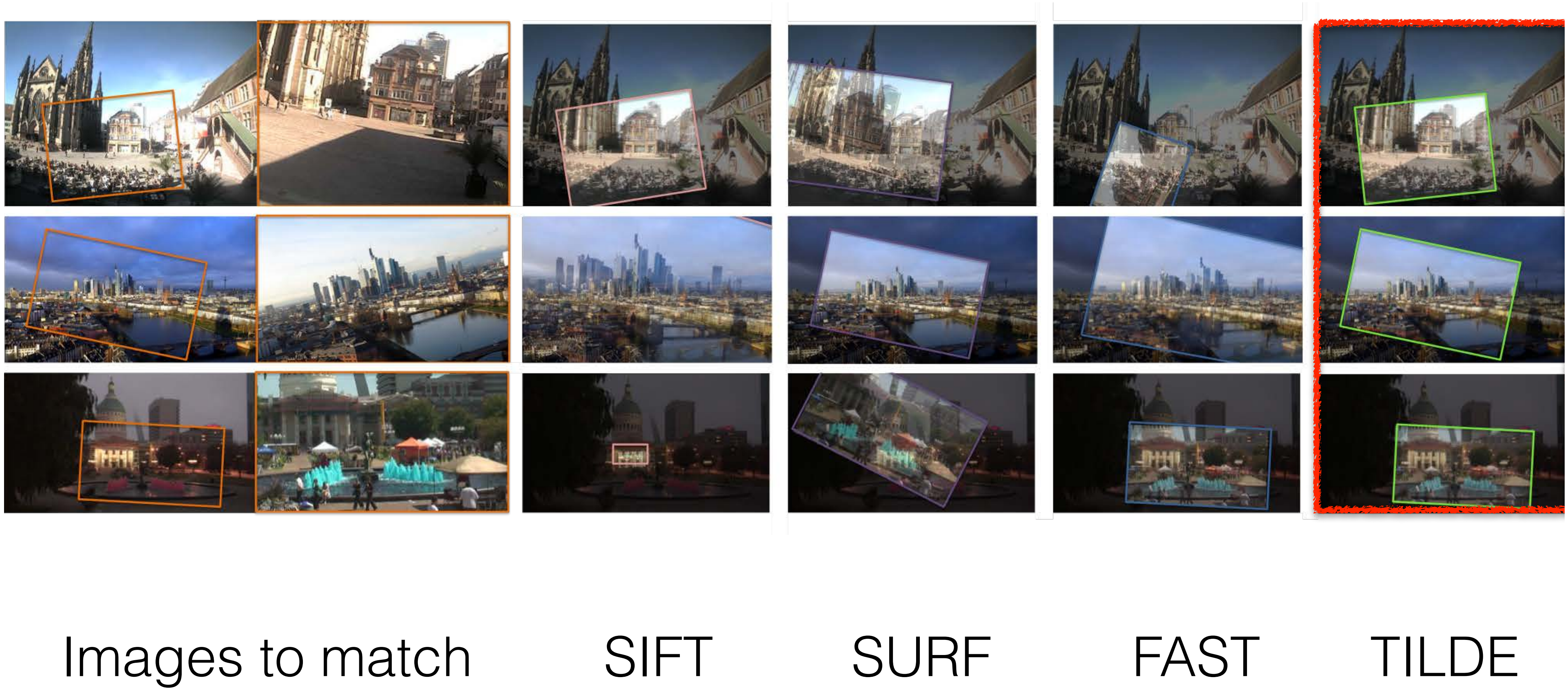
Matching with TILDE

Learning from Aligned Image Stacks



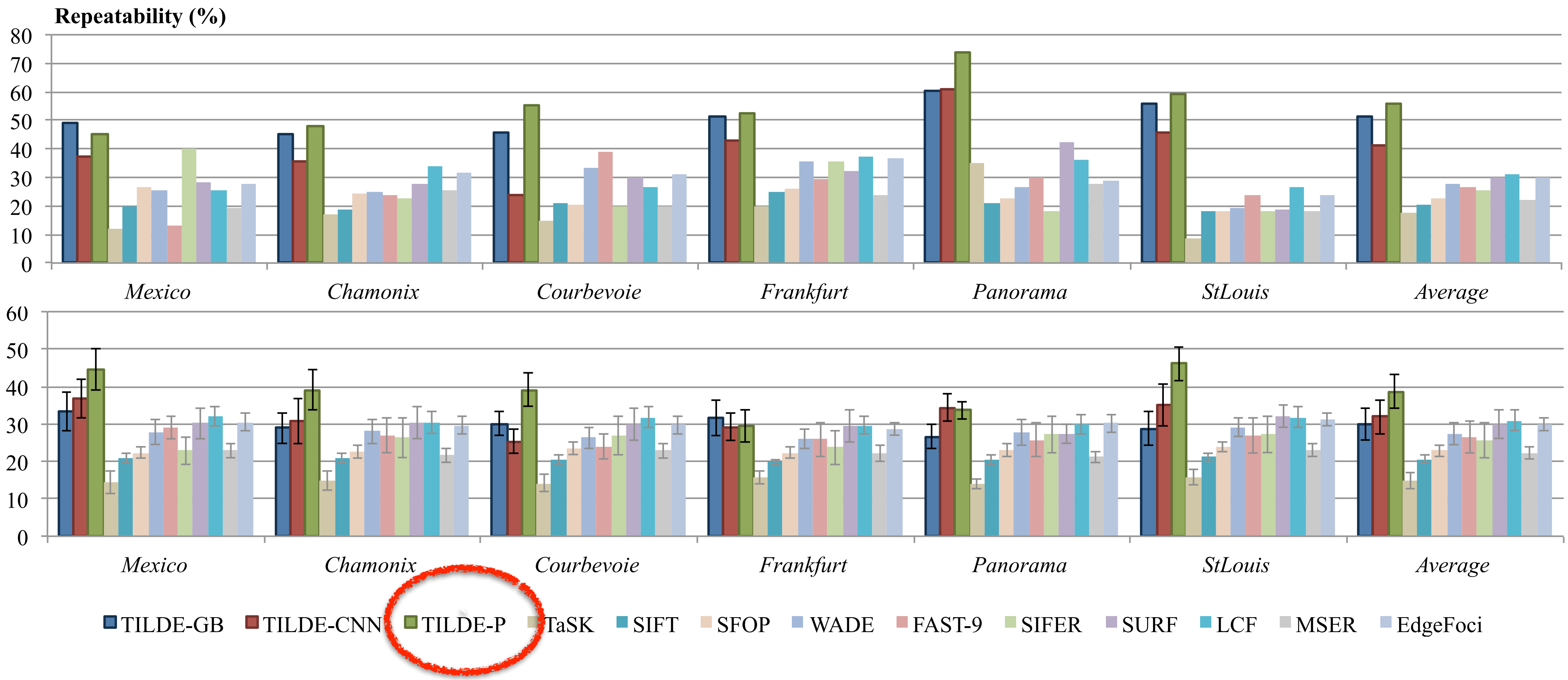
- Pre-align images of a scene.
- Find locations that are often detected by a given feature detector.
- Train a CNN regressor to find these locations.

Examples



Matching 5 days of
the *Frankfurt* sequence
with our keypoints

Quantitative Results
Webcam Dataset



Keypoint Detection in Short

- Keypoint repeatability is crucial for many applications
- We can train a regressor to find repeatable keypoints.

2. Estimating Orientation

Learning to Assign Orientations to Feature Points
(CVPR 2016)

Local Feature Pipeline

Image → Local Feature Detector → Orientation Estimation → Feature Descriptor

Local Feature Detector

- Harris, C., Stephens, M., "A Combined Corner and Edge Detector,"AVC, 1988
- Mikolajczyk, K., Schmid, C., "Scale and Affine Invariant Interest Point Detectors," IJCV, 2004
- Förstner, W., Dickscheid, T., Schindler, F., "Detecting Interpretable and Accurate Scale-Invariant Keypoints," ICCV, 2009
- Rosten, E., Porter, R., Drummond, T., "Faster and Better: A Machine Learning Approach to Corner Detection," TPAMI, 2010
- Zitnick, C., Ramnath, K., "Edge Foci Interest Points", ICCV, 2011
- Mainali, P., Lafruit, G., Tack, K., Van Gool, L., Lauwereins, R., "Derivative-Based Scale Invariant Image Feature Detector with Error Resilience", TPAMI, 2014
- Verdie, Y., Yi, K.M., Fua, P., Lepetit, V., "TILDE: A Temporally Invariant Learned DEtector,", CVPR, 2015

⋮

Orientation Estimation

- Gauglitz, S., Turk, M., Höllerer, T., "Improving Keypoint Orientation Assignment,", BMVC, 2011

Heuristics…

- Dominant Gradient Orientations (SIFT, SURF,…)
- Center of Mass (ORB,…)

Important but largely overlooked

Feature Descriptor

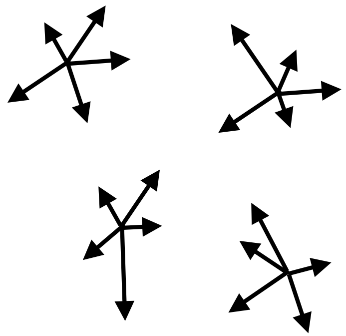
- Winder, S., Brown, M., "Learning Local Image Descriptors," CVPR, 2007
- Tola, E., Lepetit, V., Fua, P., "A Fast Local Descriptor for Dense Matching," CVPR, 2008
- Fan, B., Wu, F., Hu., Z., "Aggregating Gradient Distributions into Intensity Orders: A Novel Local Image Descriptor," CVPR, 2011
- Alahi, A., Ortiz, R., Vandergheynst, P., "FREAK: Fast Retina Keypoint," CVPR, 2012
- Simonyan, K., Vedaldi, A., Zisserman, A., "Learning Local Feature Descriptors Using Convex Optimisation," TPAMI, 2014
- Zagoruyko, S., Komodakis, N., "Learning to Compare Image Patches via Convolutional Neural Networks,", CVPR, 2015
- Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Fua, P., Moreno-Noguer, F., "Discriminative Learning of Deep Convolutional Feature Point Descriptors,", ICCV, 2015

⋮

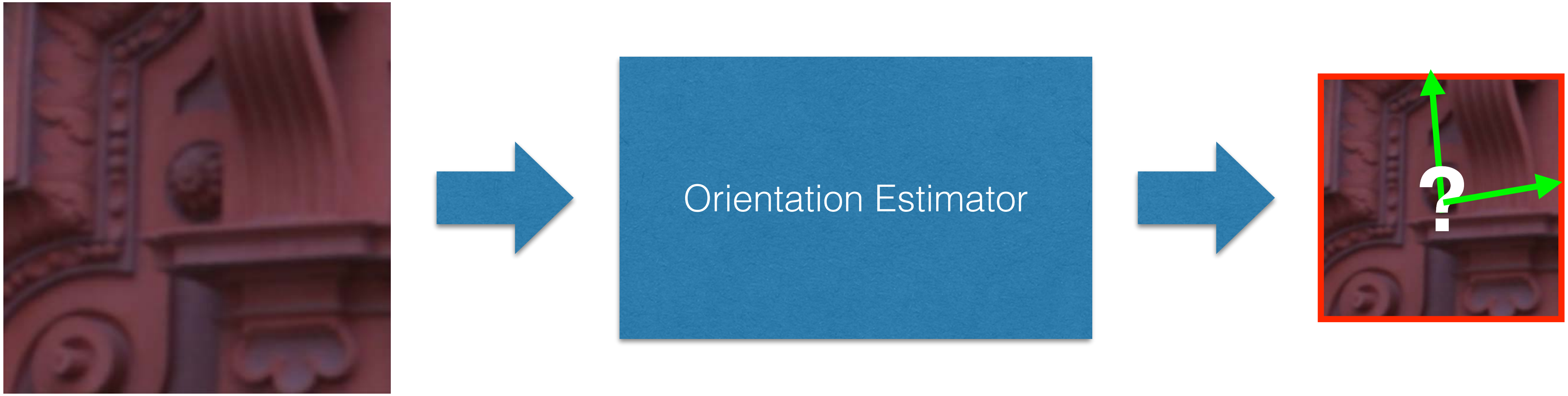
Local Feature Pipeline

Image → Local Feature Detector → Orientation Estimation → Feature Descriptor

?



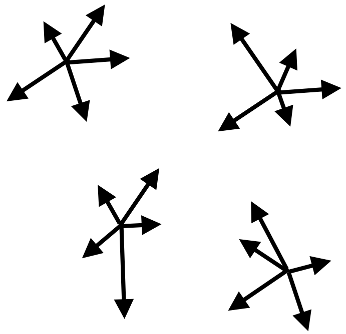
Ill-Posed Problem



There is no such thing as a **canonical orientation**

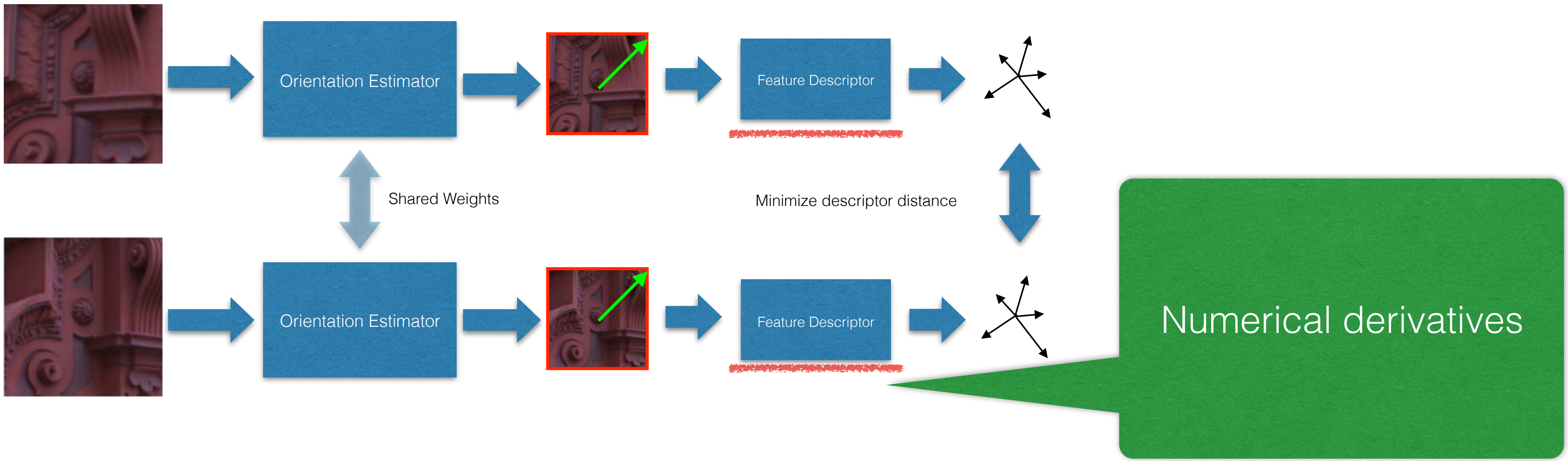
Implicit Orientations

Image → Local Feature Detector → Orientation Estimation → Feature Descriptor

Learn to estimate **consistent** and **optimal** orientations for matching purposes.

