SSY098 - Image Analysis

Lecture 13 - Visual Localization and Feature Learning

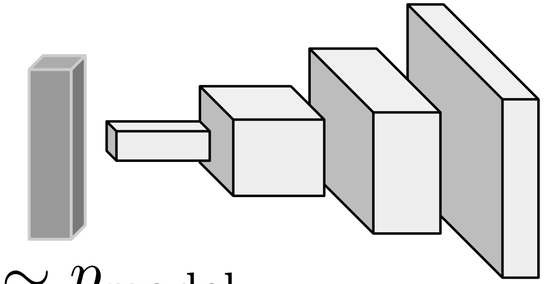
Torsten Sattler

Last Lecture

Jan. 20	Introduction, Linear classifiers and filtering	
Jan. 23	Filtering, gradients, scale	Lab 1
Jan. 27	Local features	
Jan. 30	Learning a classifier	Lab 2
Feb. 3	Convolutional neural networks	
Feb. 6	More convolutional neural networks	
Feb. 10	Robust model fitting and RANSAC	Lab 3
Feb. 13	Image registration	
Feb. 17	Camera Geometry	Lab 4
Feb. 20	More camera geometry	
Feb. 24	Generative neural networks	
Feb. 27	**Generative neural networks**	
Mar. 2	Visual Localization & Feature Learning	
Mar. 9	No lecture	

Generative Adversarial Networks (GANs)

Motivation: Learn loss function



$\mathbf{z} \sim p_{\text{model}}$

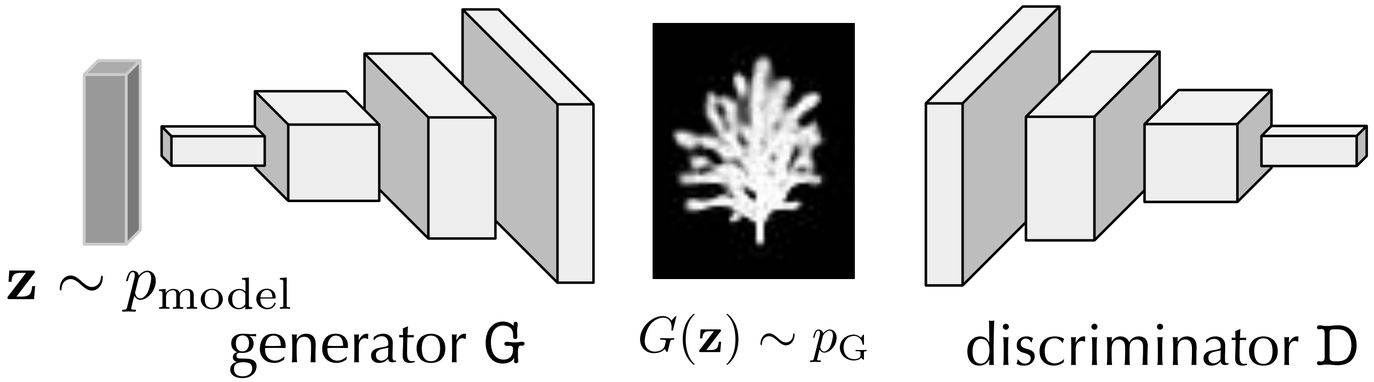
generator G

$G(\mathbf{z}) \sim p_{\text{G}}$

[Goodfellow et al., Generative Adversarial Nets, NIPS 2014]

Generative Adversarial Networks (GANs)

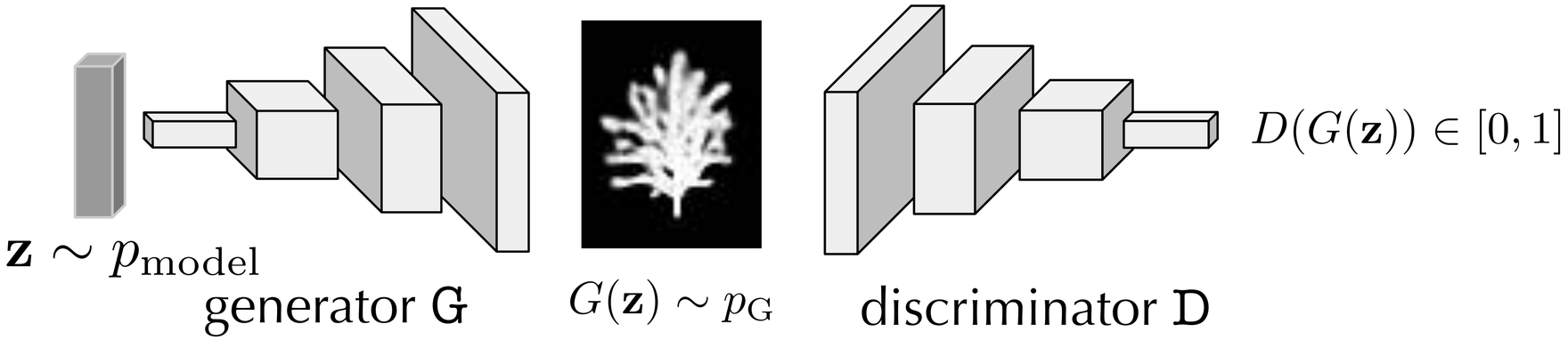
Motivation: Learn loss function



[Goodfellow et al., Generative Adversarial Nets, NIPS 2014]

Generative Adversarial Networks (GANs)

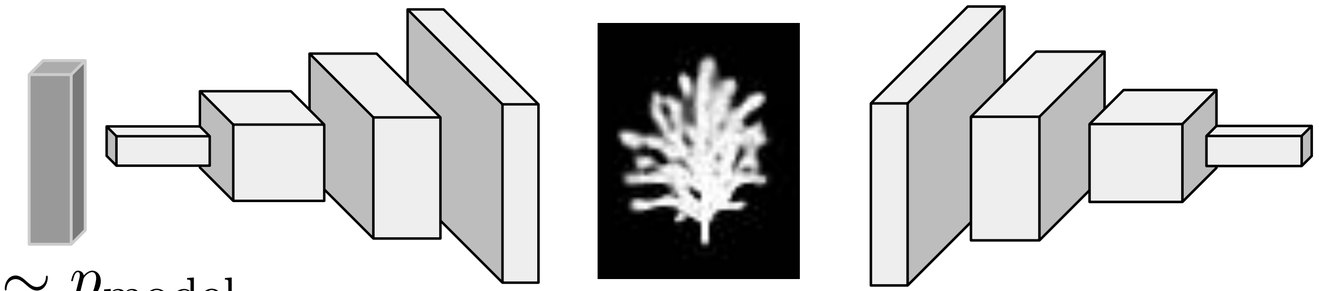
Motivation: Learn loss function



[Goodfellow et al., Generative Adversarial Nets, NIPS 2014]

Generative Adversarial Networks (GANs)

Motivation: Learn loss function



$\mathbf{z} \sim p_{\text{model}}$

generator G

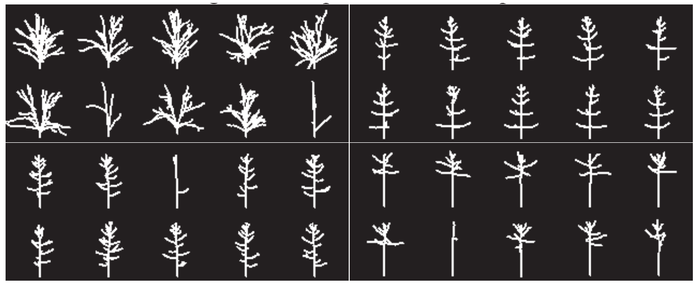
$G(\mathbf{z}) \sim p_{\text{G}}$

discriminator D

$D(G(\mathbf{z})) \in [0, 1]$

real data

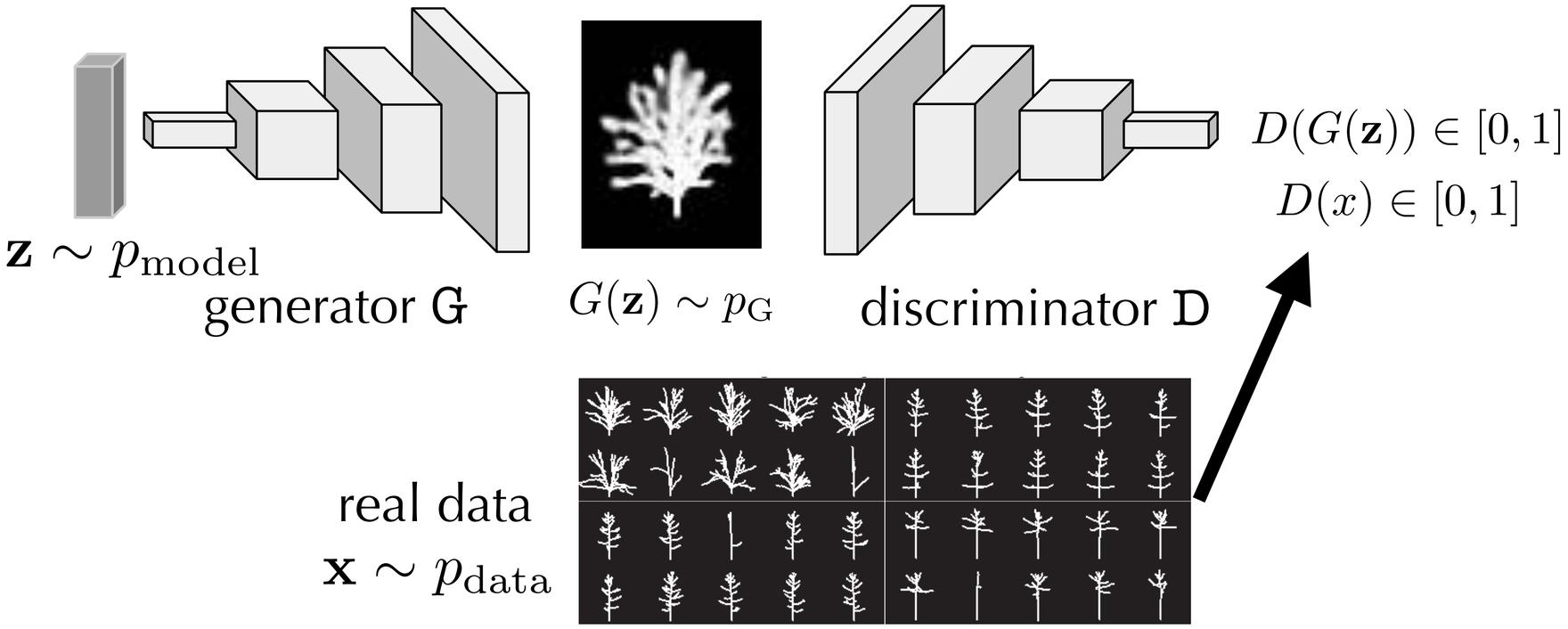
$\mathbf{x} \sim p_{\text{data}}$



[Goodfellow et al., Generative Adversarial Nets, NIPS 2014]

Generative Adversarial Networks (GANs)

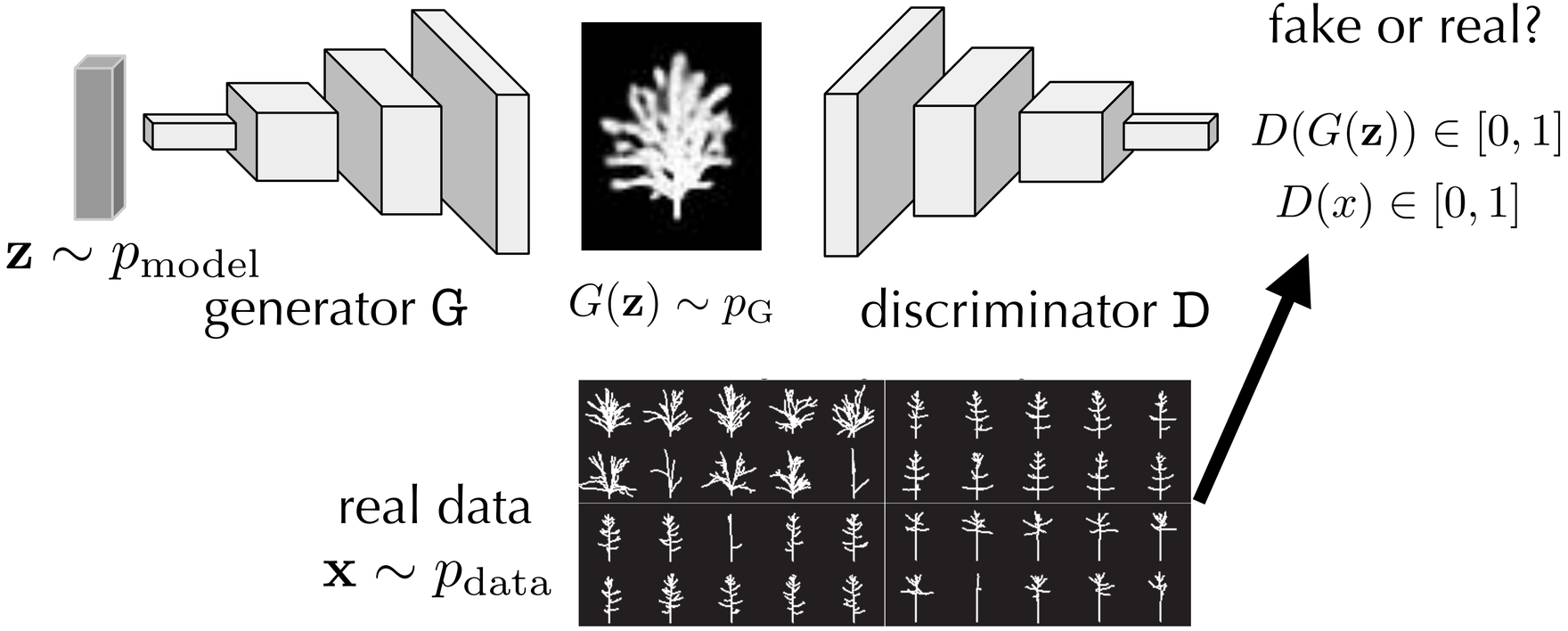
Motivation: Learn loss function



[Goodfellow et al., Generative Adversarial Nets, NIPS 2014]