Deep Siamese Network for Learning Orientation



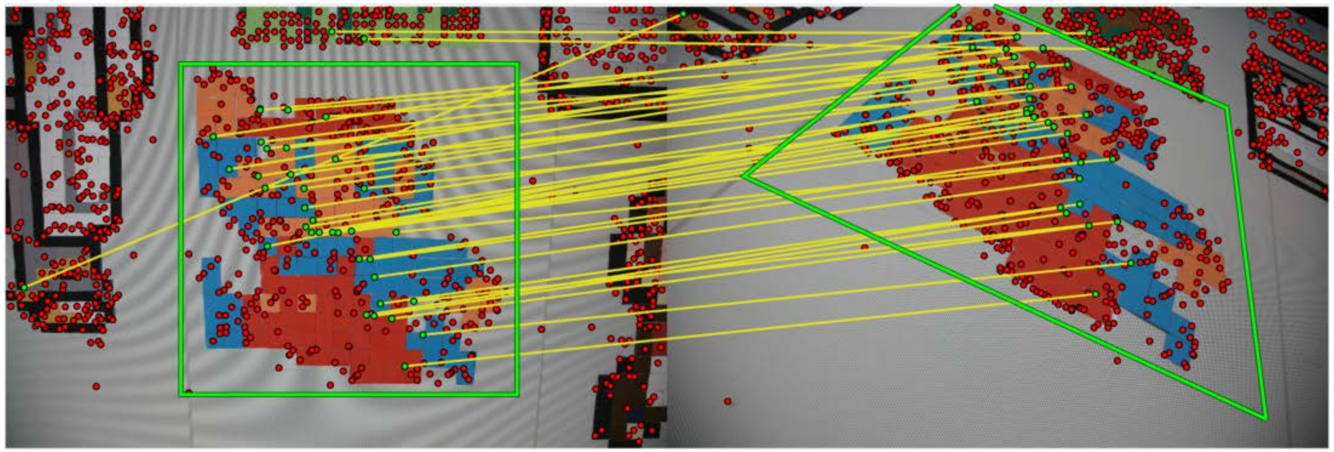
$$\text{minimize} \quad \mathcal{L}(\mathbf{Pair}) = \left\| \begin{array}{l} \mathrm{Desc}(\mathbf{Patch}_1, \mathrm{Orient}(\mathbf{Patch}_1)) \\ - \mathrm{Desc}(\mathbf{Patch}_2, \mathrm{Orient}(\mathbf{Patch}_2)) \end{array} \right\|$$

$$\mathrm{Orient}(\mathbf{patch}_1) = \mathrm{arctan2}(\mathrm{CNN}(\mathbf{patch}_1)[1], \mathrm{CNN}(\mathbf{patch}_1)[2])$$

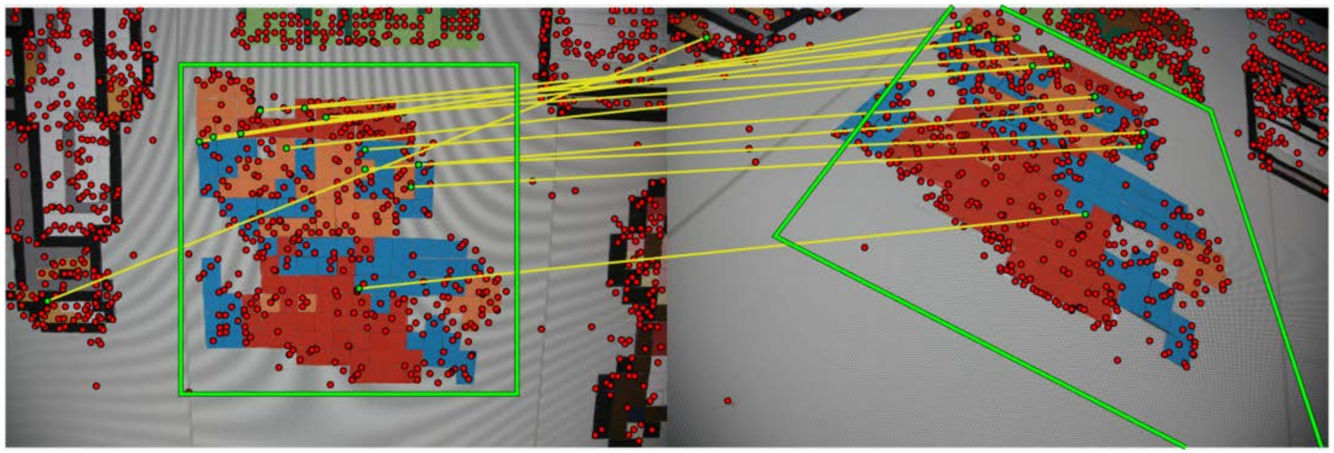
$\mathrm{Desc}(\cdot)$ is not learned. Any rotation sensitive descriptor can be used.

Matching Examples

Our Learned Orientations

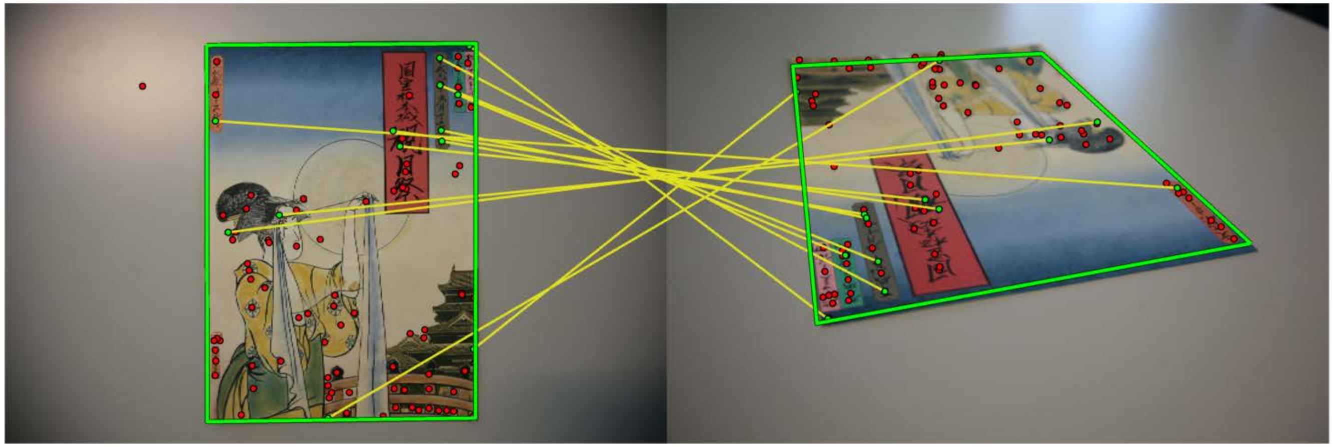


Dominant Gradient Orientations

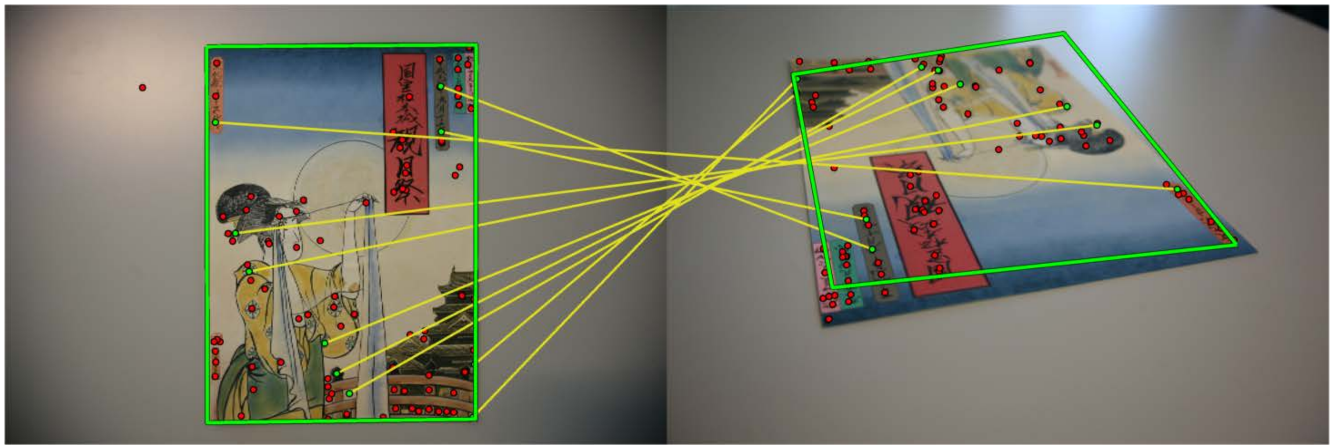


Matching Examples

Our Learned Orientations



Dominant Gradient Orientations



Quantitative Evaluation

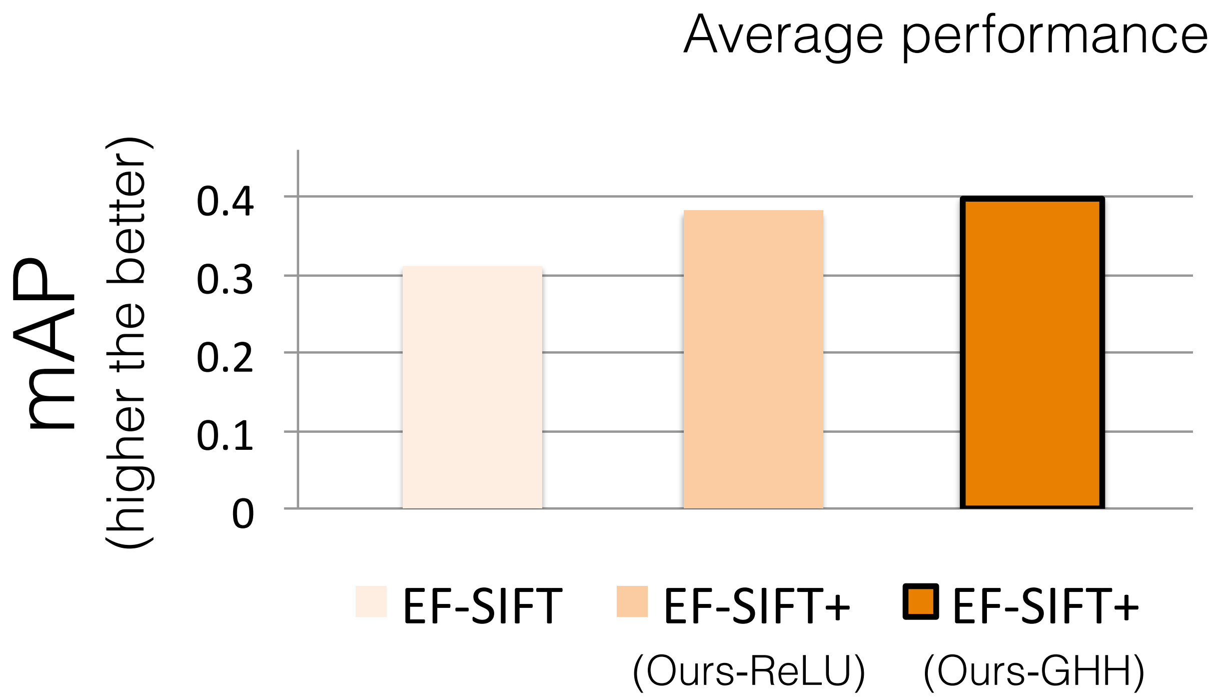
Train

Test

86 sequences, 855 images

Mikolajczyk and Schmid, 2004, Strecha et al., 2008,
Zitnick and Ramnath, 2011, Anaes et al., 2012, Verdie et al., 2015

Performance Gain with Learned Orientations



Descriptor matching performances (mAP) with nearest neighbor matching (Mikolajczyk & Schmid, IJCV'04).

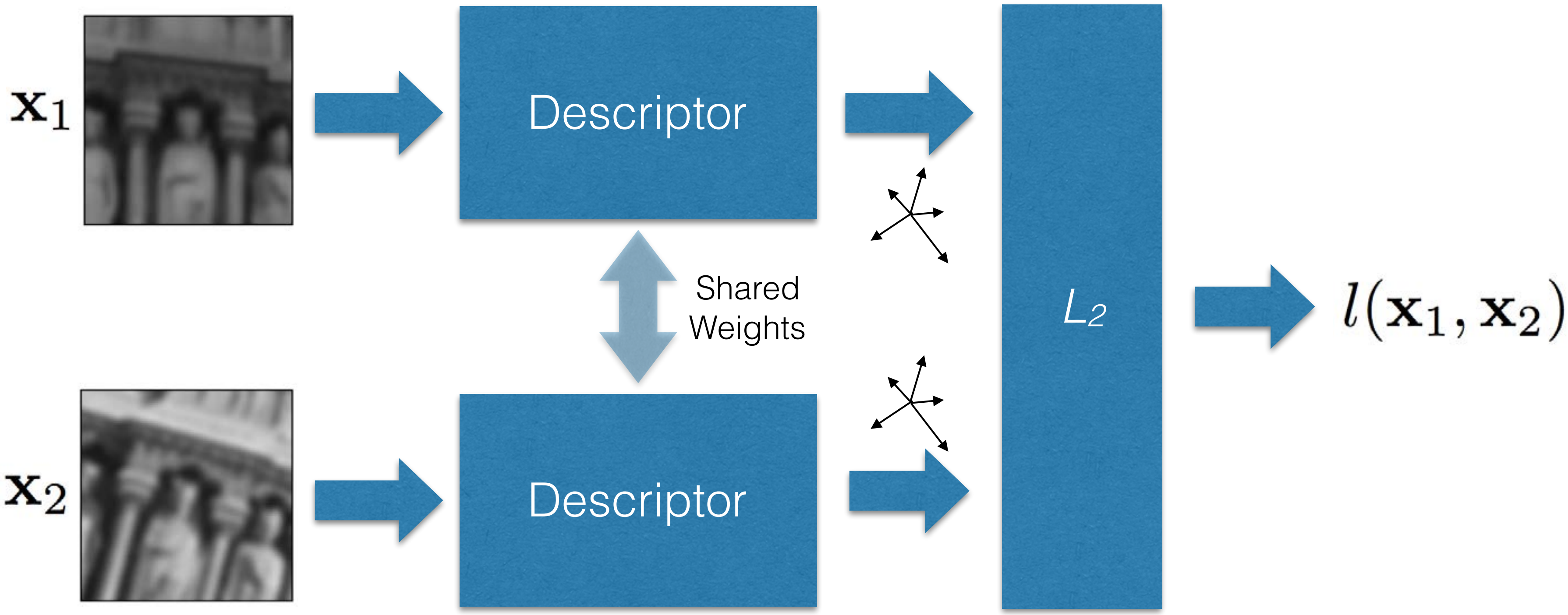
Estimating Orientation in Short

- Orientations are a key component in the local feature pipeline that has been largely **overlooked.**
- We have proposed a Deep Learning based approach to learn **good** orientations for matching purposes.
- This delivers significant performance improvements in matching performance.

3. Computing Descriptors

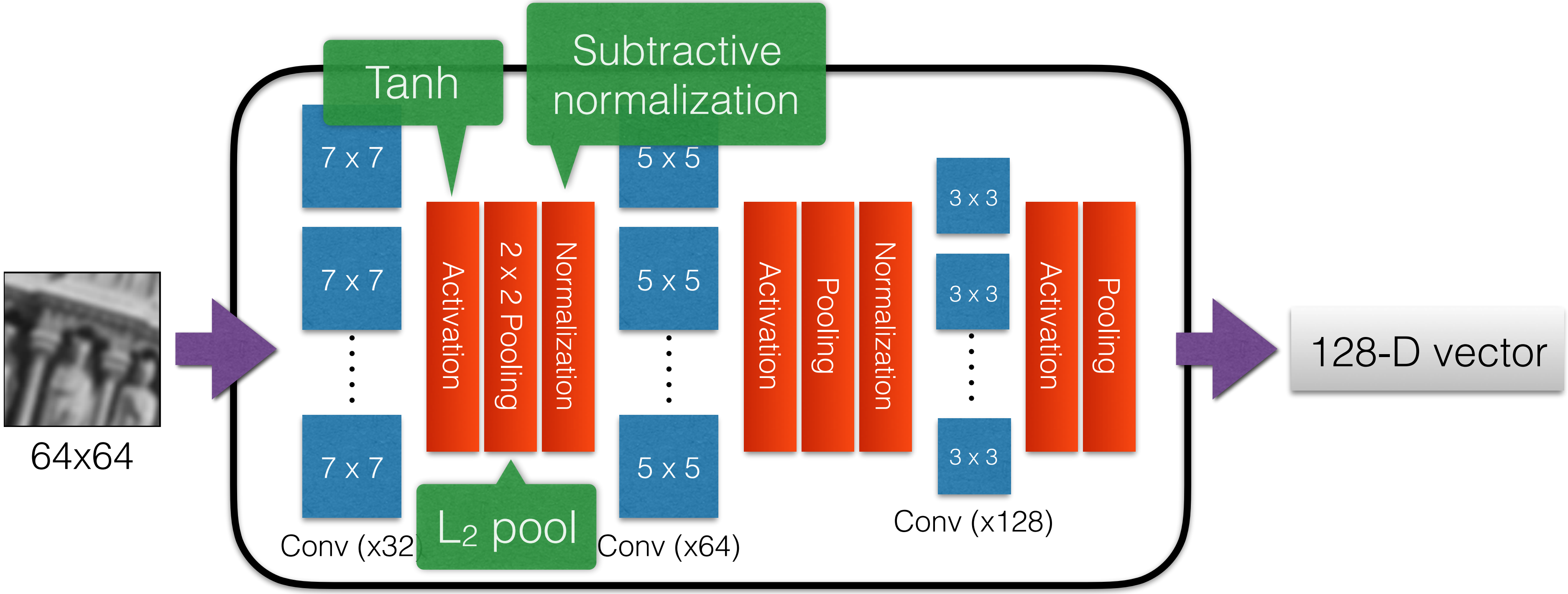
Discriminative Learning of Deep Convolutional Feature Point Descriptors (ICCV 2015)

Siamese Network



- Minimize the distance for corresponding matches.
- Maximize it for non-corresponding patches.

Our Network

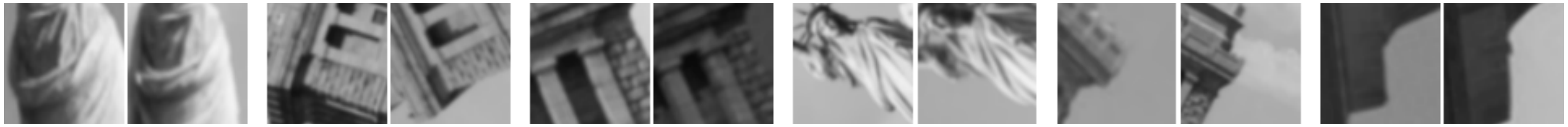


- 3 convolutional layers, no fully-connected layers.
- About 45k parameters.
- Hard mining is key to good performance.

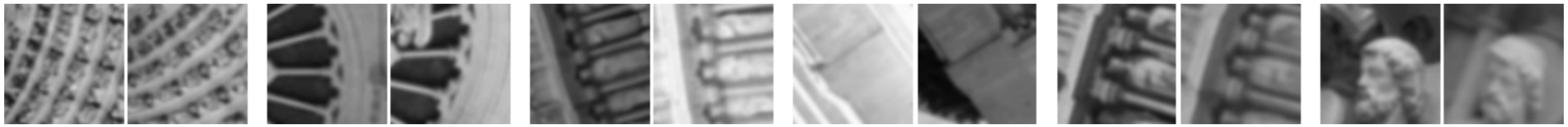
—> After training, a drop-in replacement for SIFT.

Training and Testing Data

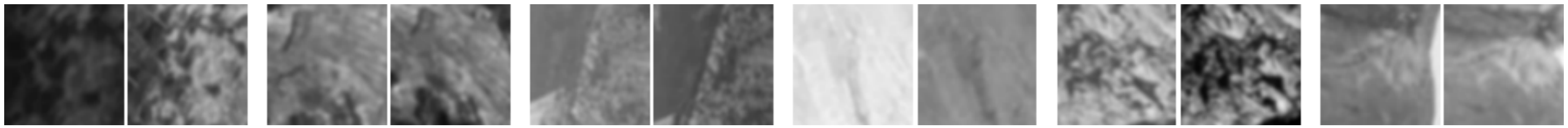
Statue of Liberty (LY)



Notre Dame (ND)



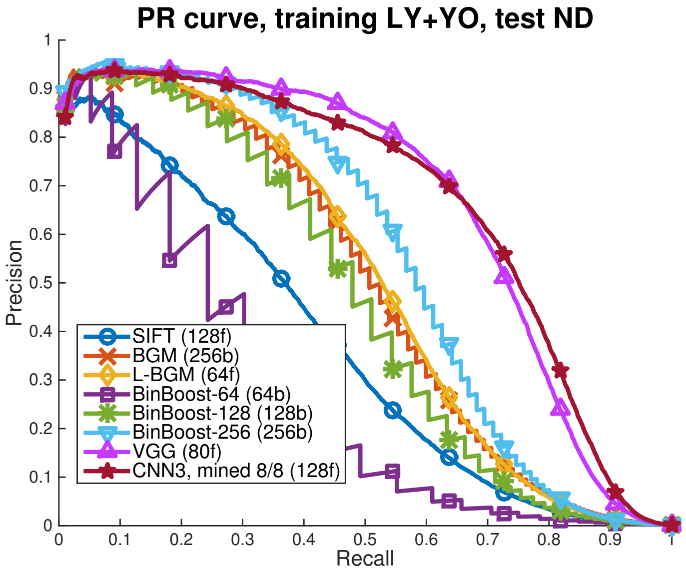
Yosemite (YO)

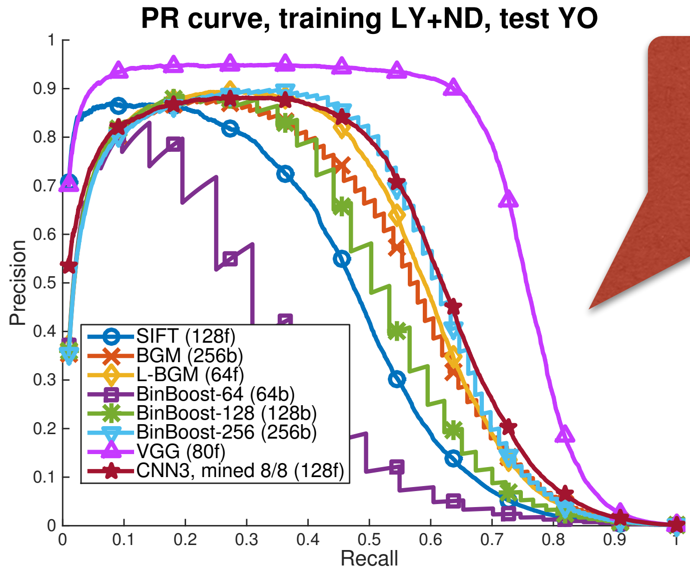


- MVS dataset (Brown et al, PAMI'11), 3 SfM sets of 64x64 grayscale patches. Each one contains ~150k 3D points and ~450k patches.

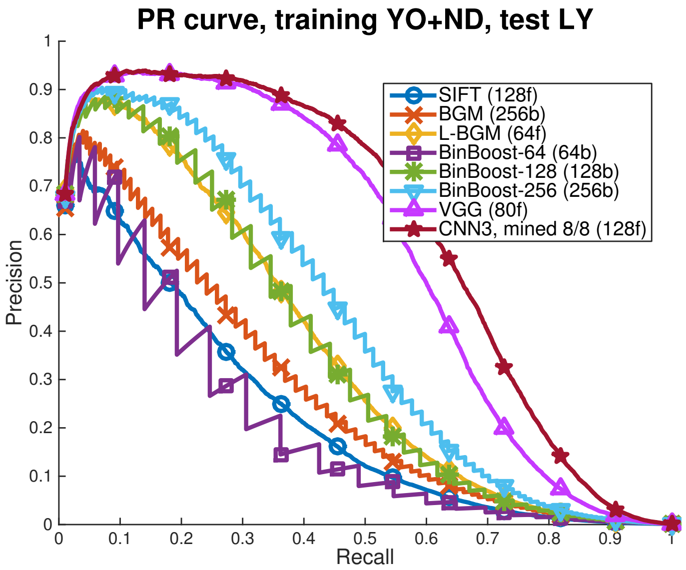
- Train on two and test on the third.

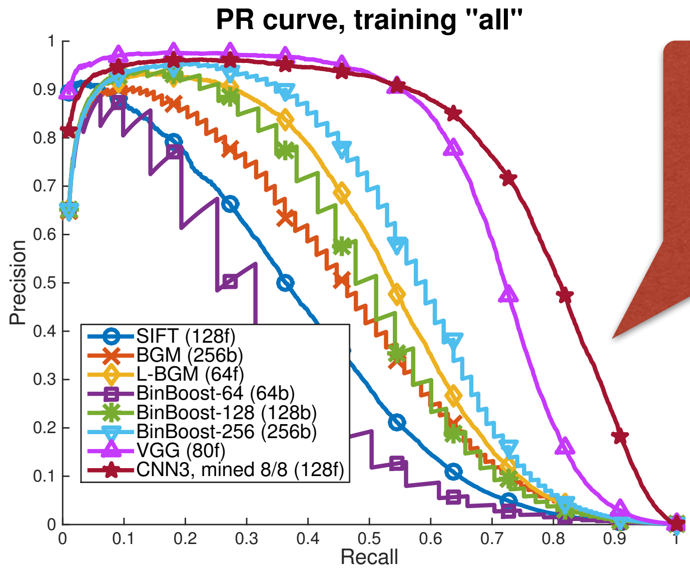
Quantitative Results





#2 on Yosemite





#1 training will all three sets

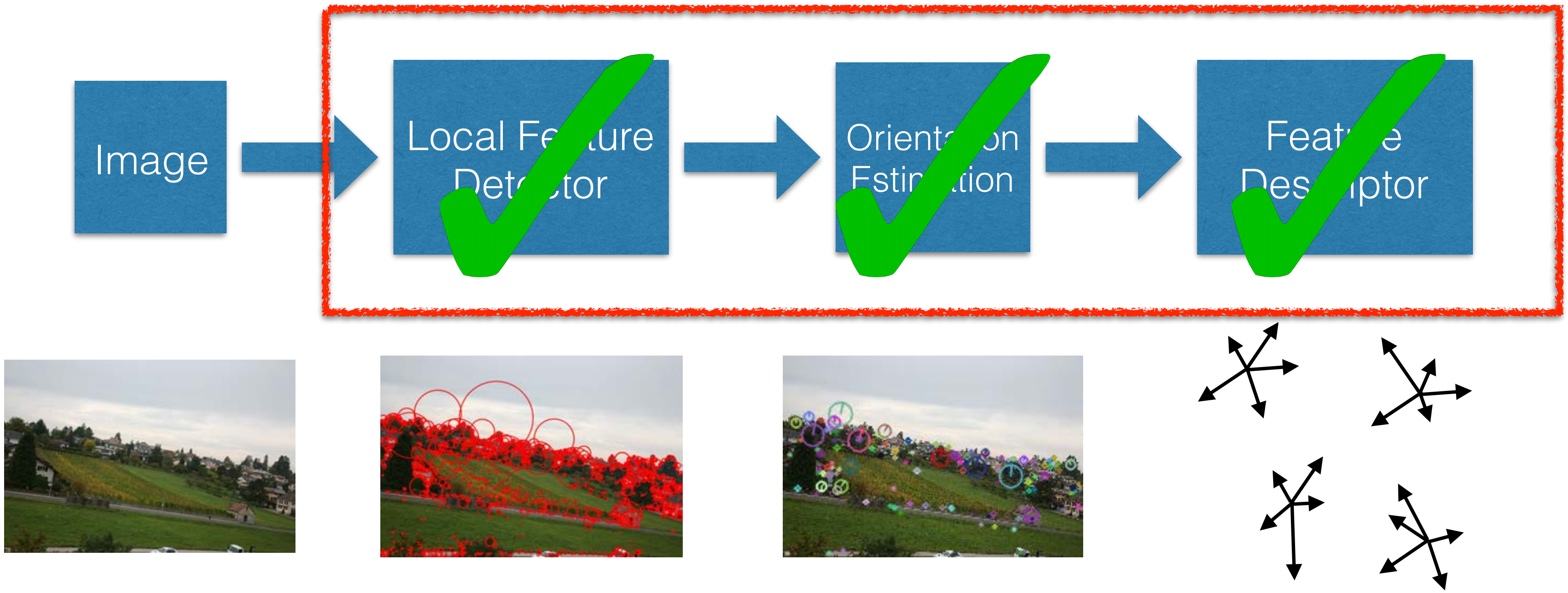
Descriptors in Short

- **Outperforms** both hand-crafted descriptors and state-of-the-art, learned descriptors.
- Good **generalization properties:** scaling, rotation, deformation, illumination changes.
- **Fast:** 0.76 ms on GPU, vs 0.14 ms for dense SIFT.
- No metric learning **→ Drop-in replacement for SIFT.**

4. Putting it all Together

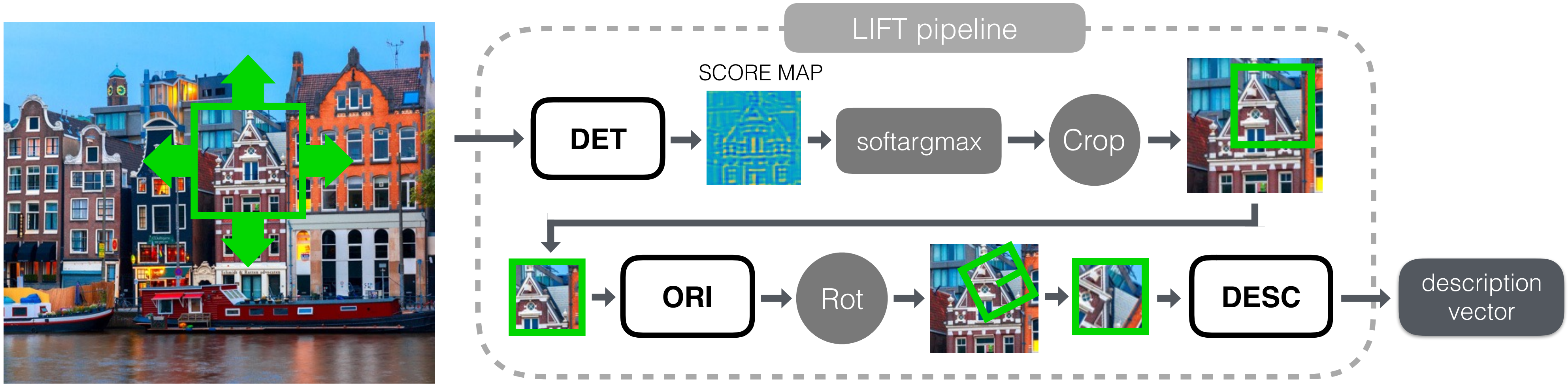
LIFT: Learned Invariant Feature Transform (ECCV 2016)

Local Feature Pipeline



All three main components are now CNNs.