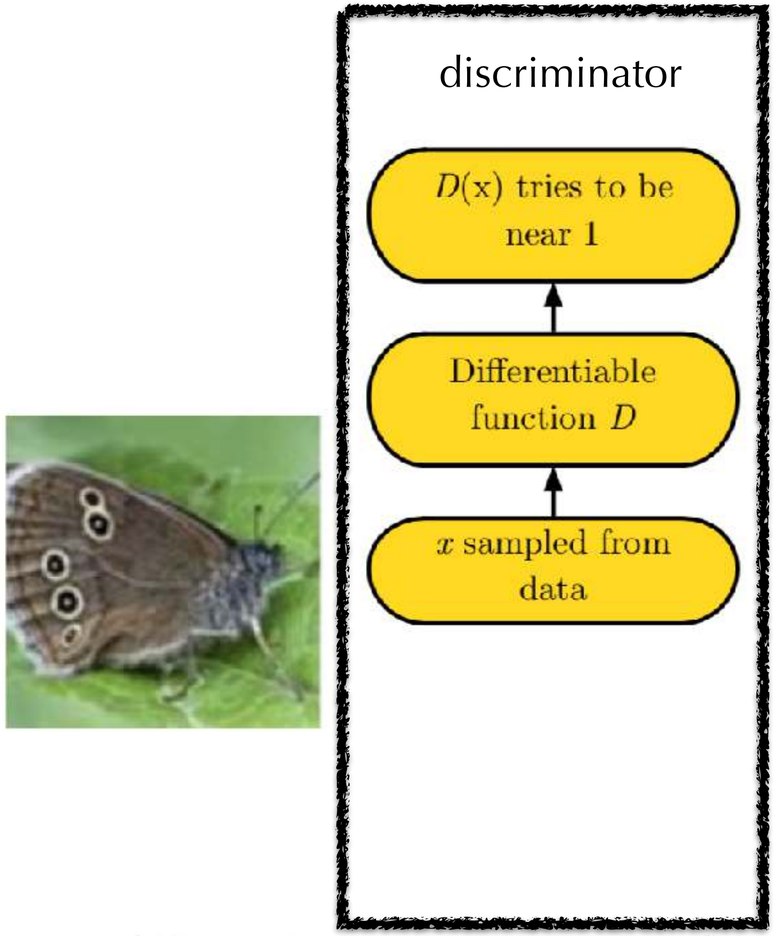
Generative Adversarial Networks (GANs)

Motivation: Learn loss function



[Goodfellow et al., Generative Adversarial Nets, NIPS 2014]

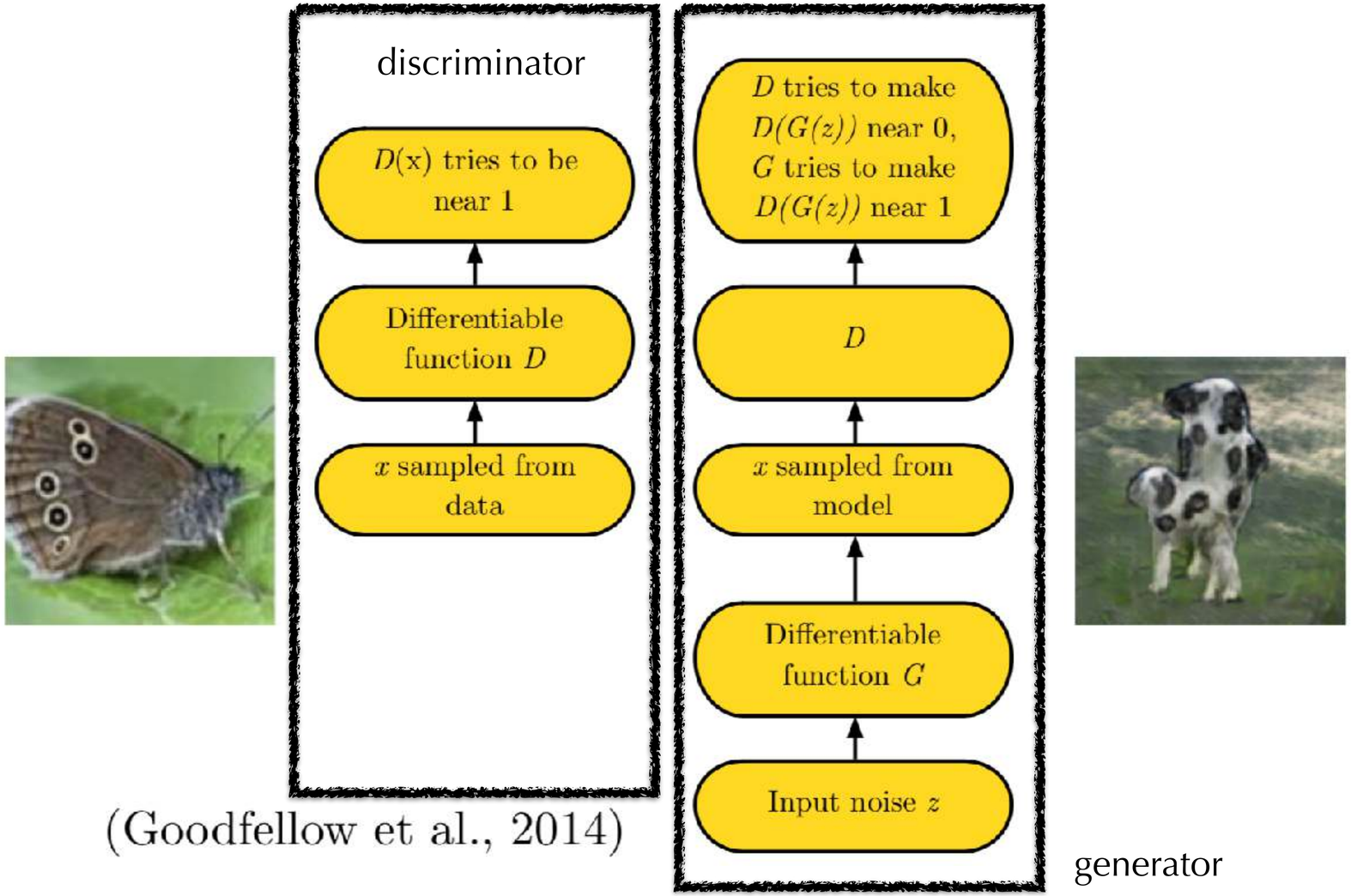
Generative Adversarial Networks (GANs)



(Goodfellow et al., 2014)

slide credit: Ian Goodfellow

Generative Adversarial Networks (GANs)



(Goodfellow et al., 2014)

slide credit: Ian Goodfellow

Game Theoretic View

- Discriminator D tries to distinguish real and fake images

Game Theoretic View

- Discriminator `D` tries to distinguish real and fake images
- Generator `G` tries to fool `D`, tries to make synthetic images look "as real as possible"

Game Theoretic View

- Discriminator `D` tries to distinguish real and fake images
- Generator `G` tries to fool `D`, tries to make synthetic images look "as real as possible"
- `G` and `D` play zero-sum minimax game:

Game Theoretic View

- Discriminator `D` tries to distinguish real and fake images
- Generator `G` tries to fool `D`, tries to make synthetic images look "as real as possible"
- `G` and `D` play zero-sum minimax game:

$$\max_{D} \quad \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log(d(\mathbf{x})) + \mathbb{E}_{\mathbf{z}\sim p_{\text{model}}} \log(1 - d(g(\mathbf{z})))$$

Game Theoretic View

- Discriminator `D` tries to distinguish real and fake images
- Generator `G` tries to fool `D`, tries to make synthetic images look "as real as possible"
- `G` and `D` play zero-sum minimax game:

$$\max_{D} \quad \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log(d(\mathbf{x})) + \mathbb{E}_{\mathbf{z}\sim p_{\text{model}}} \log(1 - d(g(\mathbf{z})))$$

Game Theoretic View

- Discriminator D tries to distinguish real and fake images
- Generator G tries to fool D, tries to make synthetic images look "as real as possible"
- G and D play zero-sum minimax game:

$$\min_{G} \max_{D} \quad \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log(d(\mathbf{x})) + \mathbb{E}_{\mathbf{z}\sim p_{\text{model}}} \log(1 - d(g(\mathbf{z})))$$

Game Theoretic View

- Discriminator D tries to distinguish real and fake images
- Generator G tries to fool D, tries to make synthetic images look "as real as possible"
- G and D play zero-sum minimax game:

$$\min_{G} \max_{D} \quad \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log(d(\mathbf{x})) + \mathbb{E}_{\mathbf{z}\sim p_{\text{model}}} \log(1 - d(g(\mathbf{z})))$$

Game Theoretic View

- Discriminator D tries to distinguish real and fake images
- Generator G tries to fool D, tries to make synthetic images look "as real as possible"
- G and D play zero-sum minimax game:

$$\min_{G} \max_{D} \quad \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log(d(\mathbf{x})) + \mathbb{E}_{\mathbf{z}\sim p_{\text{model}}} \log(1 - d(g(\mathbf{z})))$$

- Equilibrium / global optimum at $p_{\text{data}} = p_G$ i.e., when real and synthetic images are indistinguishable for D:

$$d(\mathbf{x}) = 1/2 \;\; \forall \mathbf{x} \sim p_{\text{data}}, \mathbf{x} \sim p_G$$

Why is Training GANs so Hard?

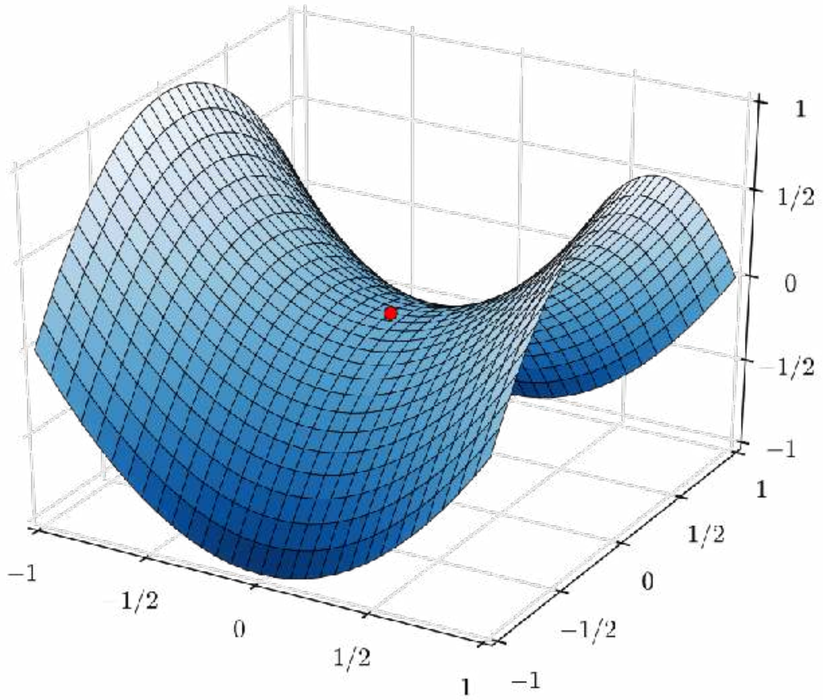


image: © Nicoguaro (CC Attribution Unported 3.0)

GAN equilibria are saddle points

$$\min_G \max_D \quad \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log(d(\mathbf{x})) + \mathbb{E}_{\mathbf{z}\sim p_{\text{model}}} \log(1 - d(g(\mathbf{z})))$$