Integrated Pipeline

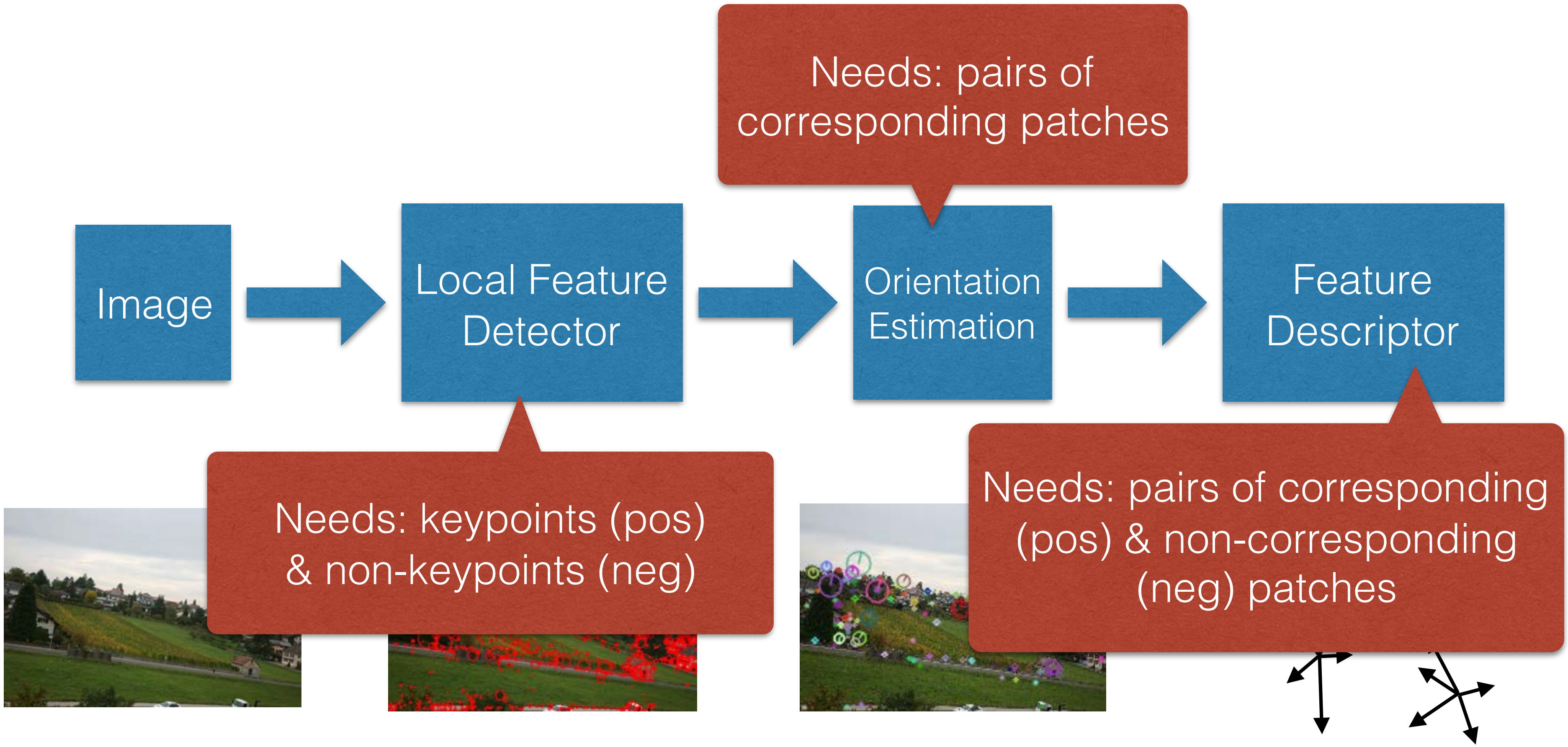


Tie everything together using **differentiable modules:**

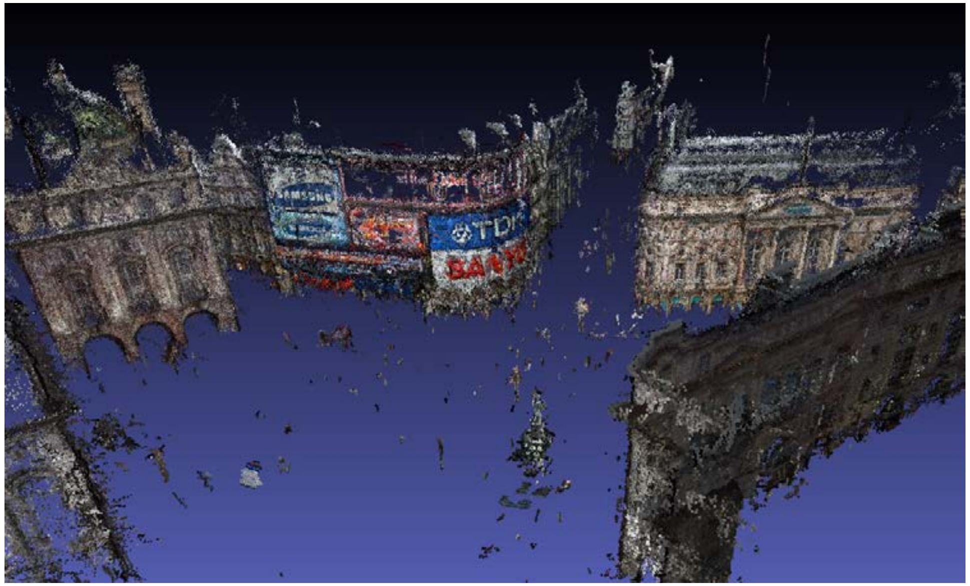
- Soft Argmax (Chapelle et al., Information Retrieval'09)
- Crop and Rotate (Spatial Transformer Networks, Jaderberg et al., NIPS'15)

—> End-to-end differentiability.

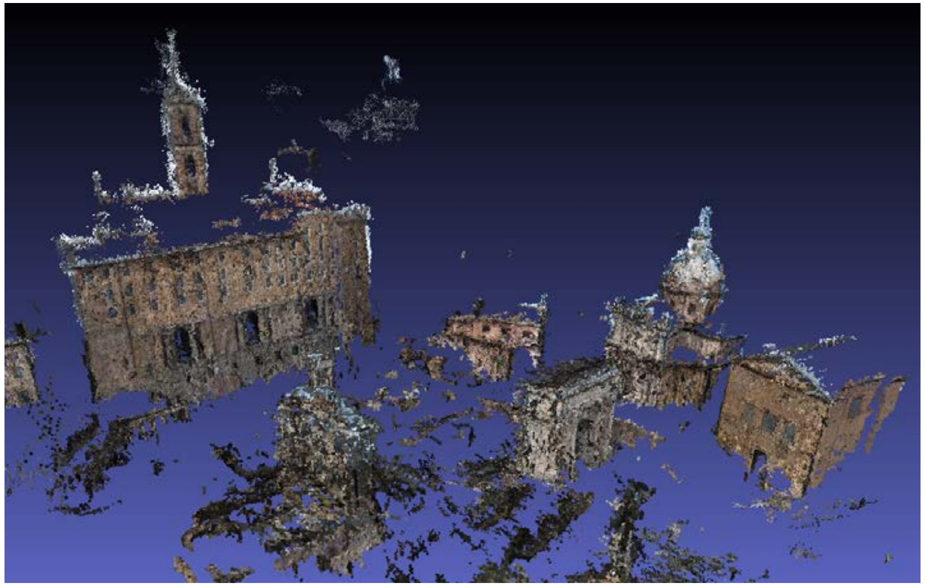
Training the pipeline



Training with SfM Keypoints



Piccadilly (pic)



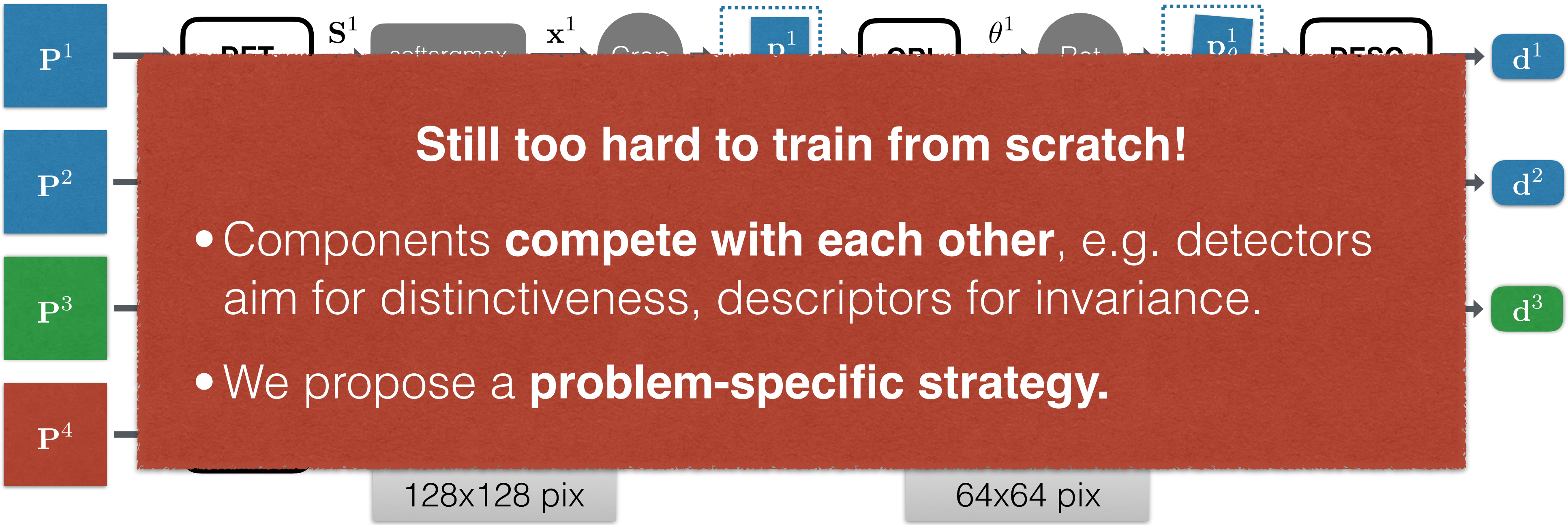
Roman Forum (rf)

- We need variability (illumination, perspective, etc). We build SfM reconstructions from **photo-tourism sets.**

- We keep only **points with SfM correspondences** as positive examples, that is, we **learn to find repeatable points.**

Quadruplet Siamese

- Use patches around SIFT locations.
- Perturb patch locations to avoid biases.



Still too hard to train from scratch!

- Components **compete with each other**, e.g. detectors aim for distinctiveness, descriptors for invariance.
- We propose a **problem-specific strategy.**

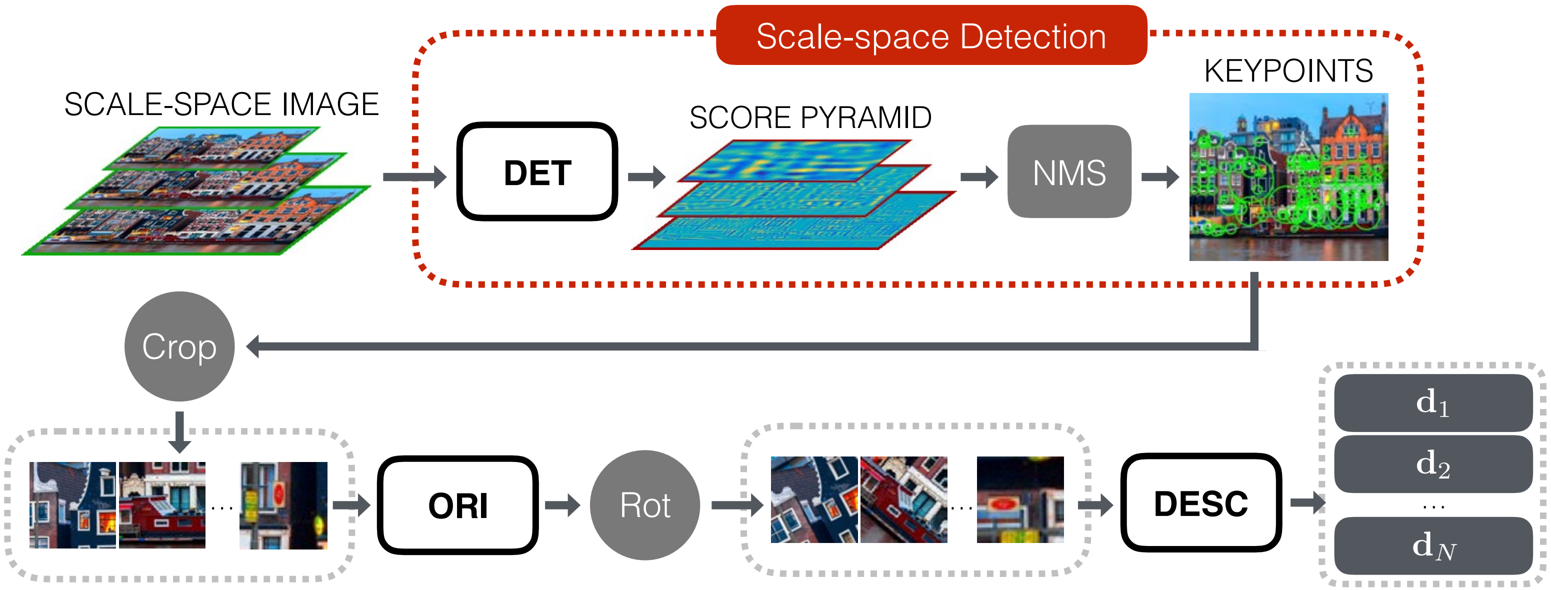
P_1, P_2: corresponding keypoints.
P_3: non-corresponding keypoint. **P_4:** non-keypoint.

Problem-Specific Training

1. Train the **Descriptor** using SfM (SIFT) patches.
2. Train the **Orientation Estimator** given the pre-trained descriptor.
3. Train the **Detector** with the pre-trained Orientation Estimator and Descriptor.

End-to-end differentiability is essential!

Runtime Pipeline

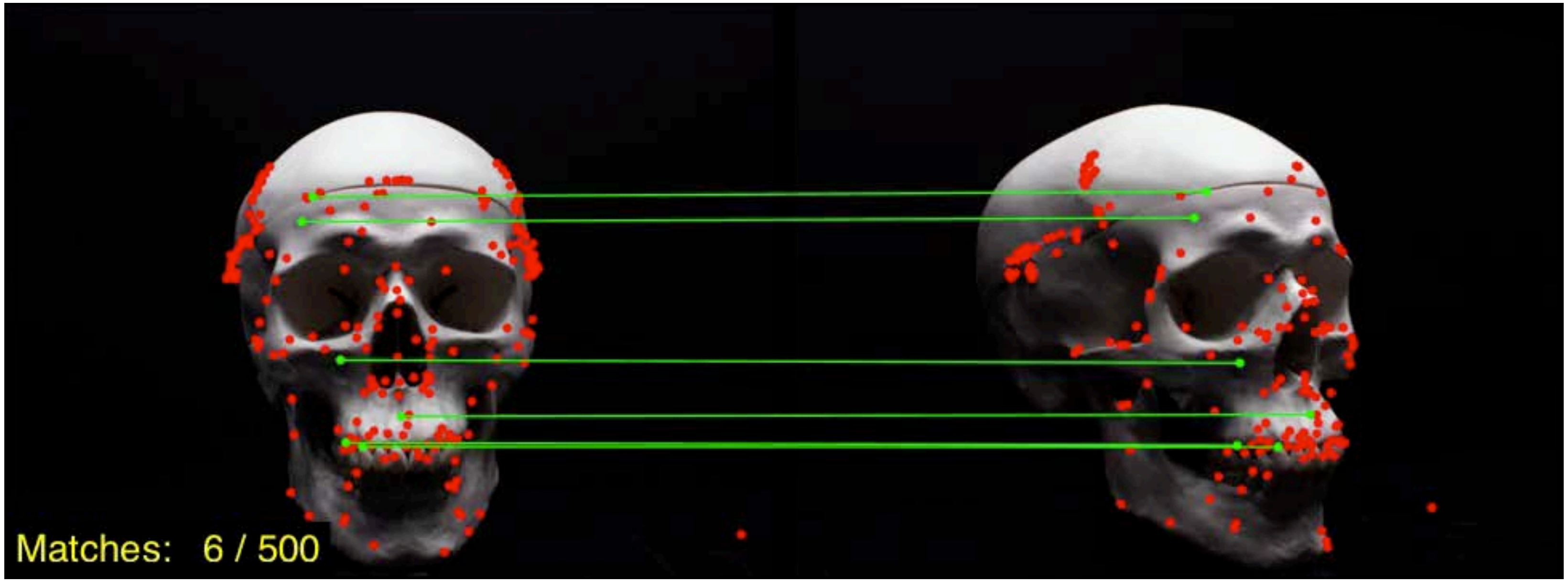


- The **Detector** runs in scale-space with traditional NMS.

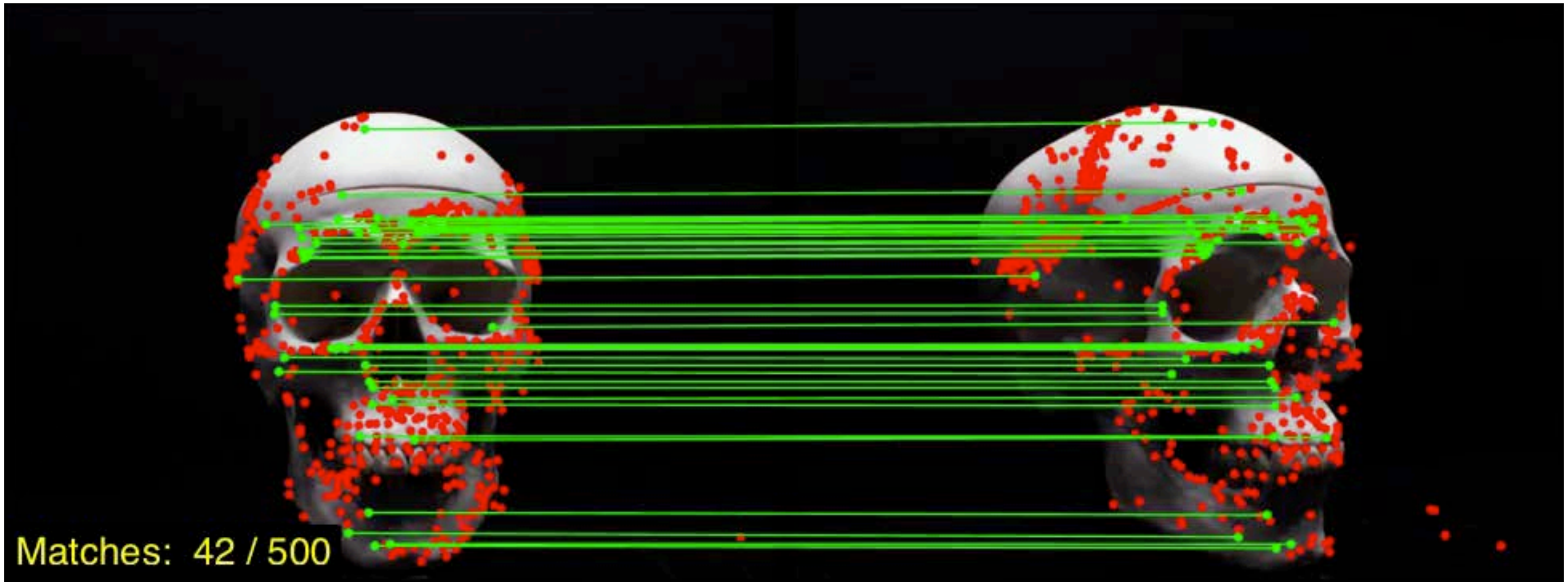
- Keypoints are passed on to the **Orientation Estimator and Descriptor** modules.

- Our TensorFlow GPU-based implementation takes ~3.0s on a 1600x1200 image, with an additional ~2.6 sec. of pure Python non-maximum suppression. On the same machine, SIFT takes ~2 sec (CPU, multi-threaded)

Matching features on **DTU** sequence **#19**.
Correct matches depicted by **green** lines.

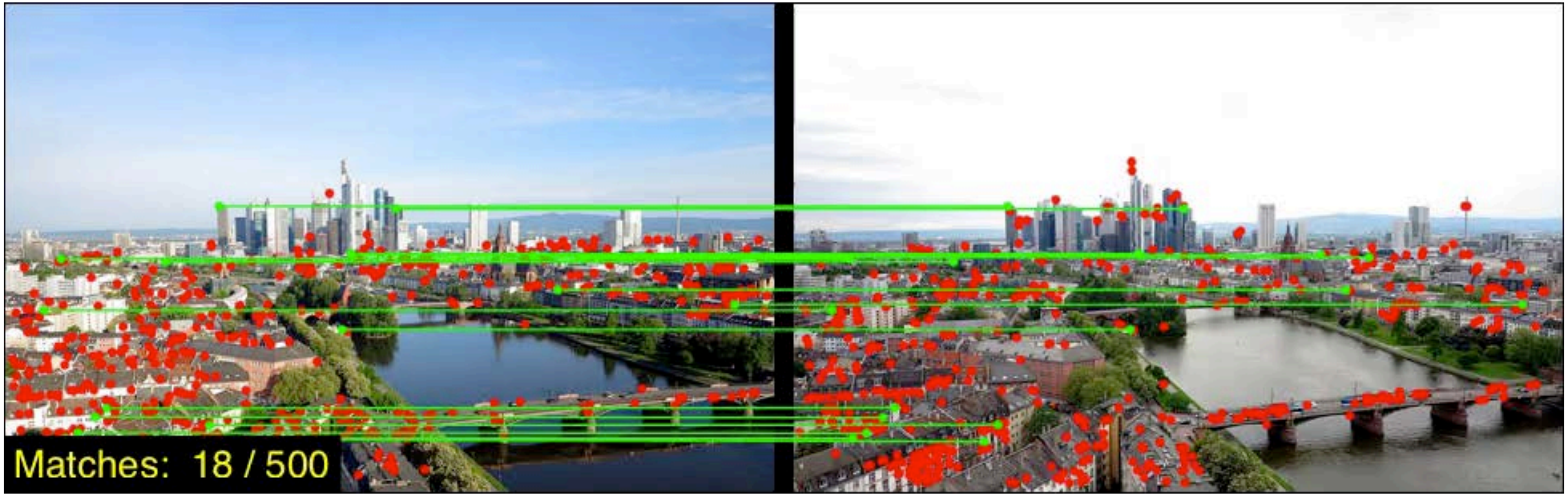


SIFT. Average: **34.1** matches

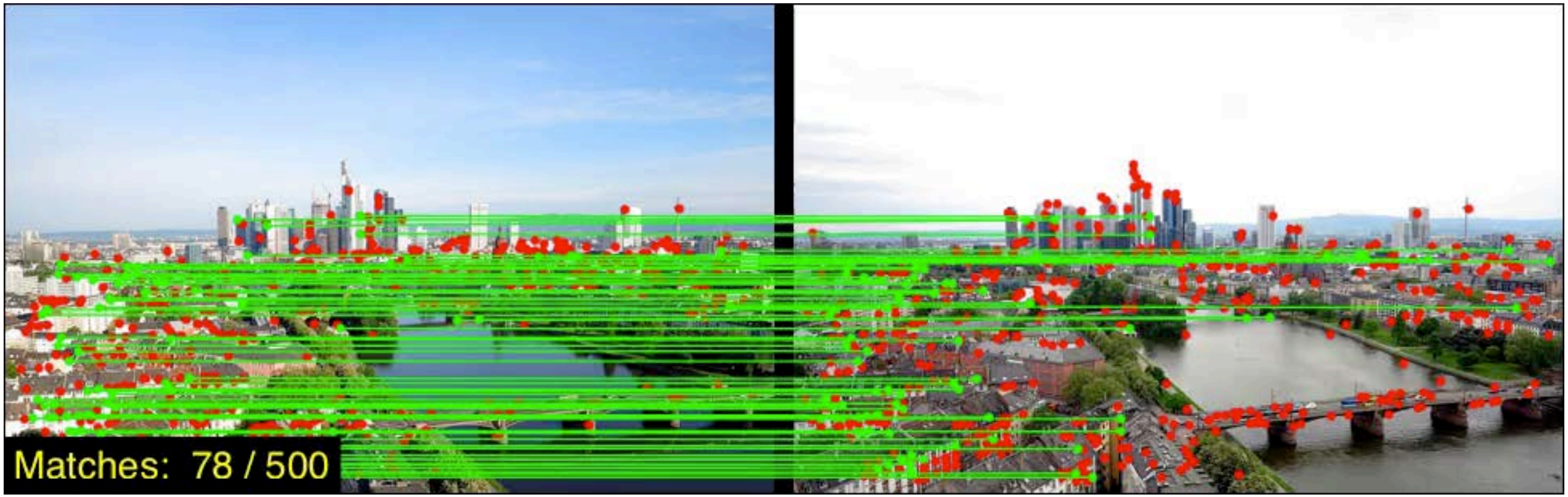


LIFT (Ours). Average: **98.5** matches

Matching features on **Webcam** sequence **Frankfurt**.
Correct matches depicted by **green** lines.



SIFT. Average: **23.1** matches



LIFT (Ours). Average: **60.6** matches

Quantitative Evaluation

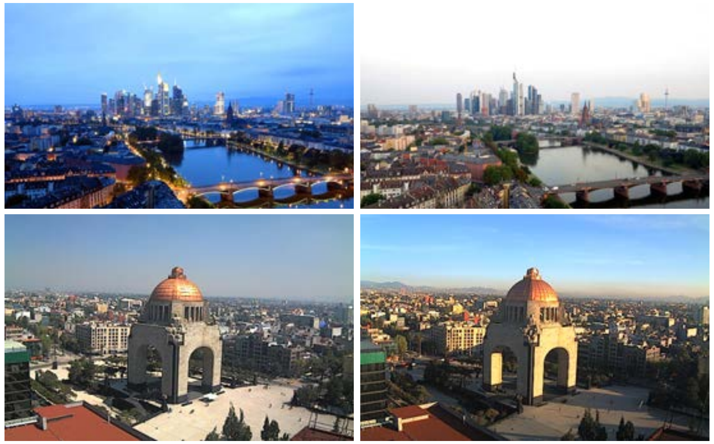
Strecha (2 seq.)

Outdoors.
Wide-baseline stereo

DTU (60 seq.)

Objects. Perspective changes.

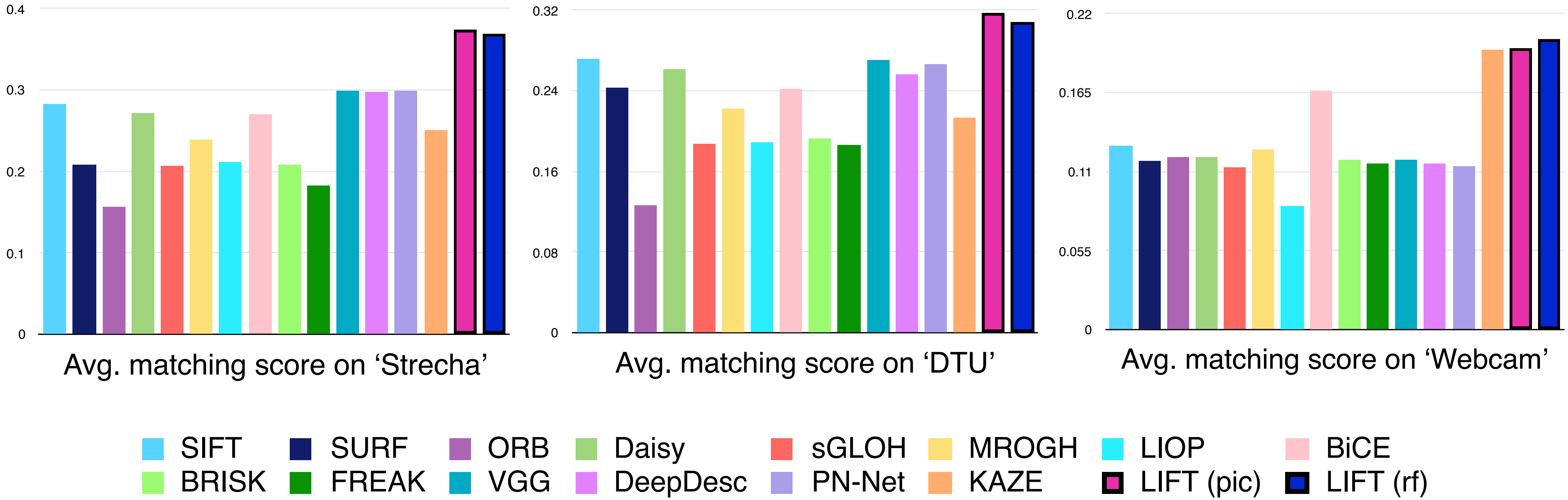
Webcam (5 seq.)



Outdoors. Fixed view, drastic illumination changes.

- **Metric:** Descriptor matching performances (mAP) with nearest neighbor matching (Mikolajczyk & Schmid, IJCV'04) as before.

Quantitative Evaluation



- **Best performance** on all datasets, with either 'pic' or 'rf'.
- **SIFT remains #3** overall (#1: ours, #2: VGG).