[Goodfellow et al., Generative Adversarial Nets, NIPS 2014]

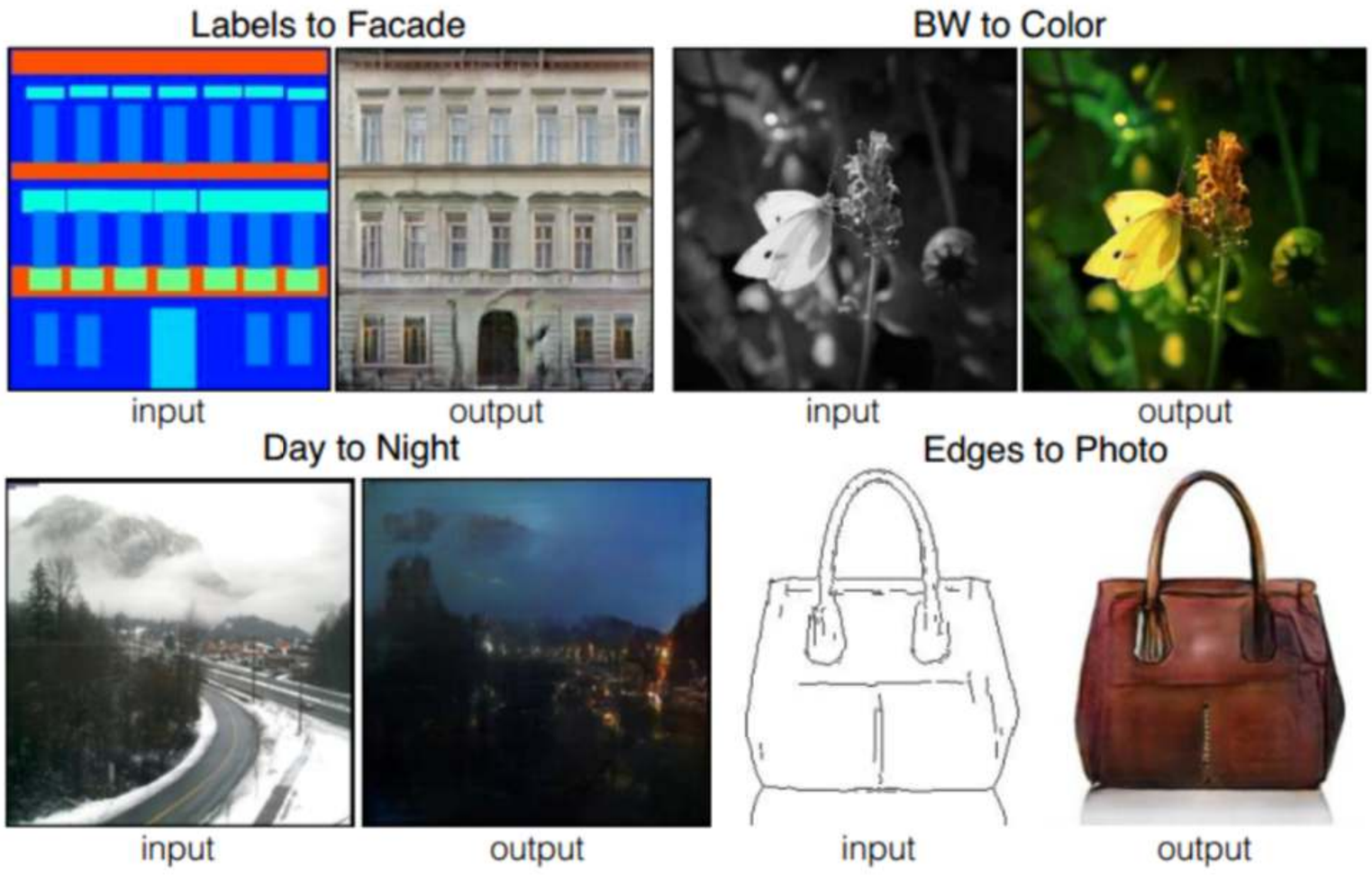
WGANs vs. GANs

GAN

WGAN

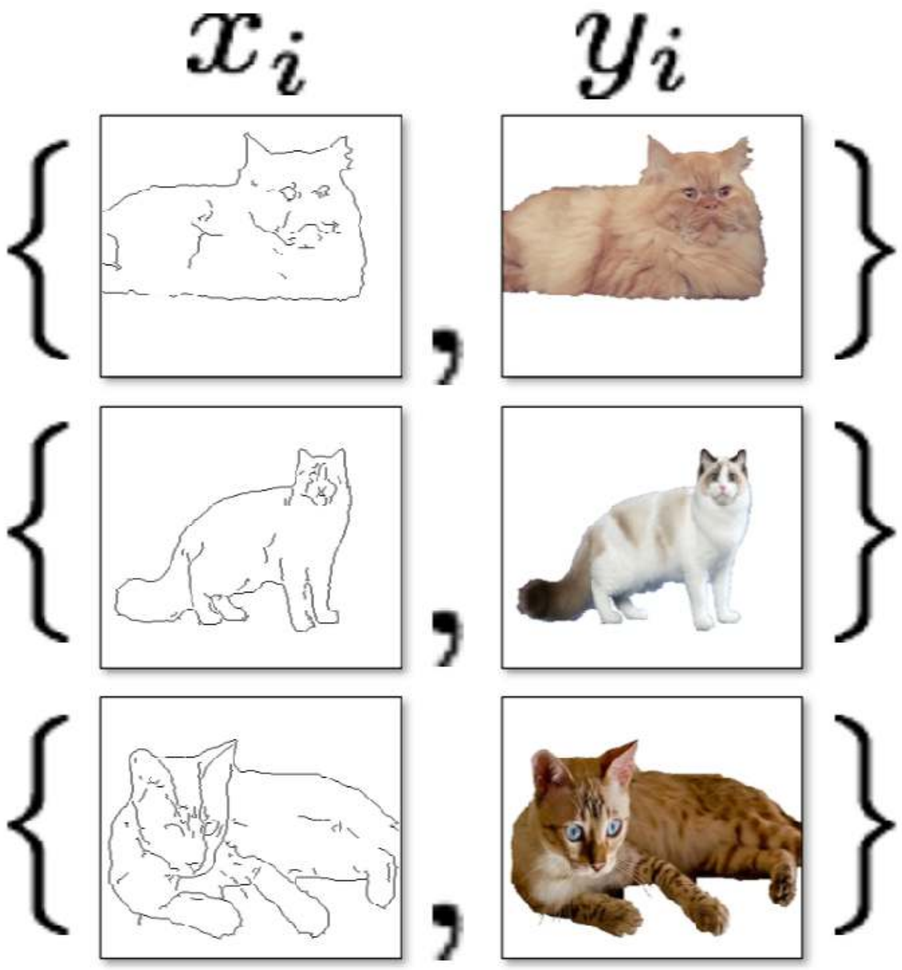
[Arjovsky et al., Wasserstein GAN, arXiv:1701.07875]

Image-To-Image Translation



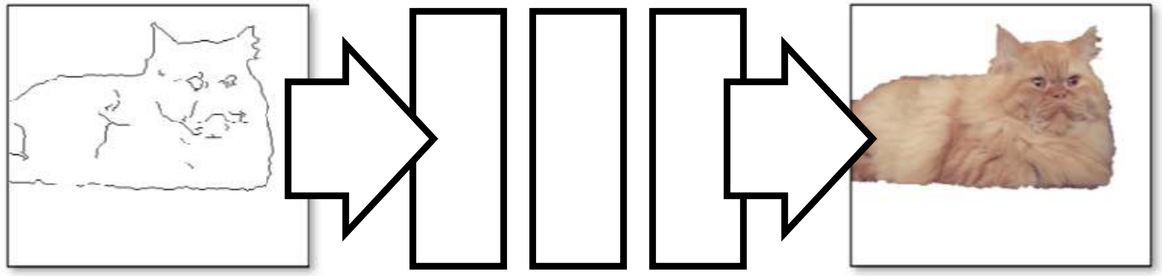
slide credit: Jun-Yan Zhu, Taesung Park

Supervised Image-To-Image Translation



slide credit: Jun-Yan Zhu, Taesung Park

Supervised Image-To-Image Translation

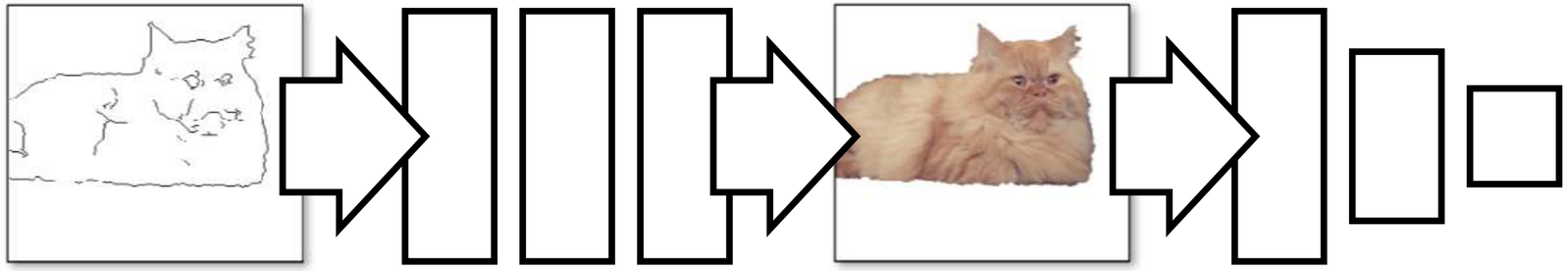


generator G

$$\min_G \max_D E_{\mathbf{y}}[\log(D(\mathbf{y}))] + E_{\mathbf{x}}[\log(1 - D(G(\mathbf{x})))]$$

slide credit: Jun-Yan Zhu, Taesung Park

Supervised Image-To-Image Translation



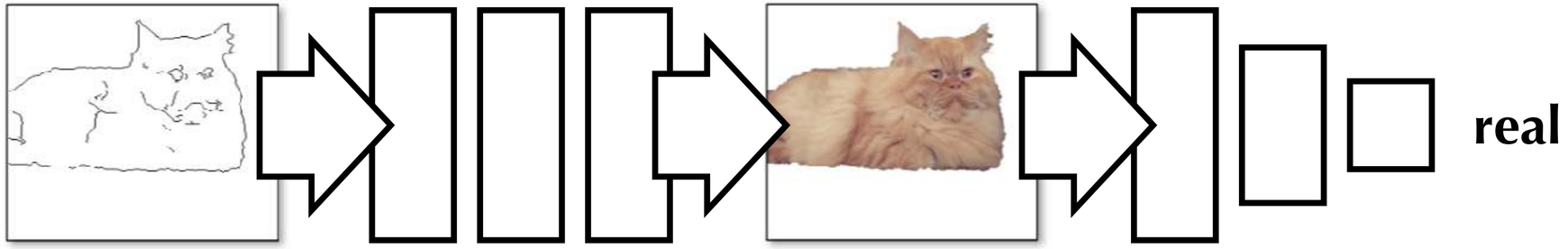
generator G

discriminator D

$$\min_G \max_D E_{\mathbf{y}}[\log(D(\mathbf{y}))] + E_{\mathbf{x}}[\log(1 - D(G(\mathbf{x})))]$$

slide credit: Jun-Yan Zhu, Taesung Park

Supervised Image-To-Image Translation



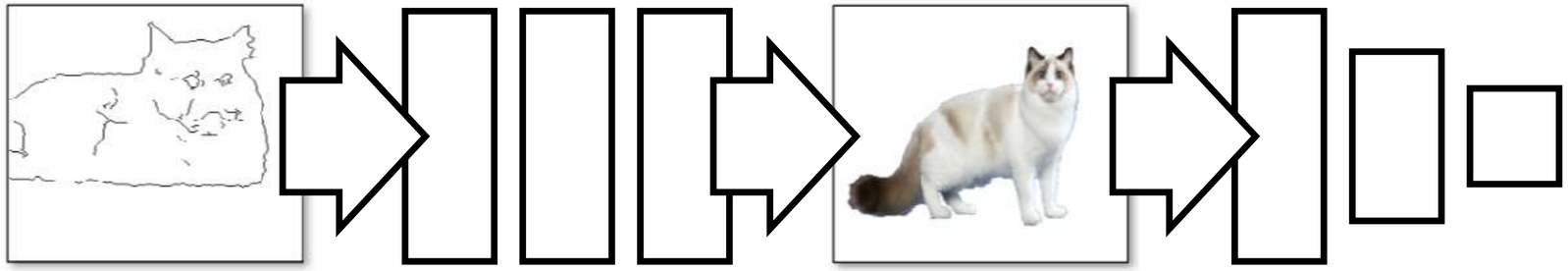
generator G

discriminator D

$$\min_G \max_D E_{\mathbf{y}}[\log(D(\mathbf{y}))] + E_{\mathbf{x}}[\log(1 - D(G(\mathbf{x})))]$$