Yi et al. ECCV'16

SFM Benchmark

		# Images	# Registered	# Sparse Points	# Observations	Track Length	Reproj. Error	# Inlier Pairs	# Inlier Matches	# Dense Points	Pose Error	Dense Error
Fountain	*SIFT*	11	11	10,004	44K	4.49	**0.30px**	49	76K	2,970K	0.002m (0.002m)	0.77 (0.90)
	SIFT-PCA		11	**14,608**	**70K**	**4.80**	0.39px	**55**	**124K**	**3,021K**	0.002m (0.002m)	0.77 (0.90)
	DSP-SIFT		11	**14,785**	**71K**	**4.80**	0.41px	54	**129K**	2,999K	0.002m (0.002m)	0.77 (0.90)
	ConvOpt		11	**14,179**	**67K**	**4.75**	0.37px	**55**	**114K**	**2,999K**	0.002m (0.002m)	0.77 (0.90)
	DeepDesc		11	13,519	61K	4.55	**0.35px**	**55**	93K	2,972K	0.002m (0.002m)	0.77 (0.90)
	TFeat		11	13,696	64K	4.68	**0.35px**	54	103K	2,969K	0.002m (0.002m)	0.77 (0.90)
	LIFT		11	10,172	46K	4.55	0.59px	**55**	83K	**3,019K**	0.002m (0.002m)	0.77 (0.90)
Herzjesu	*SIFT*	8	8	4,916	19K	4.00	**0.32px**	27	28K	2,373K	0.004m (0.004m)	0.57 (0.73)
	SIFT-PCA		8	**7,433**	**31K**	**4.19**	0.42px	**28**	**47K**	2,372K	0.004m (0.004m)	0.57 (0.73)
	DSP-SIFT		8	**7,760**	**32K**	**4.19**	0.45px	**28**	**50K**	**2,376K**	0.004m (0.004m)	0.57 (0.73)
	ConvOpt		8	6,939	28K	**4.13**	0.40px	**28**	42K	2,375K	0.004m (0.004m)	0.57 (0.73)
	DeepDesc		8	6,418	25K	3.92	**0.38px**	**28**	34K	**2,380K**	0.004m (0.004m)	0.57 (0.73)
	TFeat		8	6,606	27K	4.09	**0.38px**	**28**	38K	**2,377K**	0.004m (0.004m)	0.57 (0.73)
	LIFT		8	**7,834**	**30K**	3.95	0.63px	**28**	**46K**	2,375K	0.004m (0.004m)	0.57 (0.73)
South Building	*SIFT*	128	128	62,780	353K	5.64	**0.42px**	1K	1,003K	1,972K	–	–
	SIFT-PCA		128	**107,674**	**650K**	**6.04**	0.54px	**3K**	**2,019K**	1,993K	–	–
	DSP-SIFT		128	**110,394**	**664K**	**6.02**	0.57px	3K	**2,079K**	**1,994K**	–	–
	ConvOpt		128	**103,602**	**617K**	5.96	0.51px	**4K**	**1,856K**	**2,007K**	–	–
	DeepDesc		128	101,154	558K	5.53	**0.48px**	**6K**	1,463K	**2,002K**	–	–
	TFeat		128	94,589	566K	**5.99**	**0.49px**	3K	1,567K	1,960K	–	–
	LIFT		128	74,607	399K	5.35	0.78px	3K	1,168K	1,975K	–	–
Madrid Metropolis	*SIFT*	1,344	440	62,729	416K	**6.64**	**0.53px**	14K	1,740K	435K	–	–
	SIFT-PCA		**465**	**119,244**	**702K**	5.89	0.57px	**27K**	**3,597K**	**537K**	–	–
	DSP-SIFT		**476**	**107,028**	**681K**	**6.36**	0.64px	**21K**	**3,155K**	**570K**	–	–
	ConvOpt		**455**	**115,134**	**634K**	5.51	0.57px	**29K**	**3,148K**	**561K**	–	–
	DeepDesc		377	68,110	348K	5.11	**0.53px**	19K	1,570K	516K	–	–
	TFeat		439	90,274	512K	5.68	**0.54px**	18K	2,135K	522K	–	–
	LIFT		430	52,755	337K	**6.40**	0.76px	13K	1,498K	450K	–	–
Gendarmenmarkt	*SIFT*	1,463	950	169,900	1,010K	**5.95**	**0.64px**	28K	3,292K	1,104K	–	–
	SIFT-PCA		**953**	272,118	**1,477K**	**5.43**	0.69px	**43K**	**5,137K**	1,240K	–	–
	DSP-SIFT		**975**	**321,846**	**1,732K**	**5.38**	0.74px	**56K**	**7,648K**	**1,505K**	–	–
	ConvOpt		945	**341,591**	**1,601K**	4.69	0.70px	**56K**	**6,525K**	**1,342K**	–	–
	DeepDesc		809	244,925	949K	3.88	**0.68px**	31K	2,849K	921K	–	–
	TFeat		**953**	**297,266**	1,445K	4.86	**0.66px**	39K	4,685K	1,181K	–	–
	LIFT		942	180,746	964K	5.34	0.83px	27K	2,495K	**1,386K**	–	–
Tower of London	*SIFT*	1,576	702	142,746	963K	6.75	**0.53px**	18K	3,211K	1,126K	–	–
	SIFT-PCA		692	137,800	1,090K	**7.91**	0.60px	12K	2,455K	1,124K	–	–
	DSP-SIFT		**755**	**236,598**	**1,761K**	**7.44**	0.64px	**33K**	**8,056K**	**1,143K**	–	–
	ConvOpt		**719**	**274,987**	**1,732K**	6.30	0.62px	**39K**	**7,542K**	**1,129K**	–	–
	DeepDesc		551	196,990	964K	4.90	**0.55px**	25K	2,745K	653K	–	–
	TFeat		714	**206,142**	**1,424K**	6.91	**0.57px**	**28K**	**5,333K**	**1,182K**	–	–
	LIFT		**715**	147,851	1,045K	**7.07**	0.72px	23K	4,079K	729K	–	–
Alamo	*SIFT*	2,915	743	120,713	1,384K	11.47	0.54px	**23K**	**7,671K**	**611K**	–	–
	SIFT-PCA		**746**	108,553	1,377K	**12.69**	0.55px	12K	4,669K	564K	–	–
	DSP-SIFT		**754**	**144,341**	**1,815K**	**12.58**	0.66px	**16K**	**10,115K**	**629K**	–	–
	ConvOpt		703	102,044	1,001K	9.81	**0.48px**	3K	850K	452K	–	–
	DeepDesc		665	**152,537**	1,207K	7.92	**0.48px**	16K	4,196K	607K	–	–
	TFeat		683	**127,642**	**1,443K**	11.31	**0.52px**	16K	6,356K	**648K**	–	–
	LIFT		**768**	112,984	**1,477K**	**13.08**	0.73px	**23K**	**9,117K**	607K	–	–
Roman Forum	*SIFT*	2,364	1,407	242,192	1,805K	**7.45**	0.61px	**25K**	**6,063K**	**3,097K**	–	–
	SIFT-PCA		**1,463**	**244,556**	**1,834K**	**7.50**	0.61px	16K	4,322K	2,799K	–	–
	DSP-SIFT		**1,583**	**372,573**	**2,879K**	**7.73**	0.71px	**26K**	**9,685K**	**3,748K**	–	–
	ConvOpt		1,376	195,305	1,173K	6.01	**0.55px**	11K	2,111K	3,043K	–	–
	DeepDesc		1,173	174,532	1,275K	7.31	**0.60px**	9K	1,834K	2,434K	–	–
	TFeat		**1,450**	**271,902**	**1,963K**	7.22	**0.61px**	**19K**	**5,584K**	**3,477K**	–	–
	LIFT		1,434	220,026	1,608K	7.31	0.75px	17K	4,732K	2,898K	–	–
Cornell	*SIFT*	6,514	**4,999**	1,010,544	6,317K	6.25	**0.53px**	71K	25,603K	**12,970K**	**1.537m (0.793m)**	–
	SIFT-PCA		3,049	640,553	4,335K	**6.77**	**0.54px**	26K	13,793K	6,135K	11.498m (1.088m)	–
	DSP-SIFT		**4,946**	1,177,916	**7,233K**	6.14	0.67px	73K	26,150K	**11,066K**	**2.943m (1.001m)**	–
	ConvOpt		1,986	632,613	4,747K	**7.50**	0.57px	42K	18,615K	5,321K	5.824m (0.904m)	–
	DeepDesc		3,489	**1,225,780**	6,977K	5.69	**0.55px**	**73K**	**28,845K**	10,159K	3.832m (0.695m)	–
	TFeat		**5,428**	**1,499,117**	**9,830K**	**6.56**	0.59px	**89K**	**40,640K**	**15,605K**	**2.126m (0.593m)**	–
	LIFT		3,798	**1,455,732**	**7,377K**	5.07	0.71px	**81K**	**39,812K**	10,512K	3.113m (0.712m)	–

Table 3. Results for our reconstruction benchmark. Pose error as mean (median) over all images. Dense error for 2cm (10cm) threshold [19]. **First**, **second**, **third** best results highlighted in bold. Number of images, sparse points, and dense points visualized in Figs. 1, 2, and 3.

- Reprojection errors are all under one 1pixel.
- LIFT's error is a little higher, probably because we trade recall for accuracy.

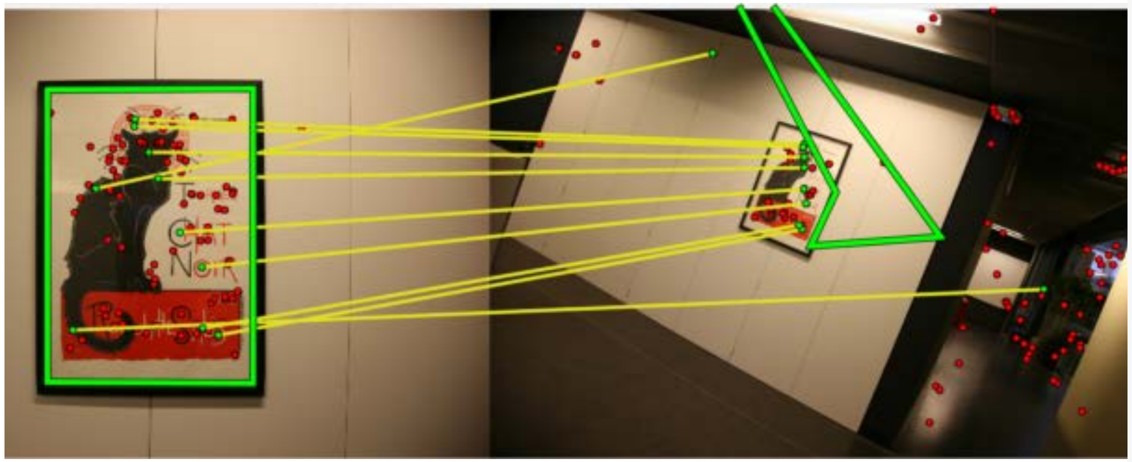
- Pose accuracy is relatively similar for all.

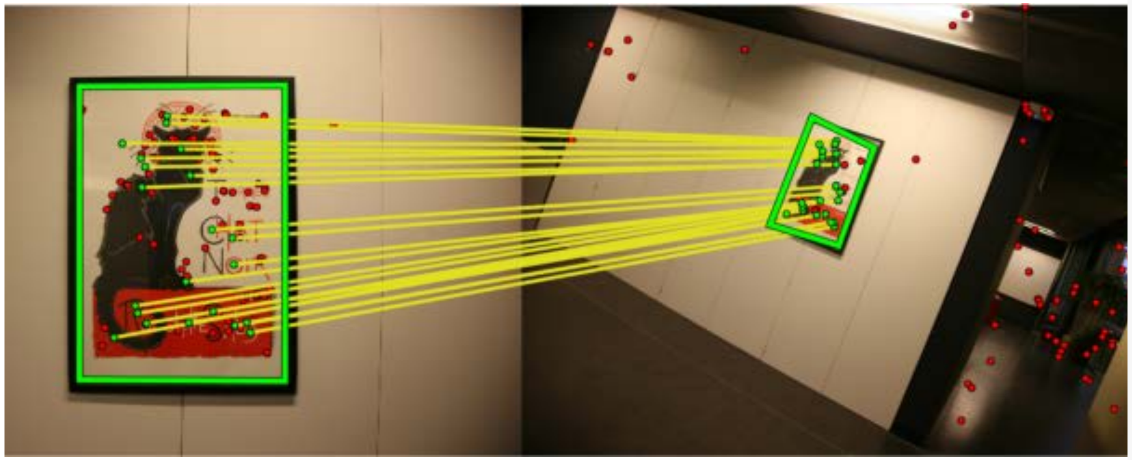
- Sometimes the two do not correlate exactly.

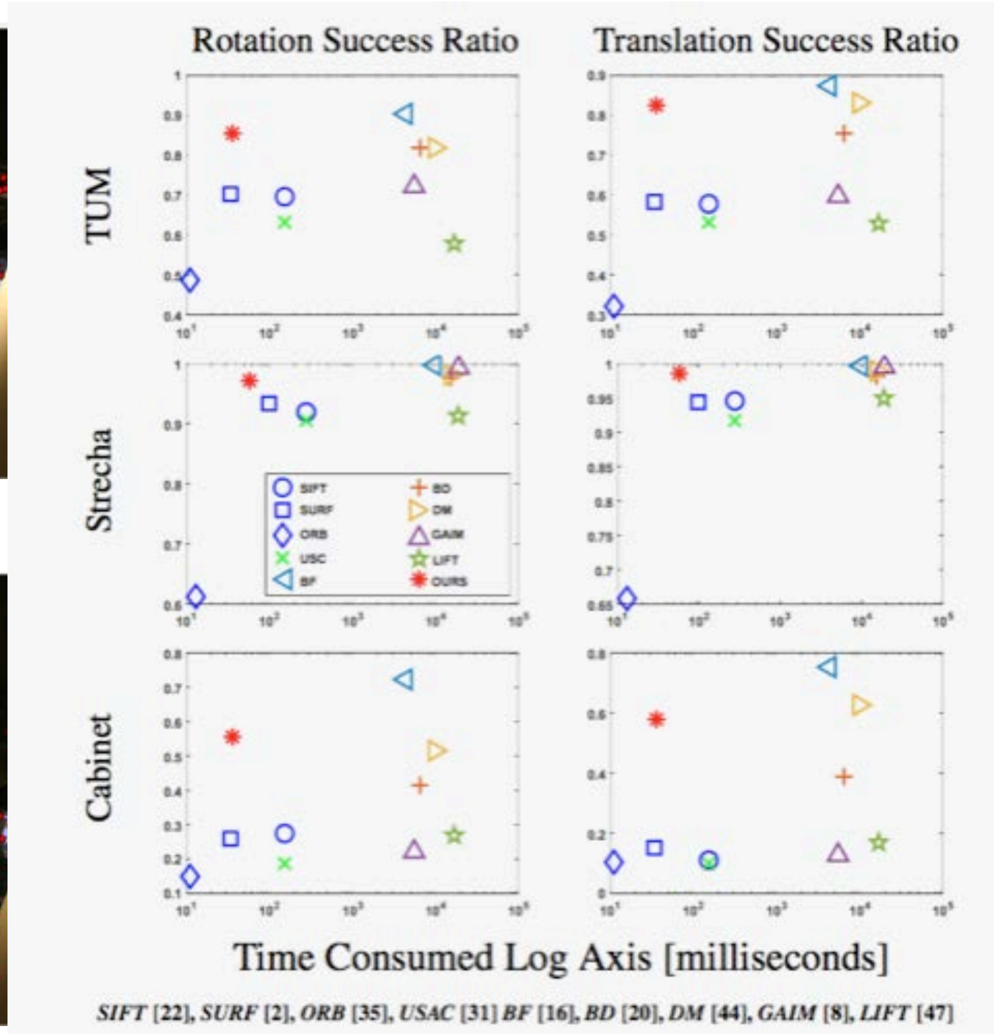
—> For the purpose of SFM, the chosen approach to establishing correspondences and rejecting outliers may be more important than the specific features being used.

Schoenberger et al. CVPR'17

Keypoints are only a means to an end!





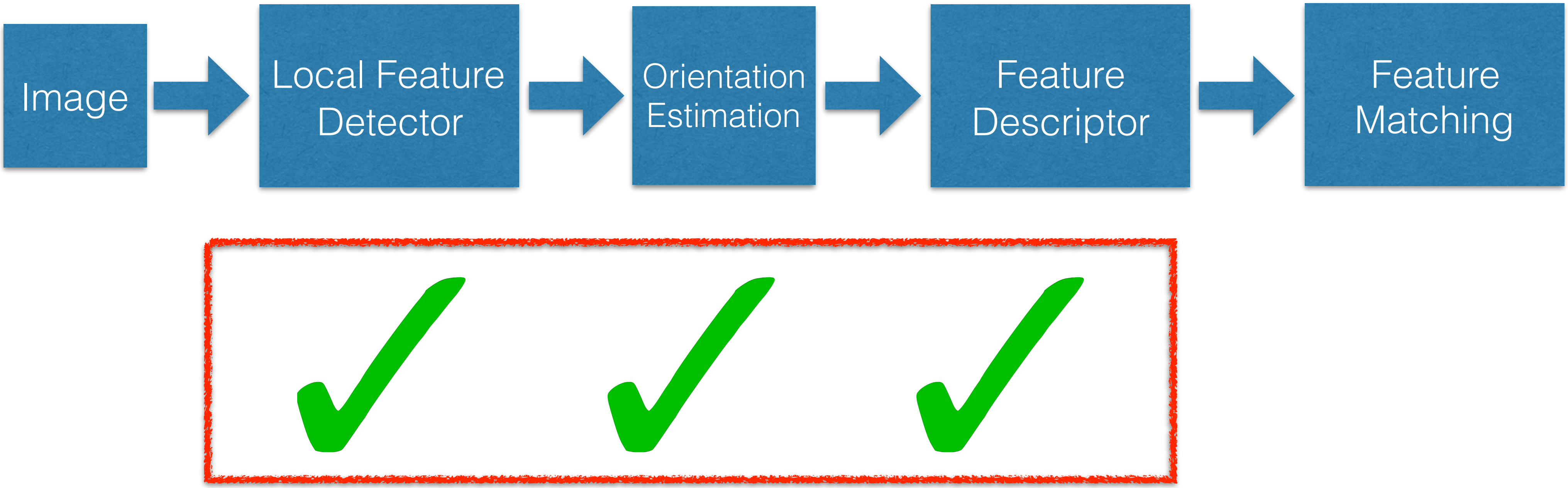


- LIFT maximises the number of matches.
- Not all of them are useful.

—> Need a good way to learn which ones are.