slide credit: Jun-Yan Zhu, Taesung Park

Supervised Image-To-Image Translation



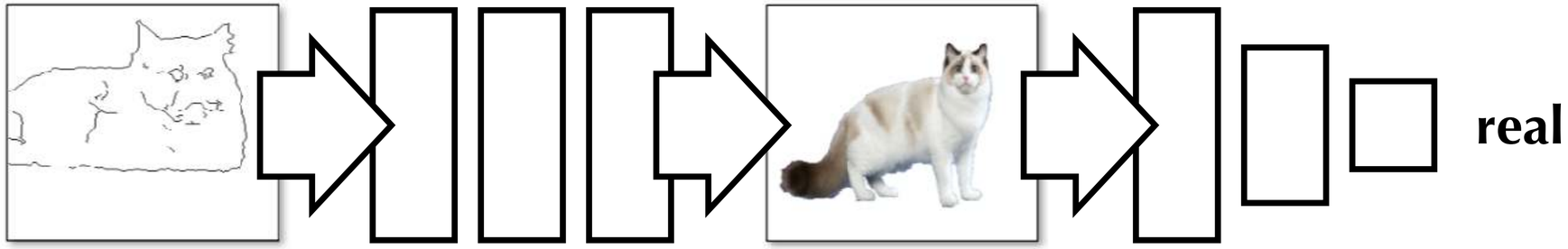
generator G

discriminator D

$$\min_G \max_D E_{\mathbf{y}}[\log(D(\mathbf{y}))] + E_{\mathbf{x}}[\log(1 - D(G(\mathbf{x})))]$$

slide credit: Jun-Yan Zhu, Taesung Park

Supervised Image-To-Image Translation



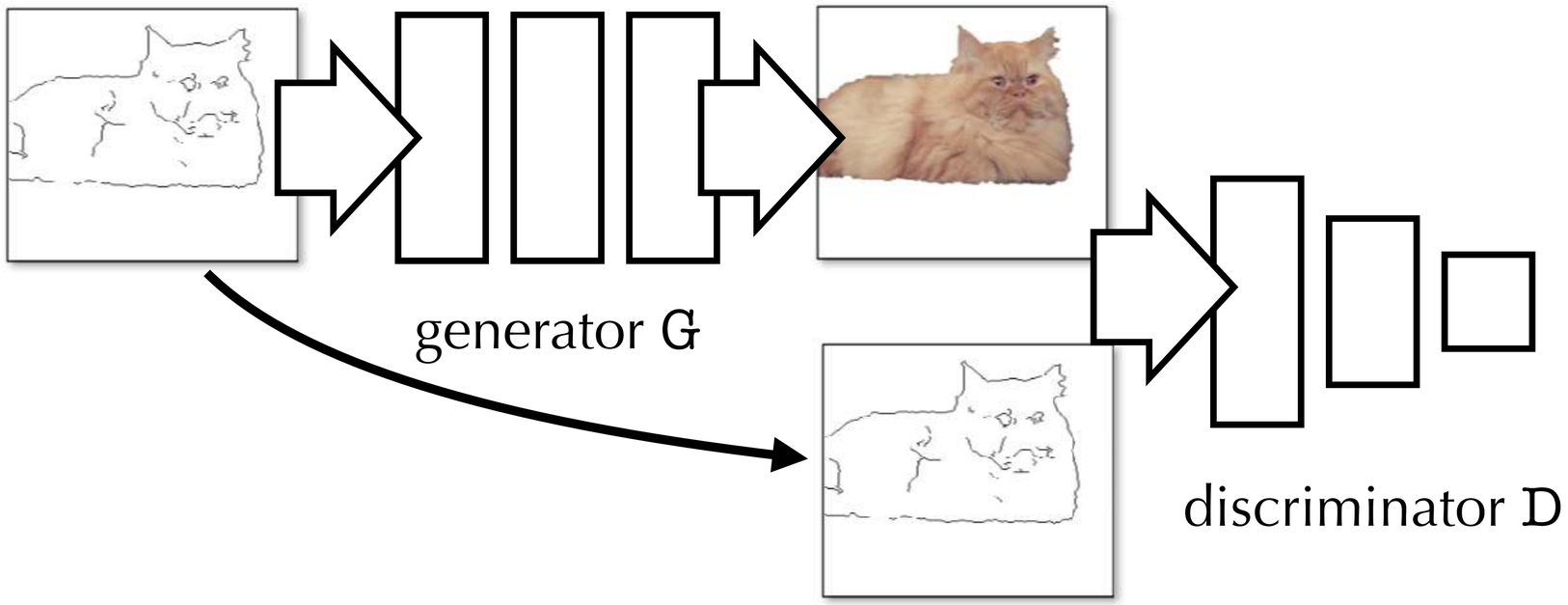
generator G

discriminator D

$$\min_G \max_D E_{\mathbf{y}}[\log(D(\mathbf{y}))] + E_{\mathbf{x}}[\log(1 - D(G(\mathbf{x})))]$$

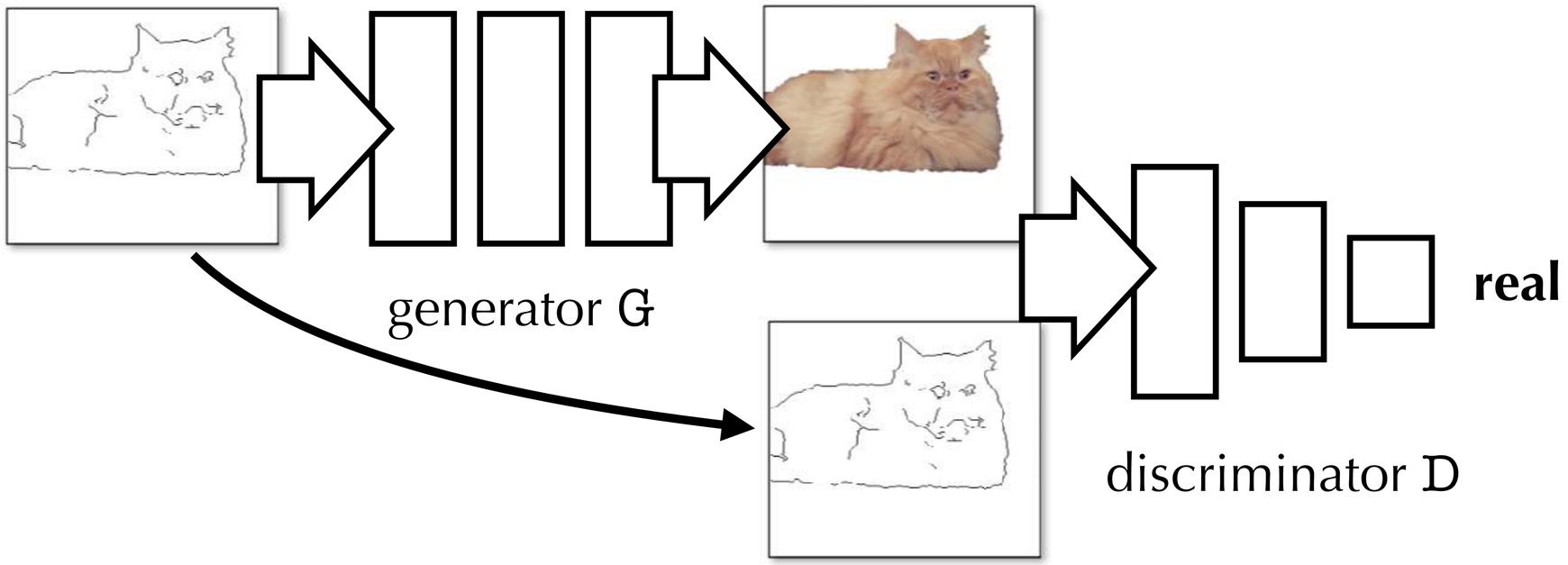
slide credit: Jun-Yan Zhu, Taesung Park

Conditional GANs



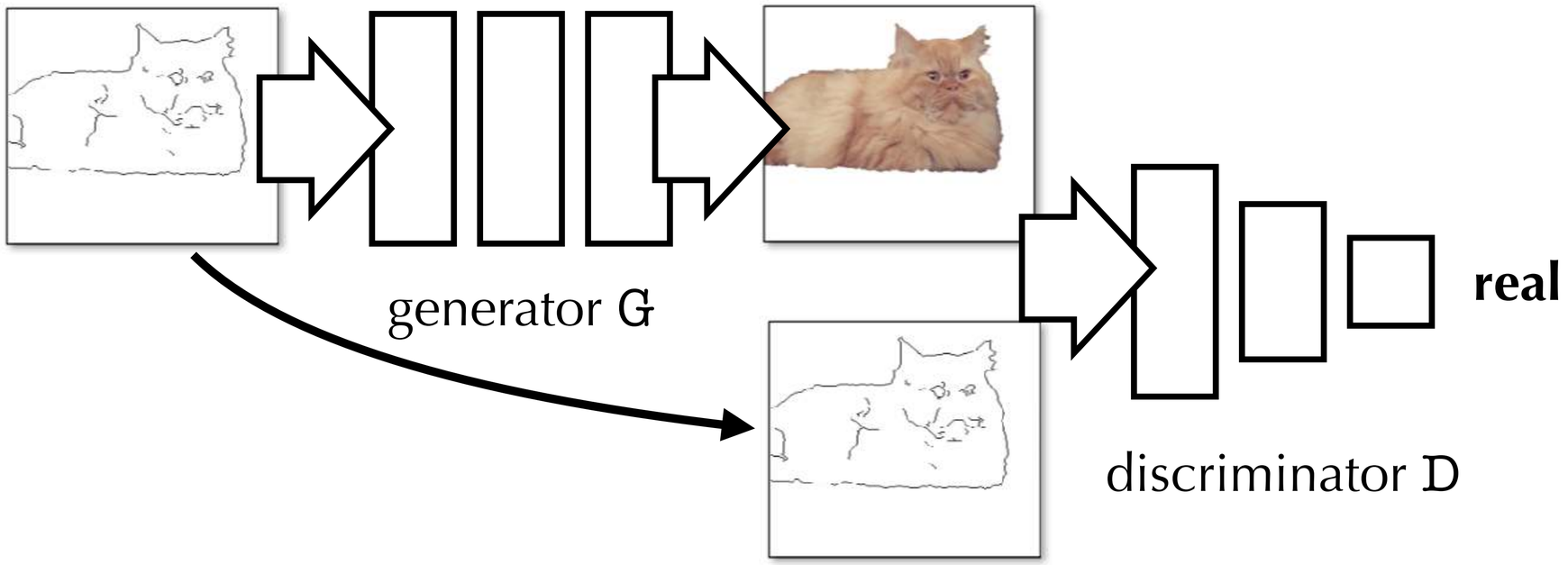
[Isola et al., Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017]

Conditional GANs



[Isola et al., Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017]

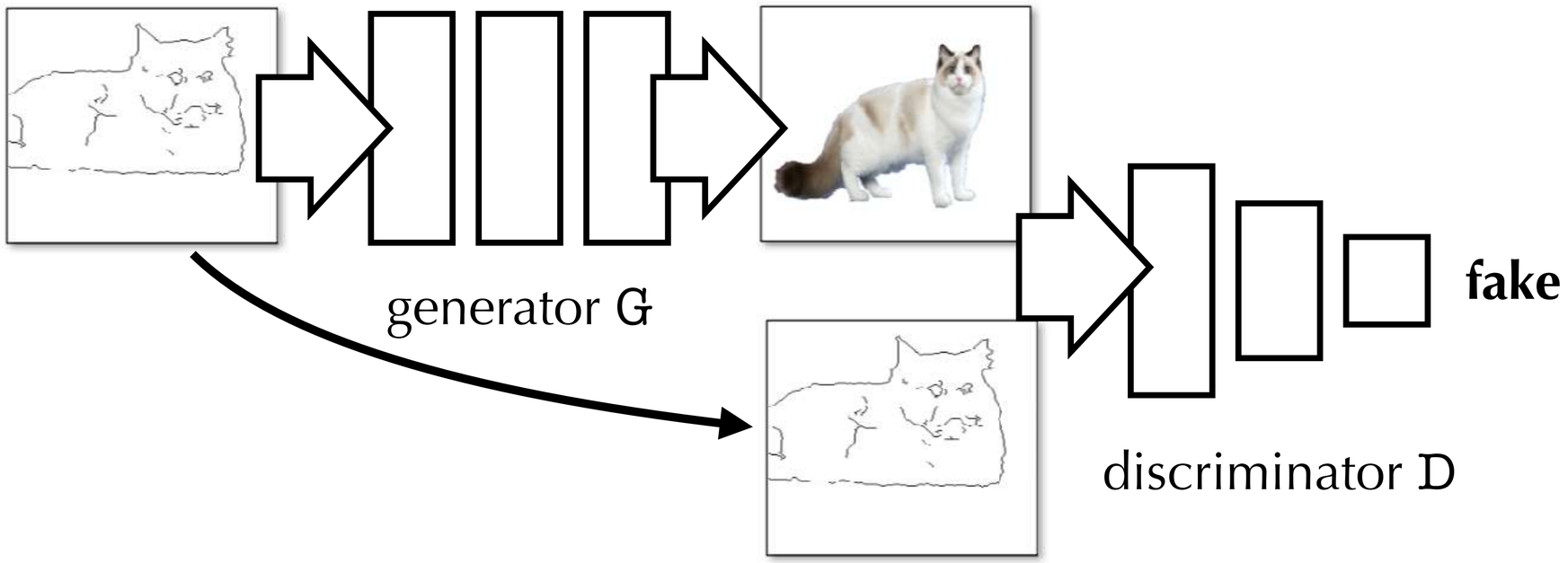
Conditional GANs



$$\min_G \max_D E_{\mathbf{x},\mathbf{y}}[\log(D(\mathbf{x},\mathbf{y}))] + E_{\mathbf{x},\mathbf{y}}[\log(1 - D(\mathbf{x}, G(\mathbf{x})))]$$

[Isola et al., Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017]

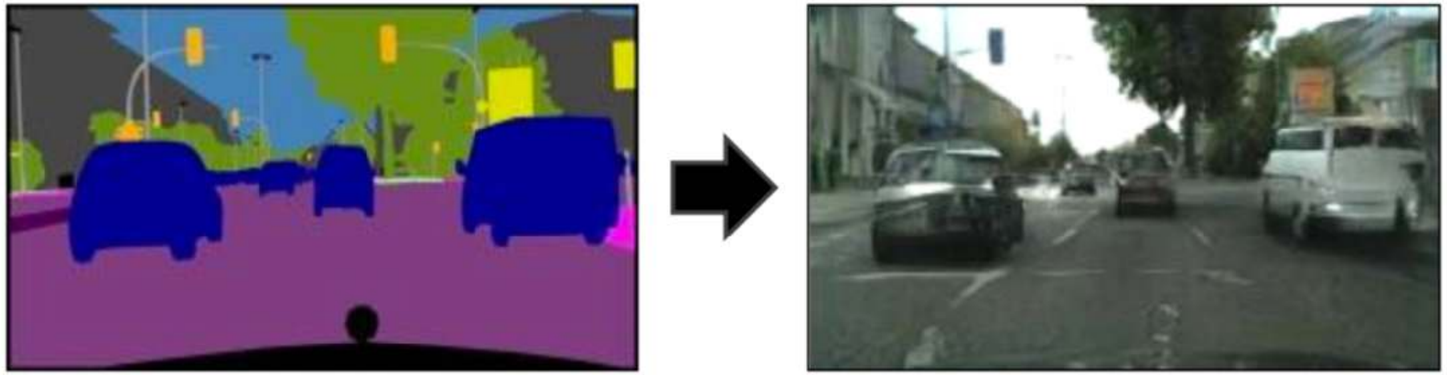
Conditional GANs



$$\min_G \max_D E_{\mathbf{x},\mathbf{y}}[\log(D(\mathbf{x},\mathbf{y}))] + E_{\mathbf{x},\mathbf{y}}[\log(1 - D(\mathbf{x}, G(\mathbf{x})))]$$

[Isola et al., Image-to-Image Translation with Conditional Adversarial Networks, CVPR 2017]

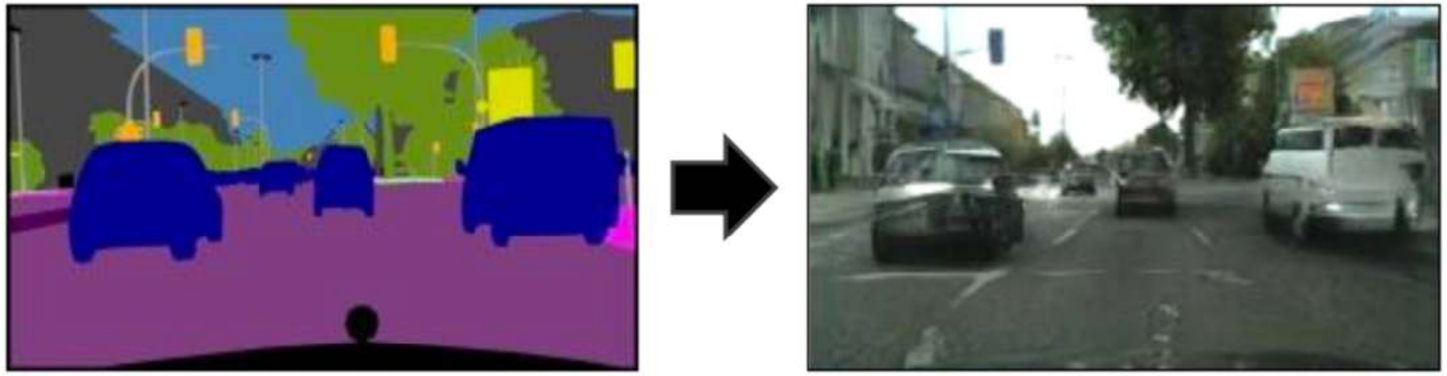
Supervised Image-To-Image Translation



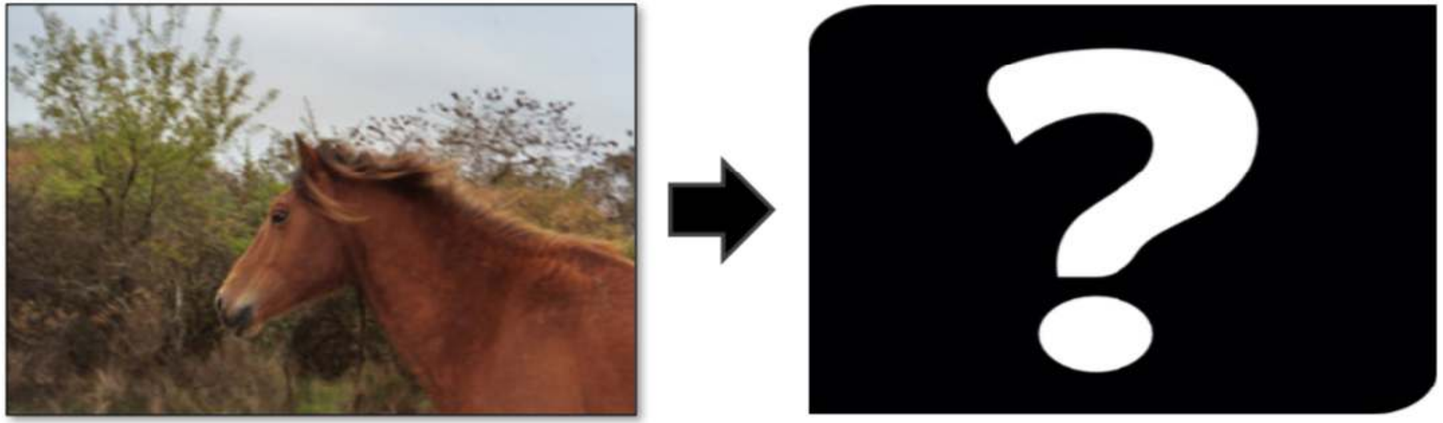
Label ↔ photo: per-pixel labeling

slide credit: Jun-Yan Zhu, Taesung Park

Supervised Image-To-Image Translation



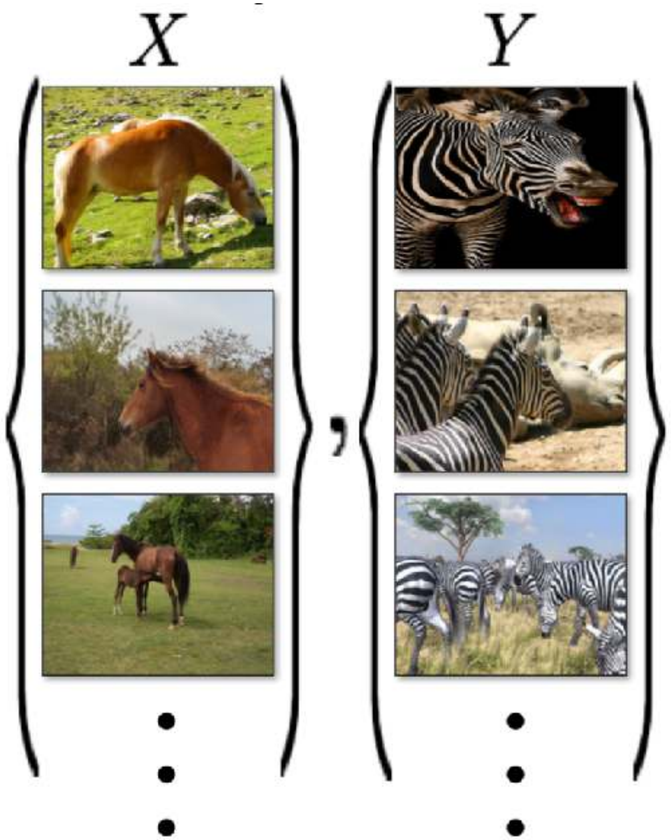
Label ↔ photo: per-pixel labeling



Labelled data often hard or impossible to obtain

slide credit: Jun-Yan Zhu, Taesung Park

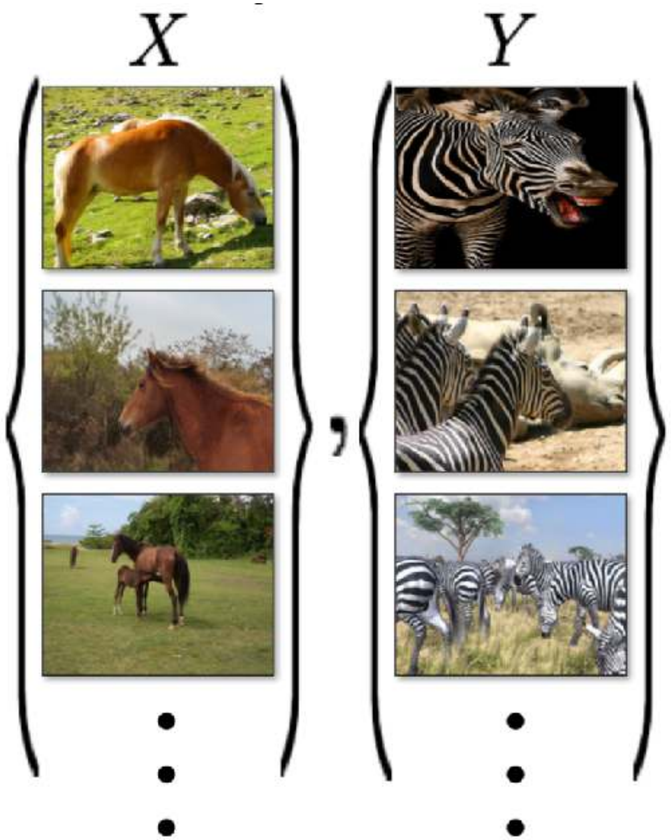
Unsupervised Image-To-Image Translation



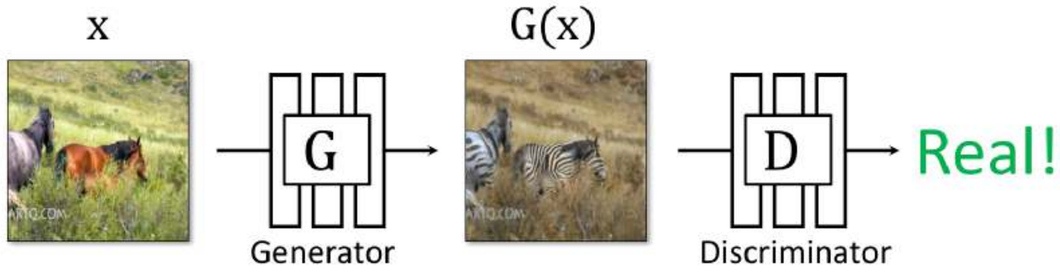
independent sets,
no pairing!

slide credit: Jun-Yan Zhu, Taesung Park

Unsupervised Image-To-Image Translation

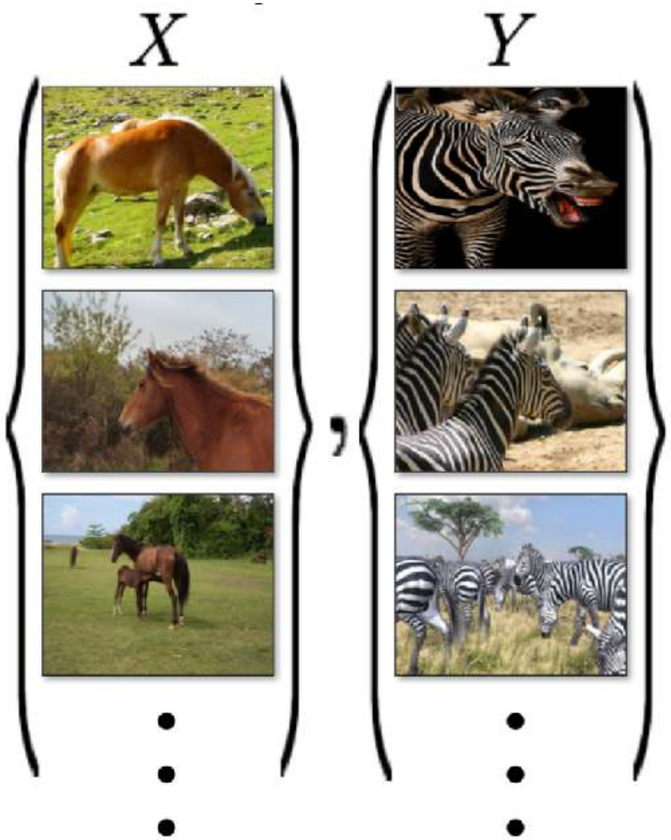


independent sets,
no pairing!