Bian et al, CVPR 17

Local Feature Pipeline Revisited

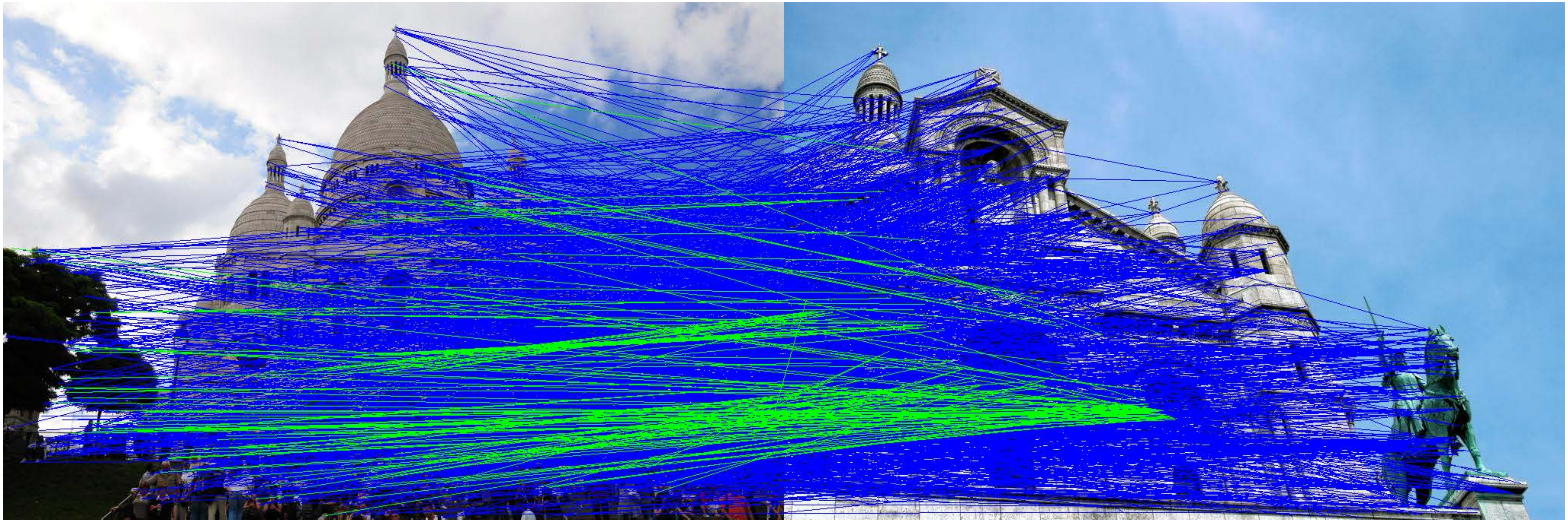


- Three of the four main components are now CNNs.
- They have now been integrated into a single pipeline.

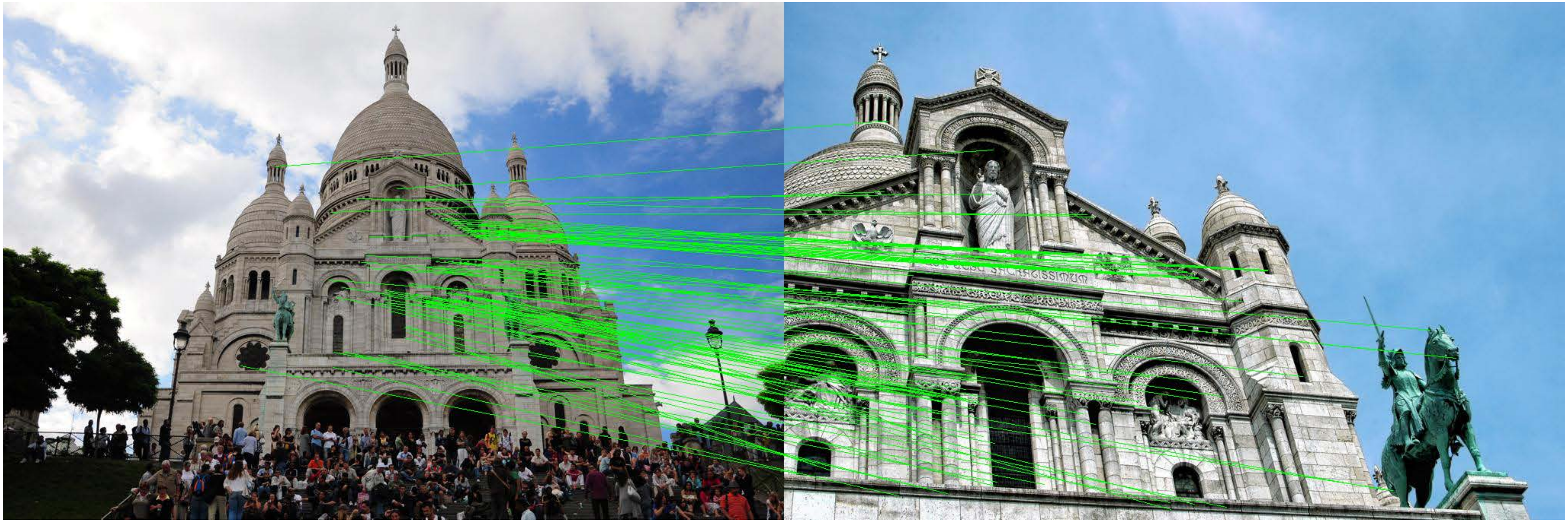
—> Must now work on the fourth!

4. Correspondences

RANSAC + 5 point method is not enough



Deep Learning to the Rescue

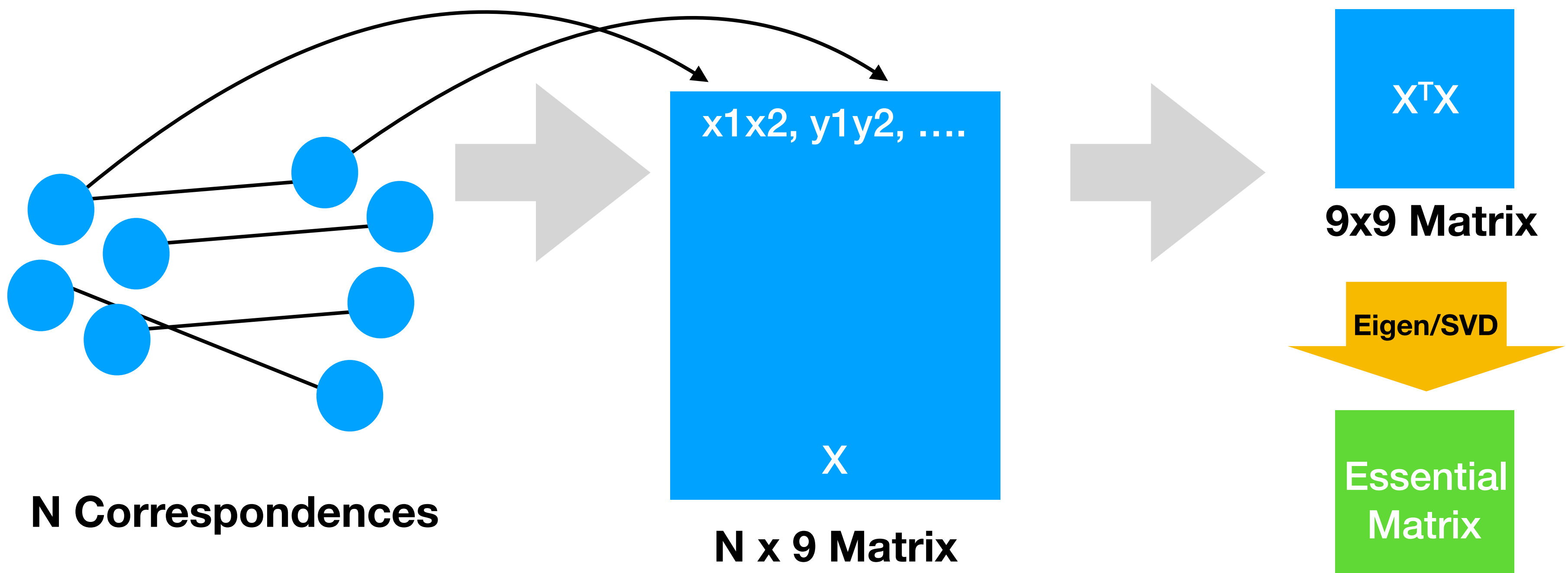


Learn to reject outliers and estimate the Essential matrix simultaneously.

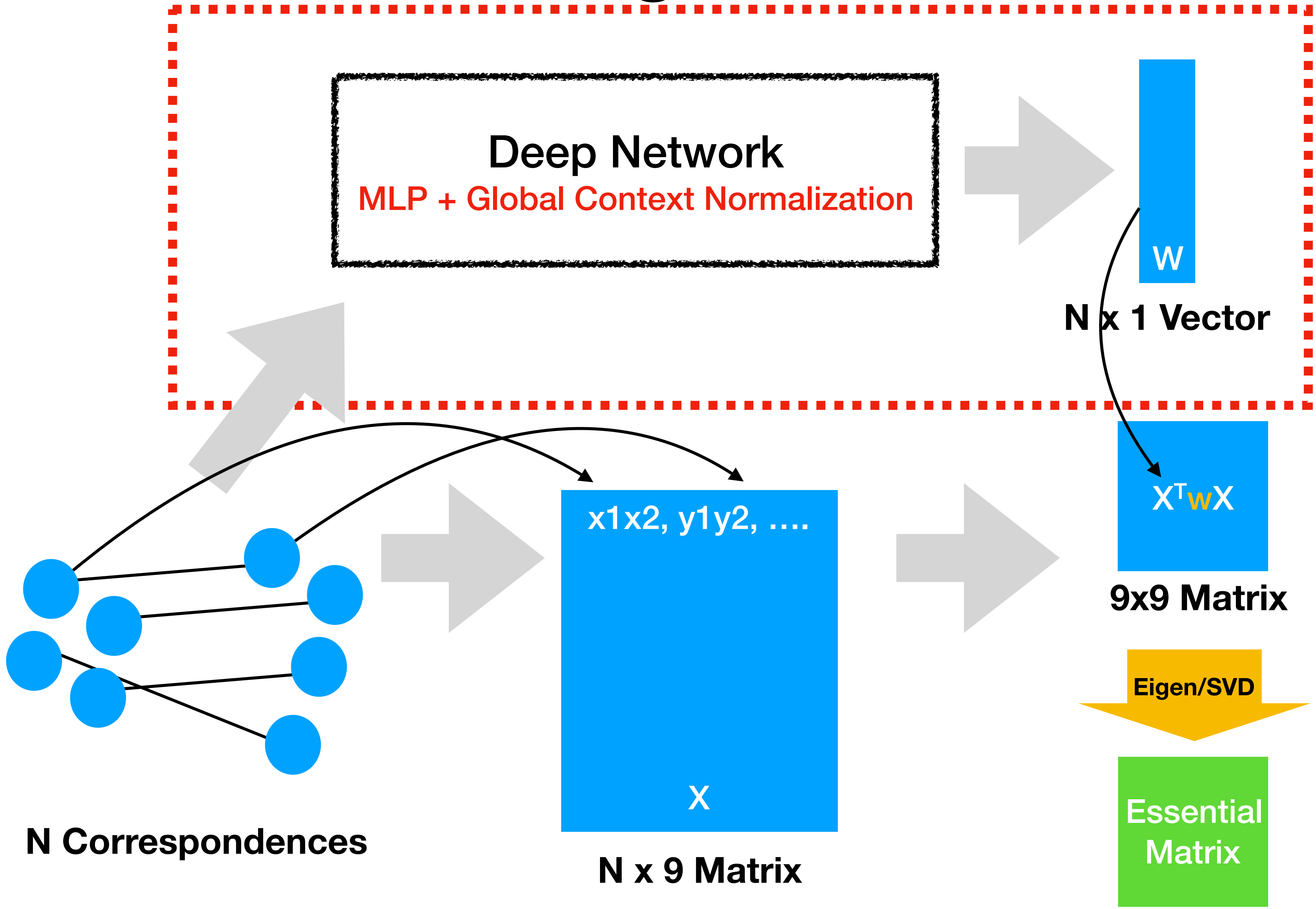
—> Incorporate global context into the matching process.

Hartley and Zisserman, 2003

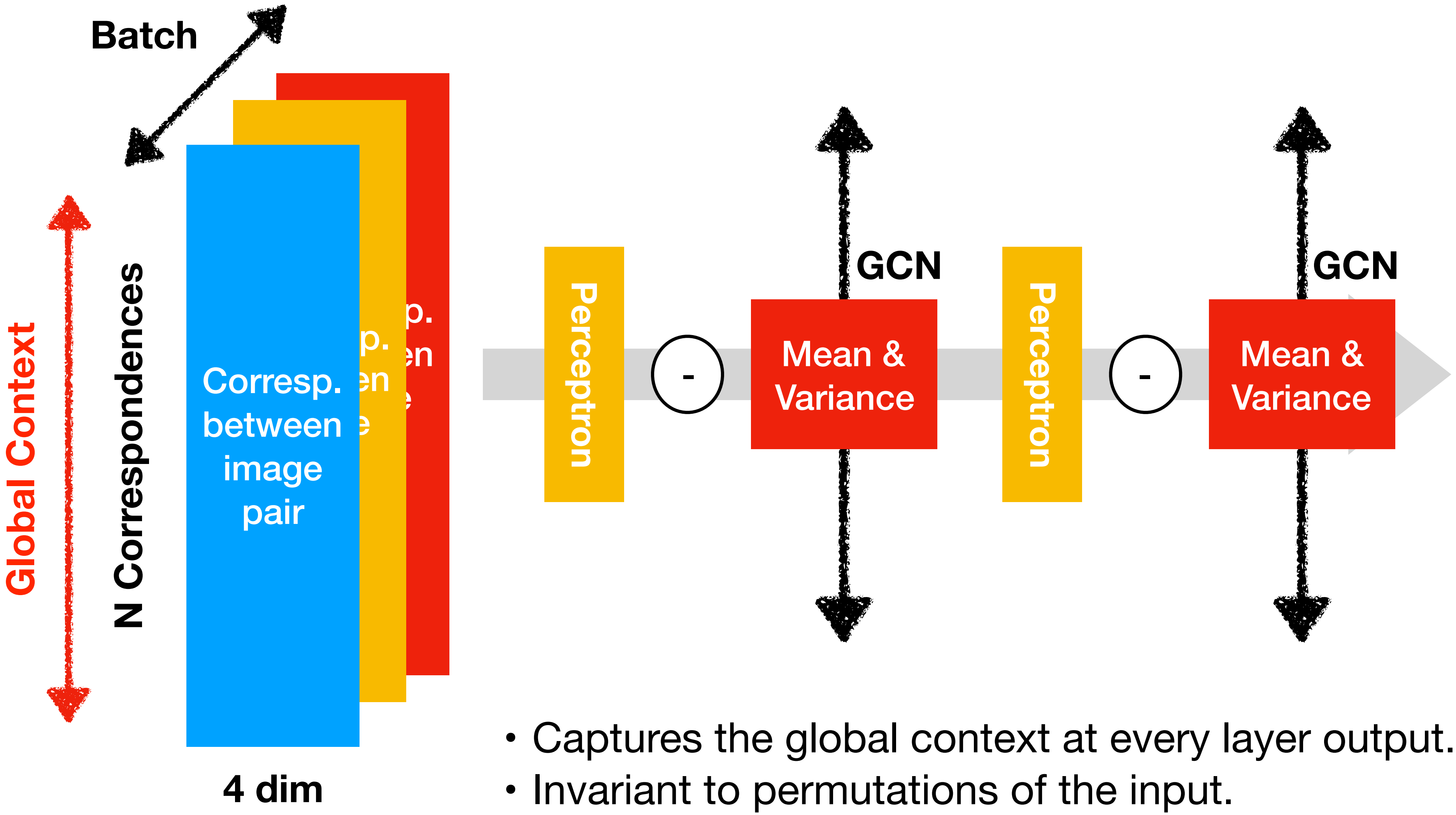
Revisiting the 8-point Algorithm



Simultaneous Classification and Regression

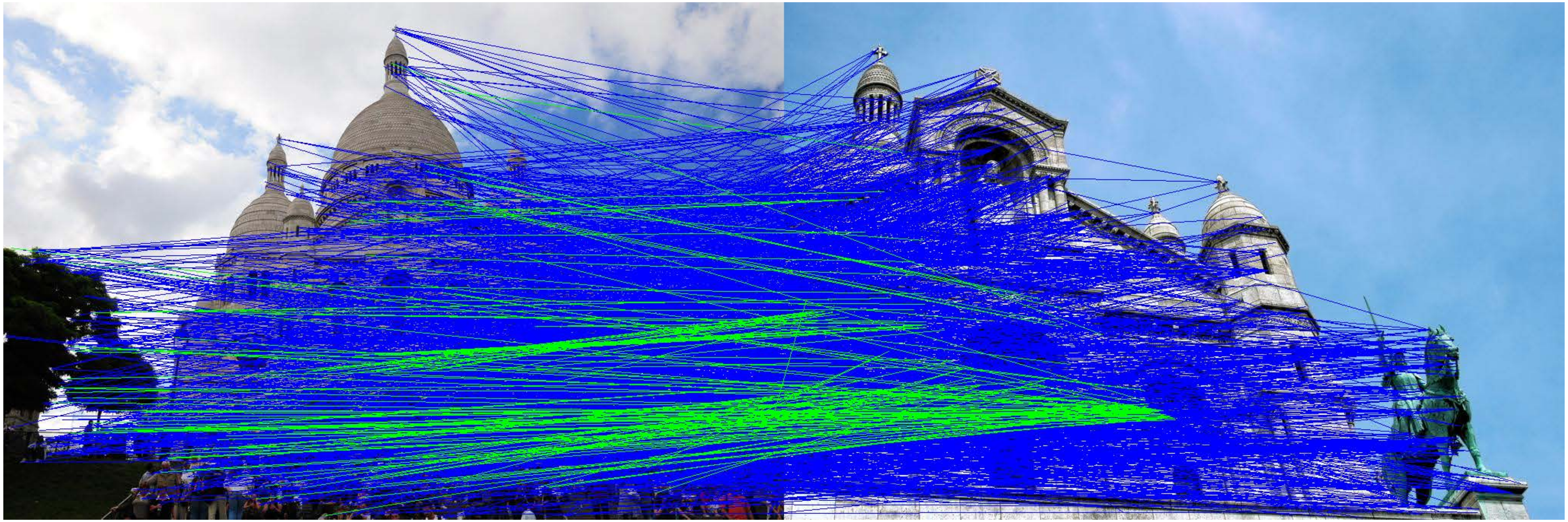


Multi-Layer Perceptron with Global Context Normalization (GCN)