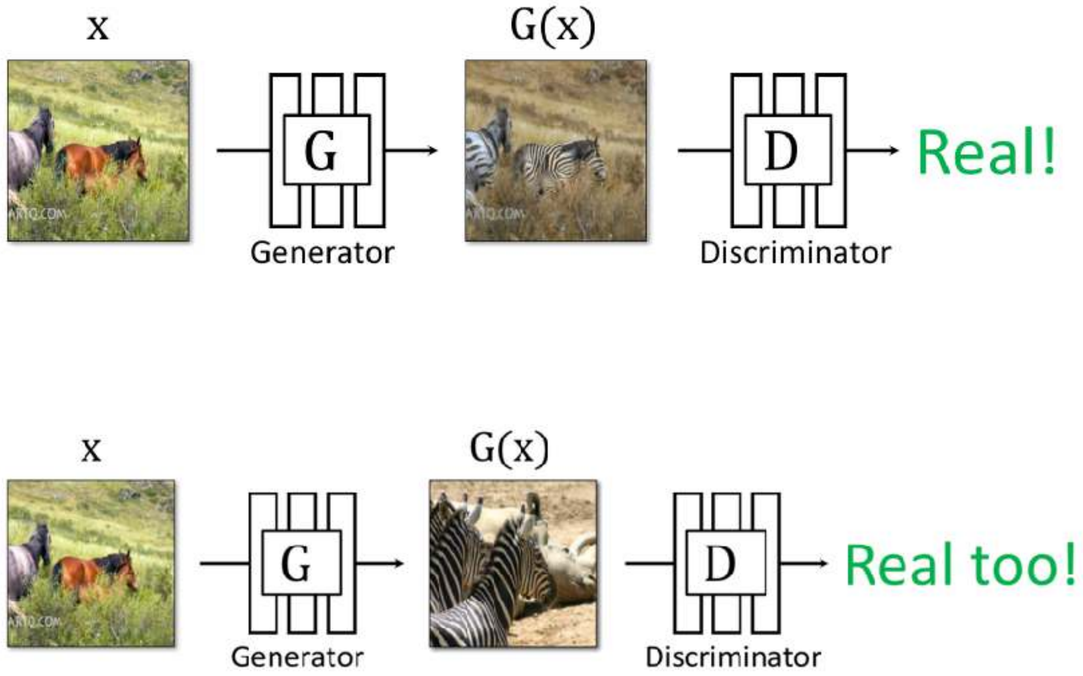


slide credit: Jun-Yan Zhu, Taesung Park

Unsupervised Image-To-Image Translation

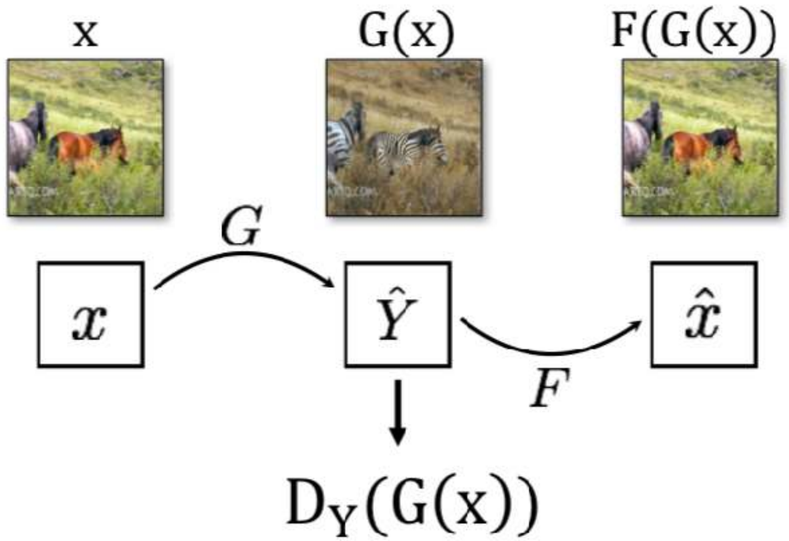


independent sets,
no pairing!



slide credit: Jun-Yan Zhu, Taesung Park

CycleGan: Cycle -Consistent Adversarial Networks

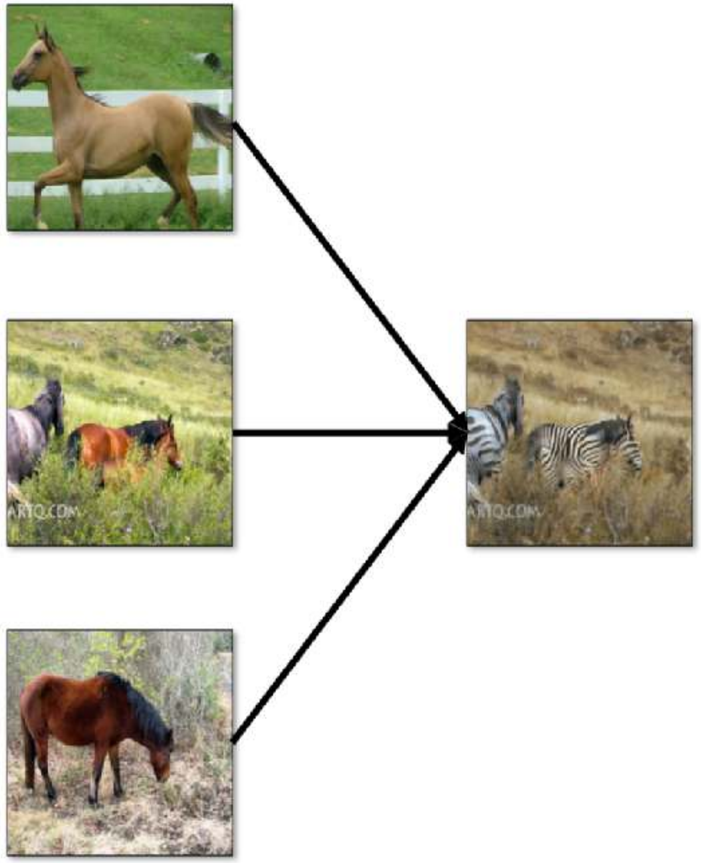


[Zhu & Park et al., Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017]

slide credit: Jun-Yan Zhu, Taesung Park

CycleGan: Cycle -Consistent Adversarial Networks

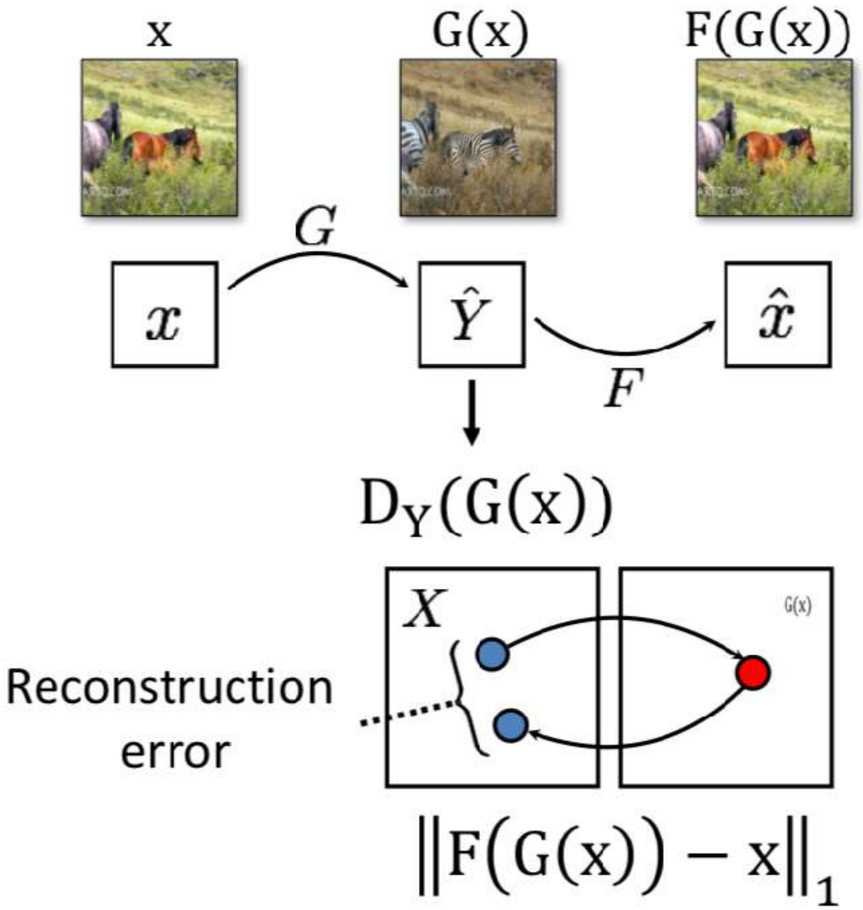
Does not prevent mode collapse



[Zhu & Park et al., Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017]

slide credit: Jun-Yan Zhu, Taesung Park

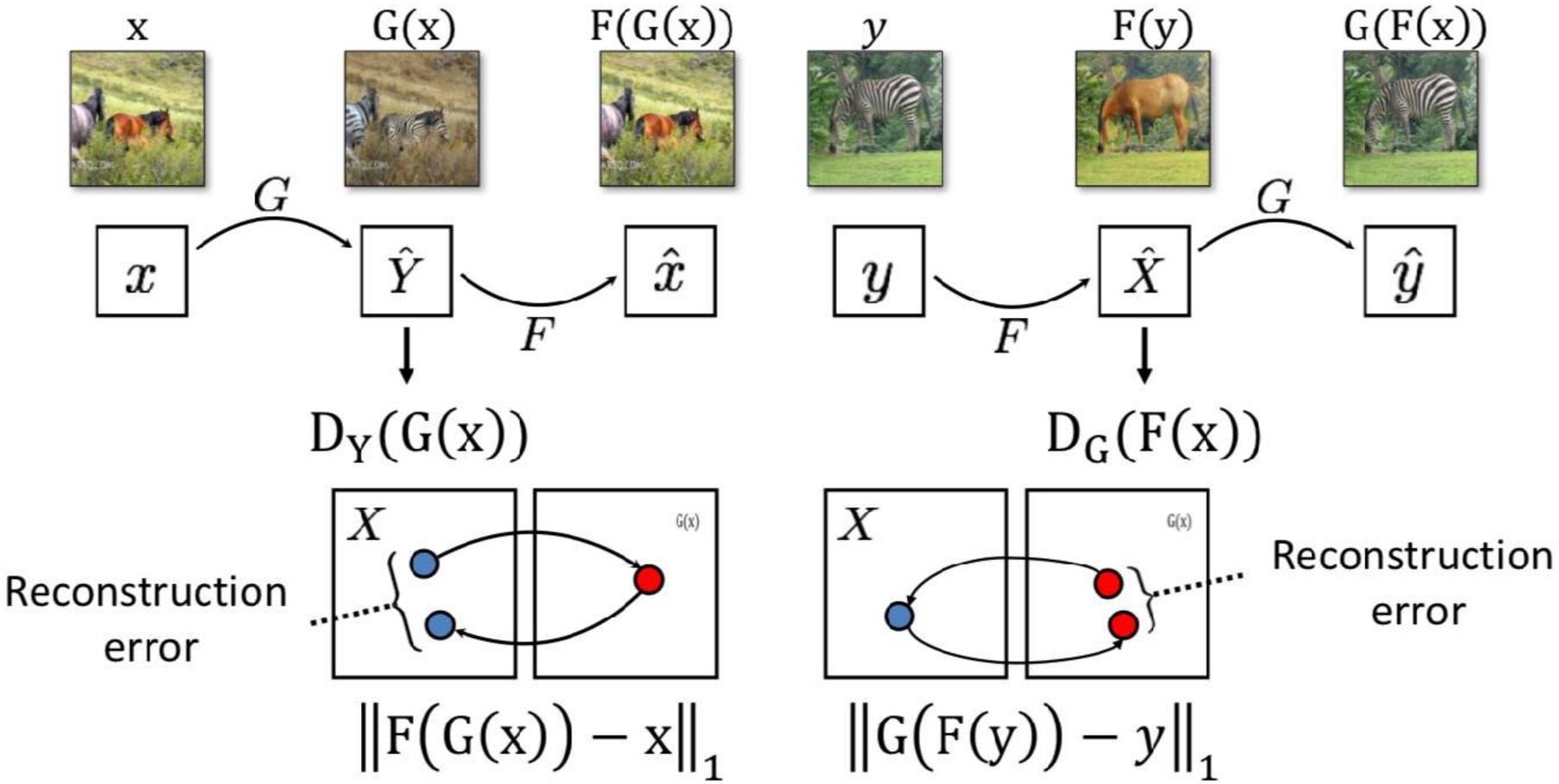
CycleGan: Cycle -Consistent Adversarial Networks



[Zhu & Park et al., Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017]

slide credit: Jun-Yan Zhu, Taesung Park

CycleGan: Cycle -Consistent Adversarial Networks



[Zhu & Park et al., Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, ICCV 2017]

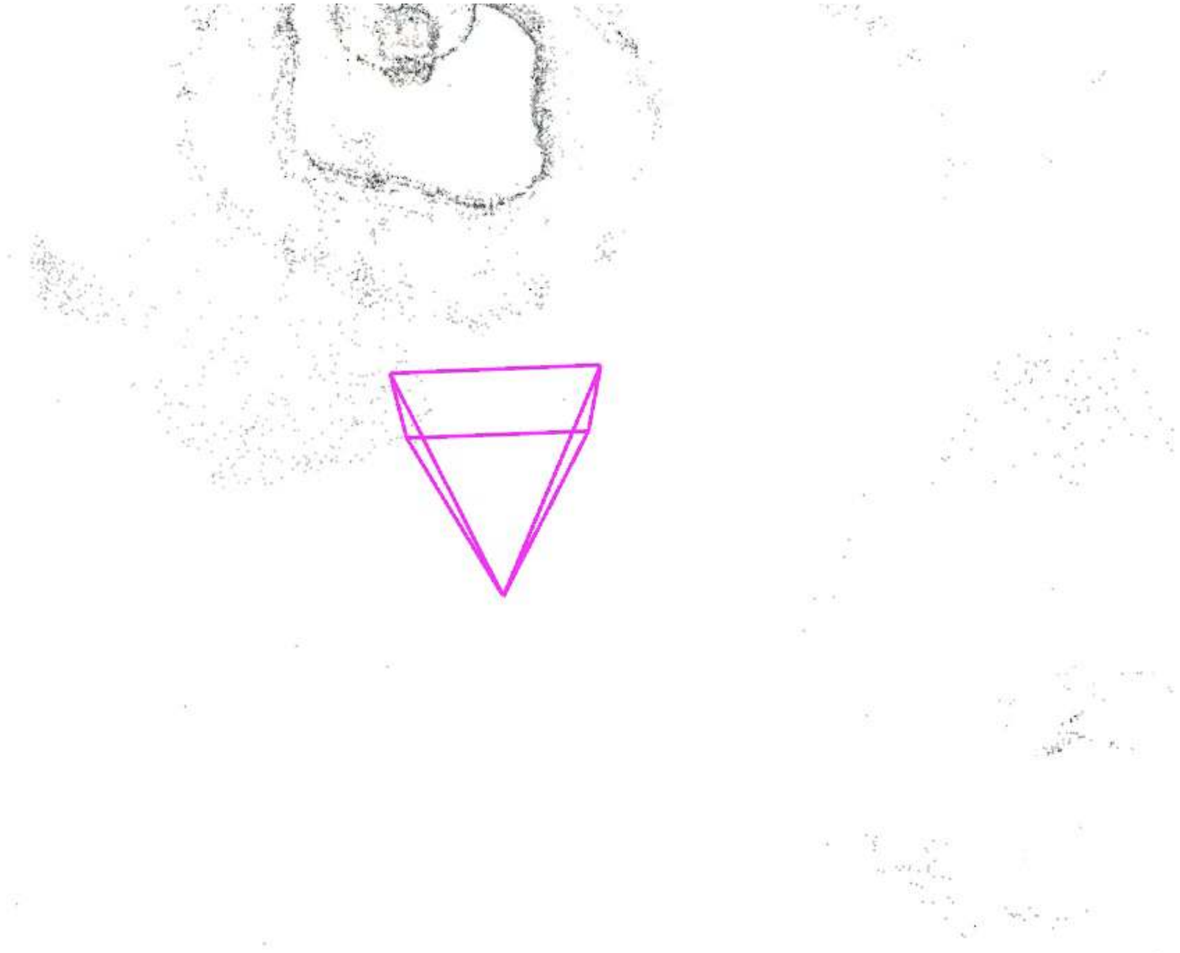
slide credit: Jun-Yan Zhu, Taesung Park

Today

Visual Localization and Feature Learning

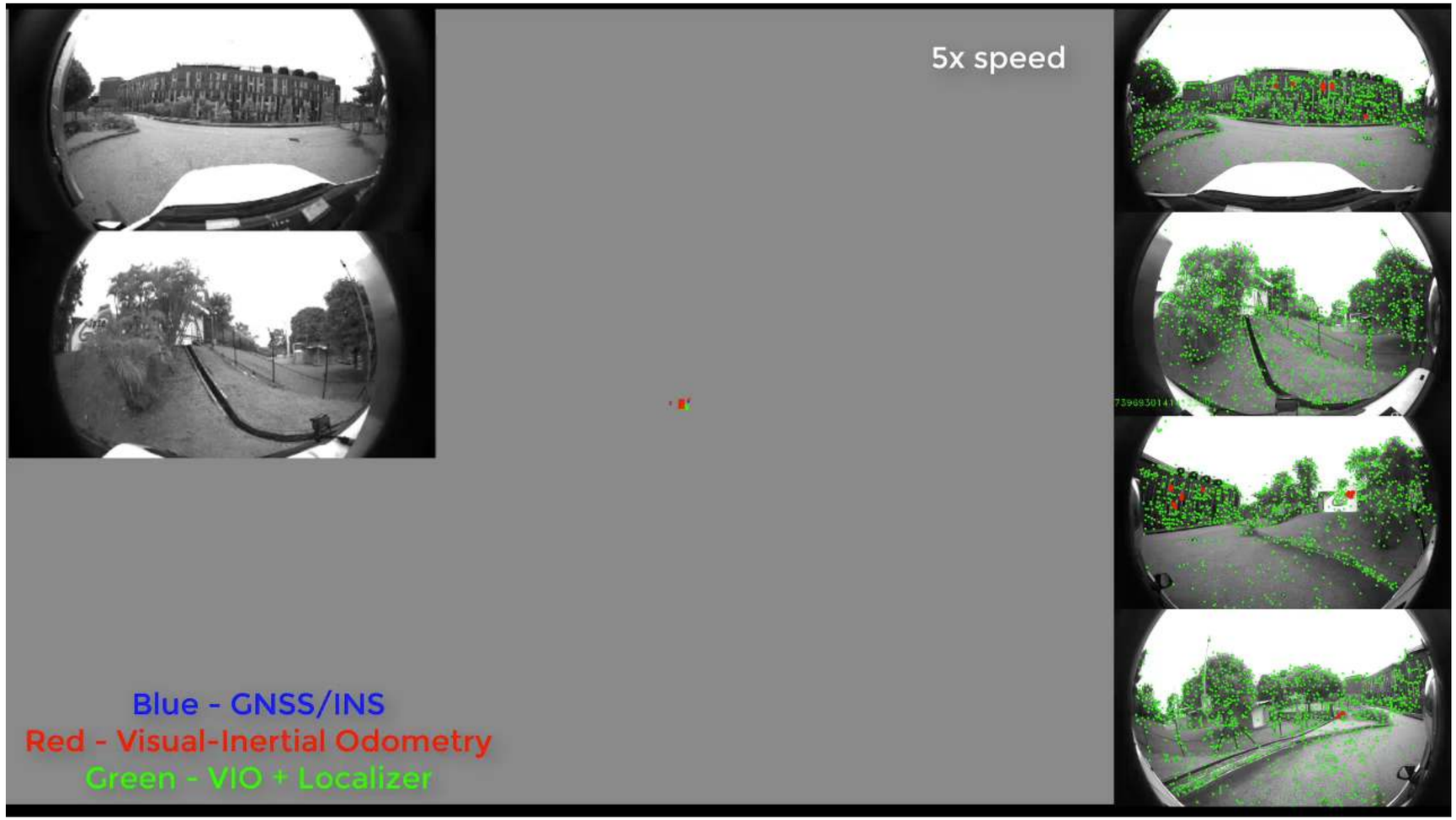
- **Visual Localization**
- Learning Feature Detectors

The Visual Localization Problem



Compute **exact position and orientation** of query image

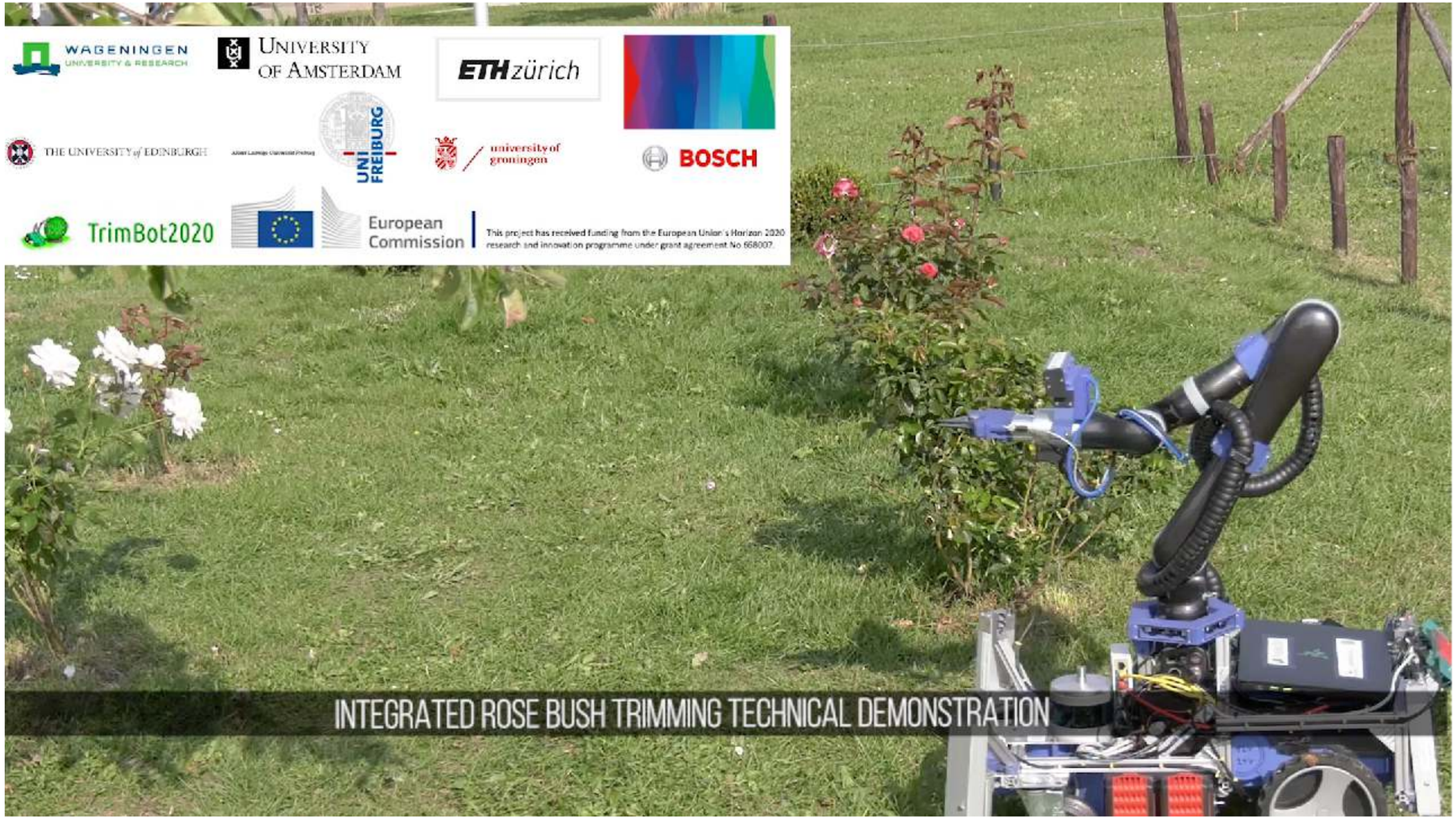
Applications: Autonomous Driving



AutoVision
3D Vision for Autonomous Vehicles

[Geppert, Liu, Cui, Pollefeys, Sattler, Efficient 2D-3D Matching for Multi-Camera Visual Localization, ICRA 2019]