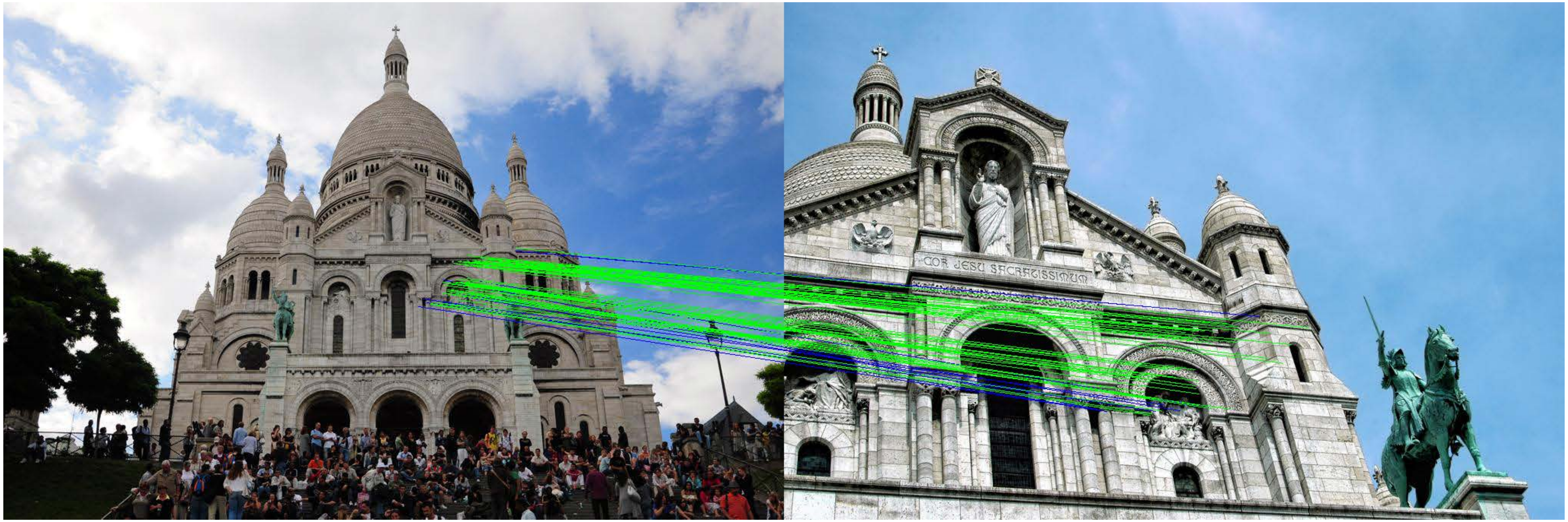
- Captures the global context at every layer output.
- Invariant to permutations of the input.
- Loss is the sum of a classification and a regression term

Outlier rejection



RANSAC

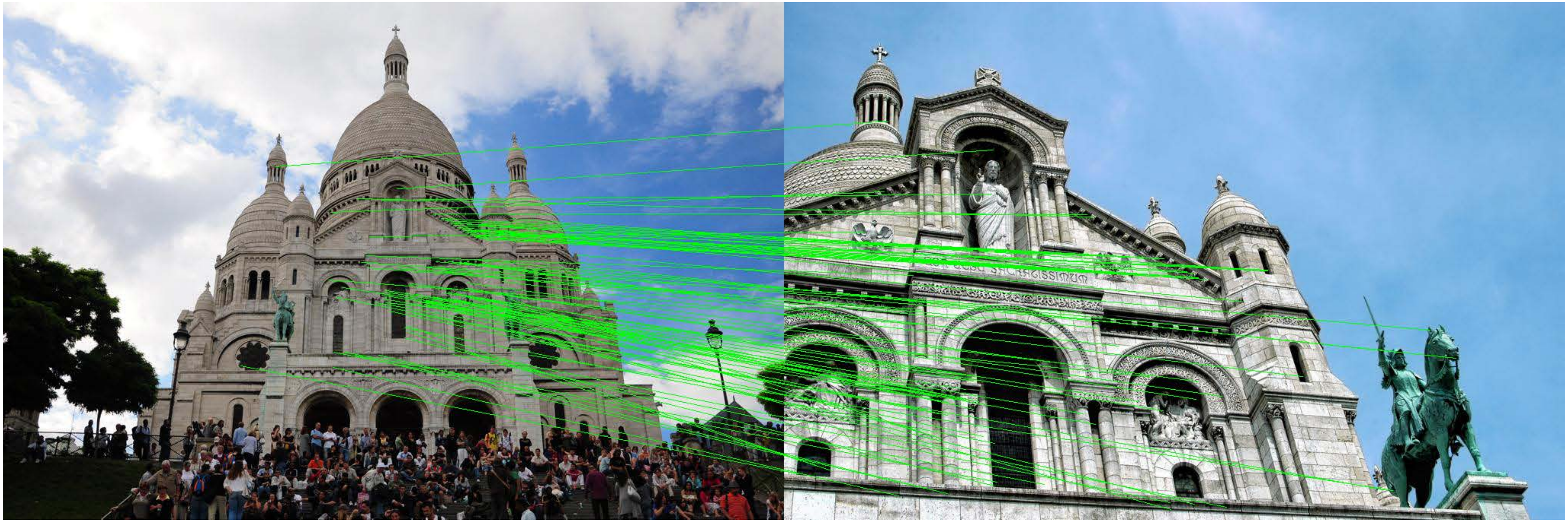
Outlier rejection



Grid-Based Motion Statistics

Bian et al, CVPR 17

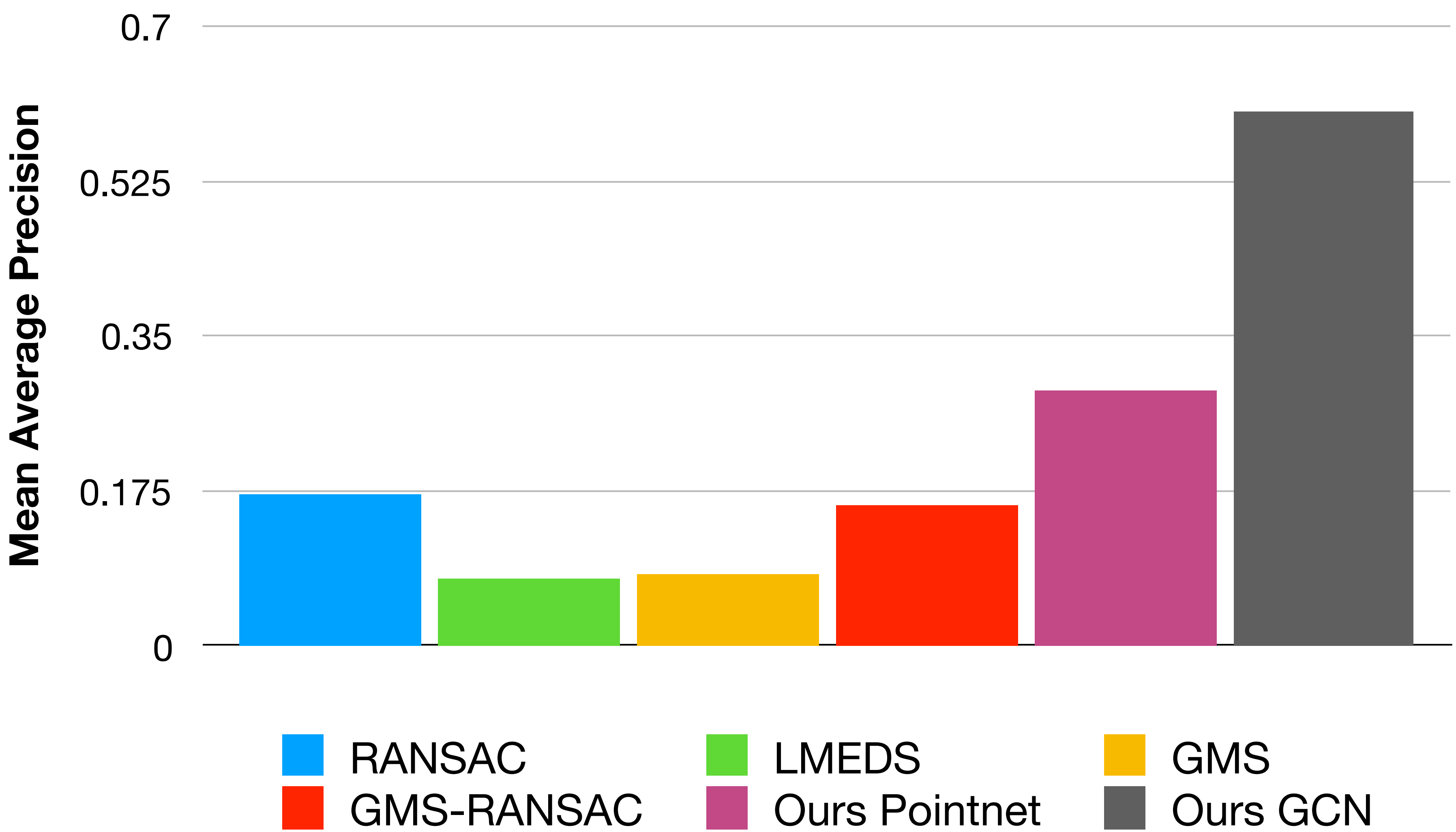
Outlier rejection



Our results

Mean Accuracy = ratio of pairs below error threshold of X, while X goes from zero to 20 (degrees) —> AUC

antitative Results



1000 randomly chosen pairs from
Yahoo Flickr Creative Commons 100 millions

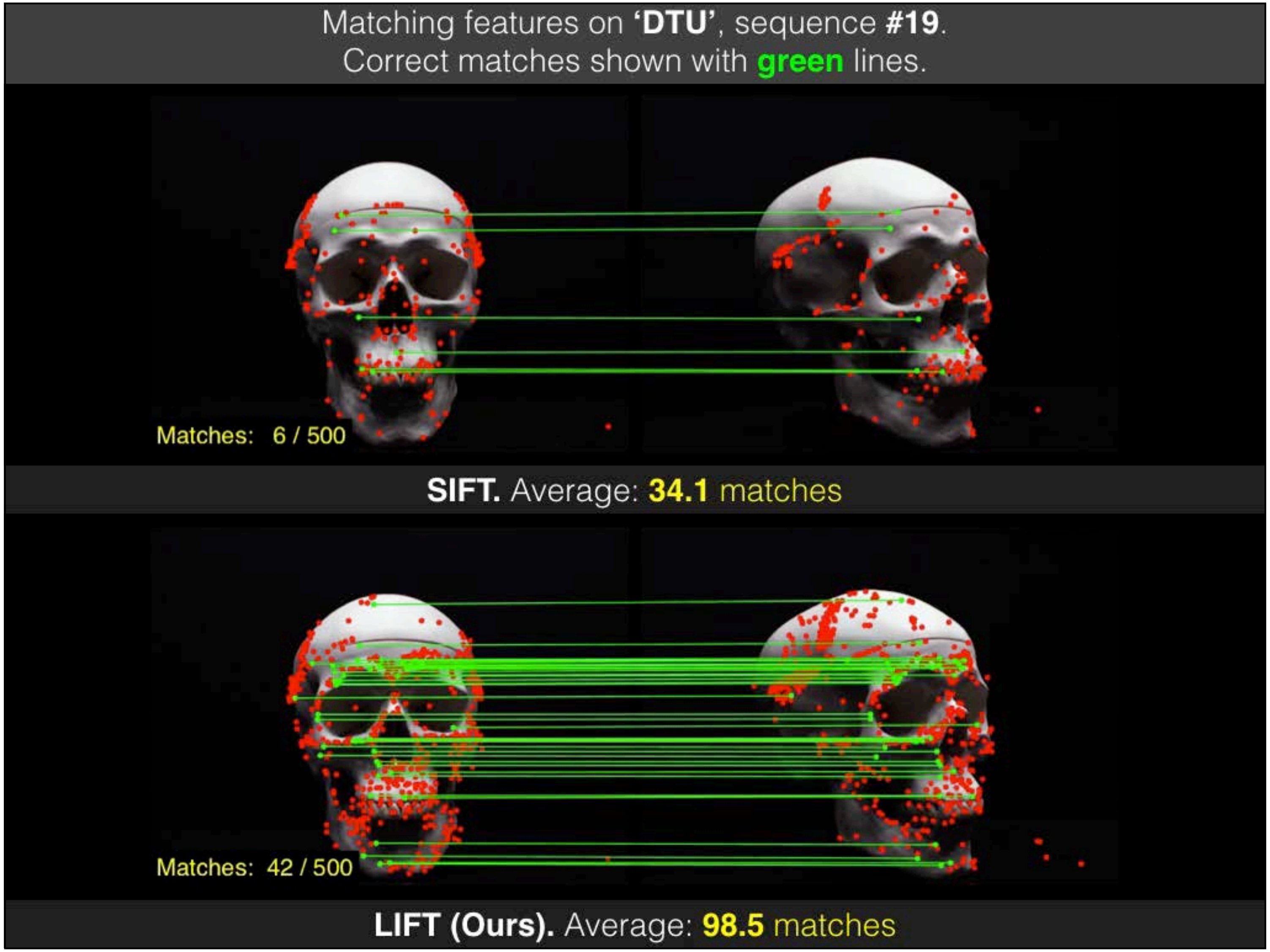
Conclusion

- We implemented the **full keypoint extraction pipeline** using Deep Networks while preserving end-to-end differentiability.
- We showed how to train it **effectively** and **outperform** the state-of-the-art.
- We are now working on reformulating the extraction **and** matching problem as end-to-end trainable CNN.

Software

Source code and pre-trained models are available for every component of the pipeline:

- ✓ TILDE detector:
  - **github.com/cvlab-epfl/TILDE**
- ✓ Orientation estimator:
  - **github.com/cvlab-epfl/learn-orientation**
- ✓ Descriptors:
  - **github.com/cvlab-epfl/deepdesc-release**
- ✓ One LIFT to rule them all:
  - **github.com/cvlab-epfl/tf-lift**



Thank you. Questions?