Application: Robotics



Horizon 2020
European Union funding
for Research & Innovation

rijksuniversiteit groningen

WAGENINGEN UNIVERSITY & RESEARCH

BOSCH

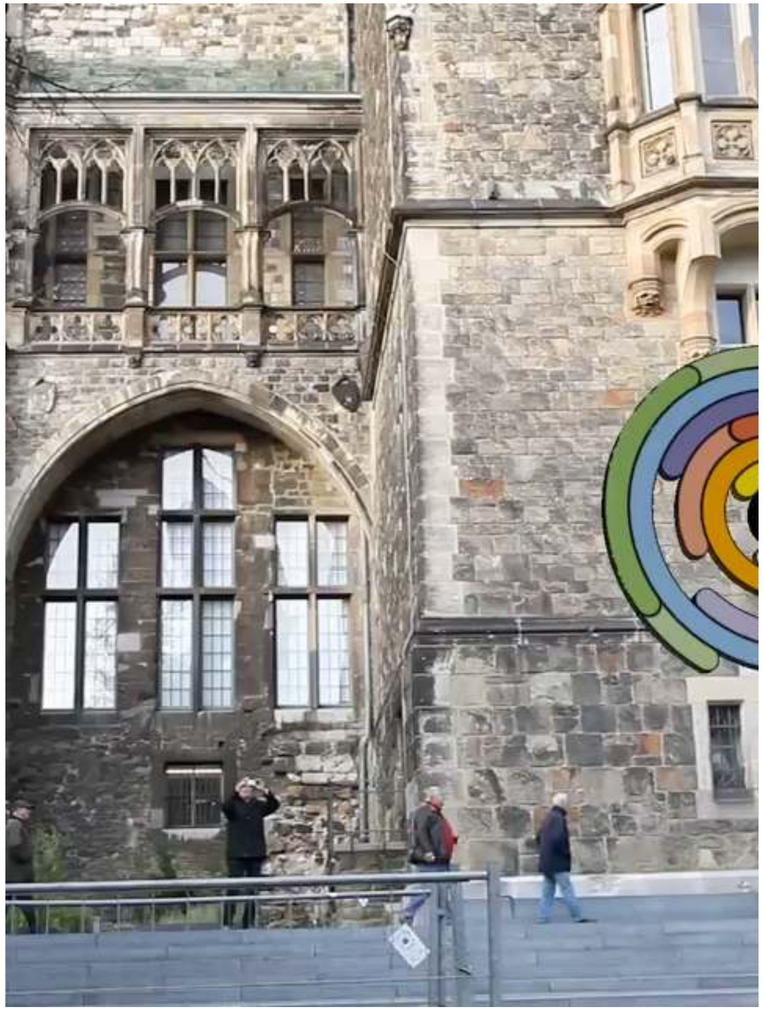
THE UNIVERSITY of EDINBURGH

UNIVERSITY OF AMSTERDAM

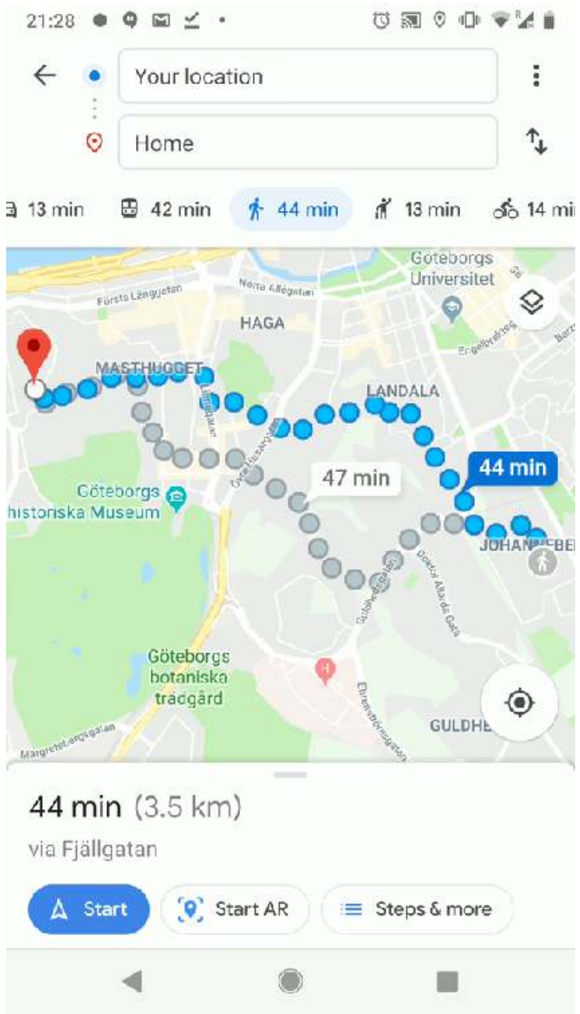
ETH zürich

UNI FREIBURG

Application: Augmented Reality

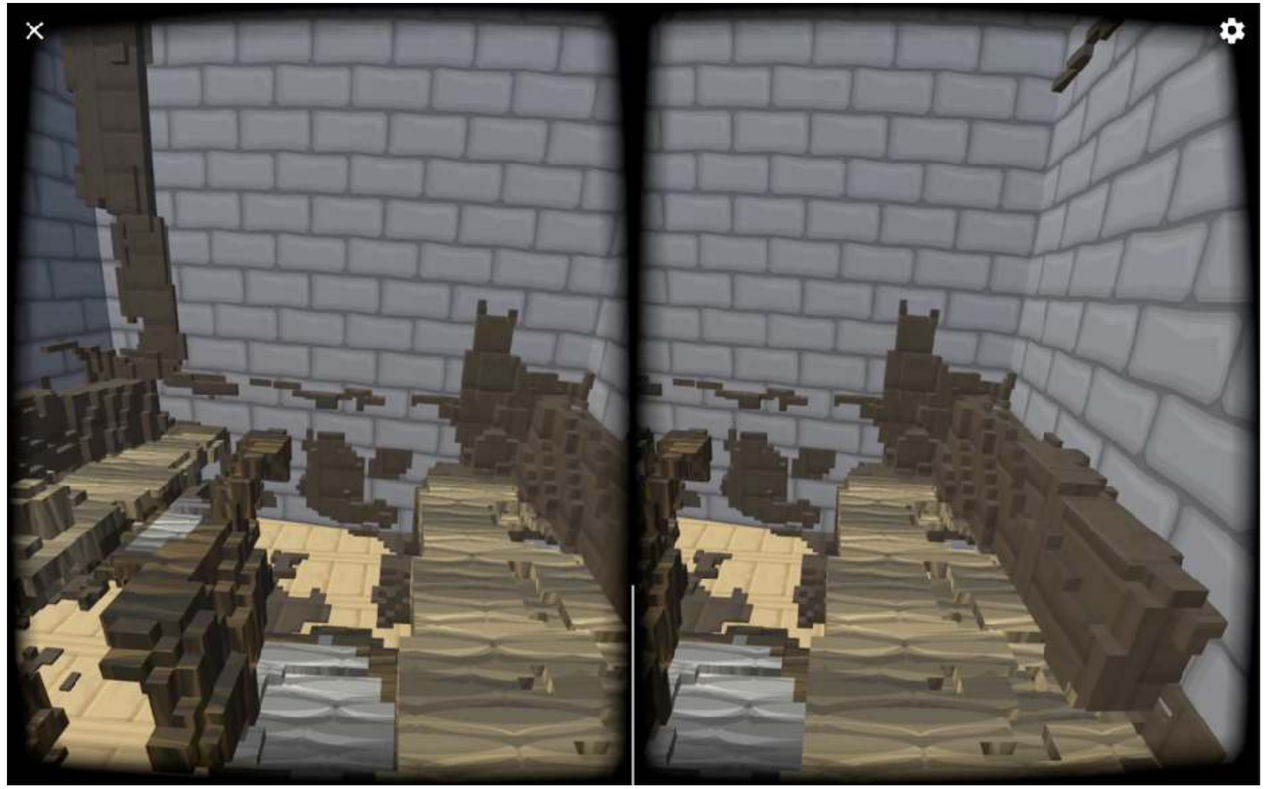


[Middelberg, Sattler, Untzelmann, Kobbelt, Scalable 6-DOF Localization on Mobile Devices, ECCV 2014]



AR navigation in Google Maps

Applications: Virtual Reality



Master thesis by Benjamin Steger, ETH Zurich
Textures: PureBDcraft ResourcePack by Sphax - BDcraft.net

Visual Localization Approaches

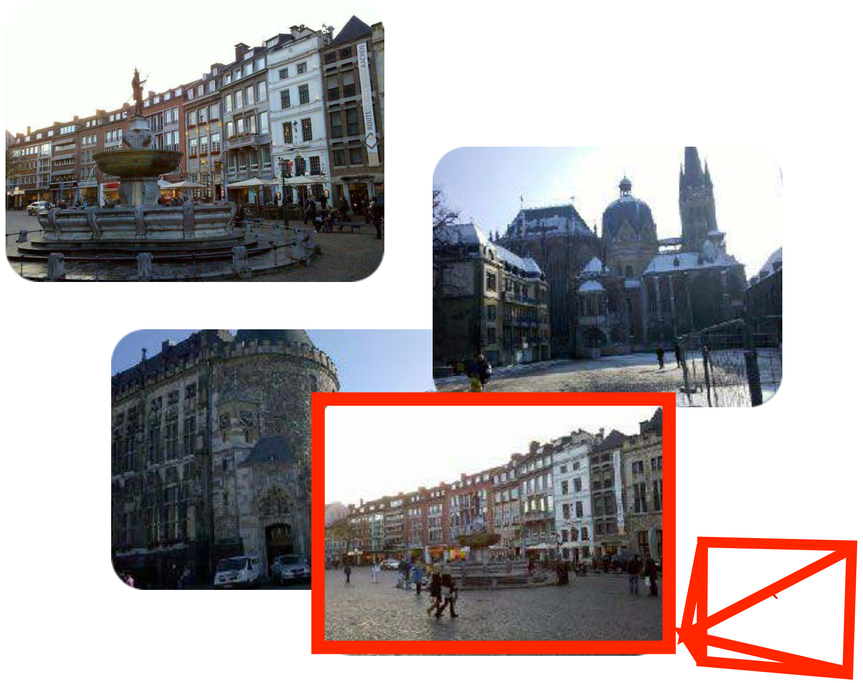
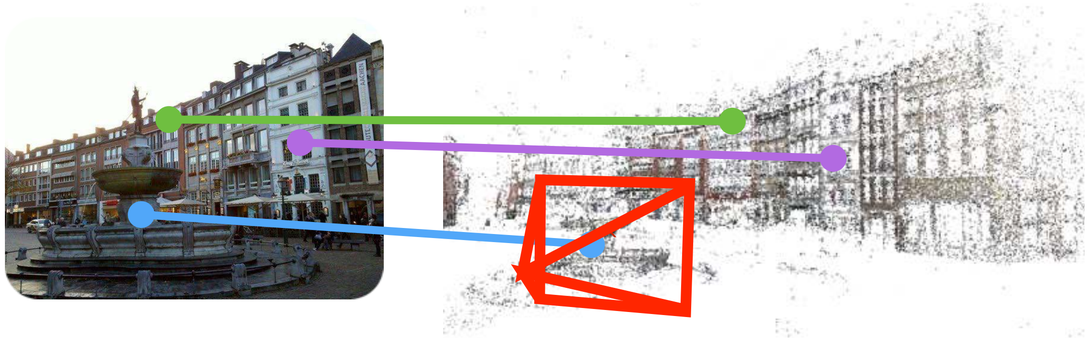
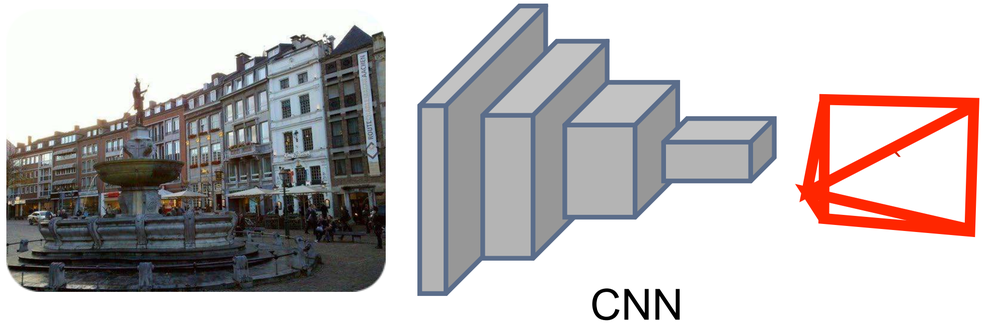


Image Retrieval



Structure-based



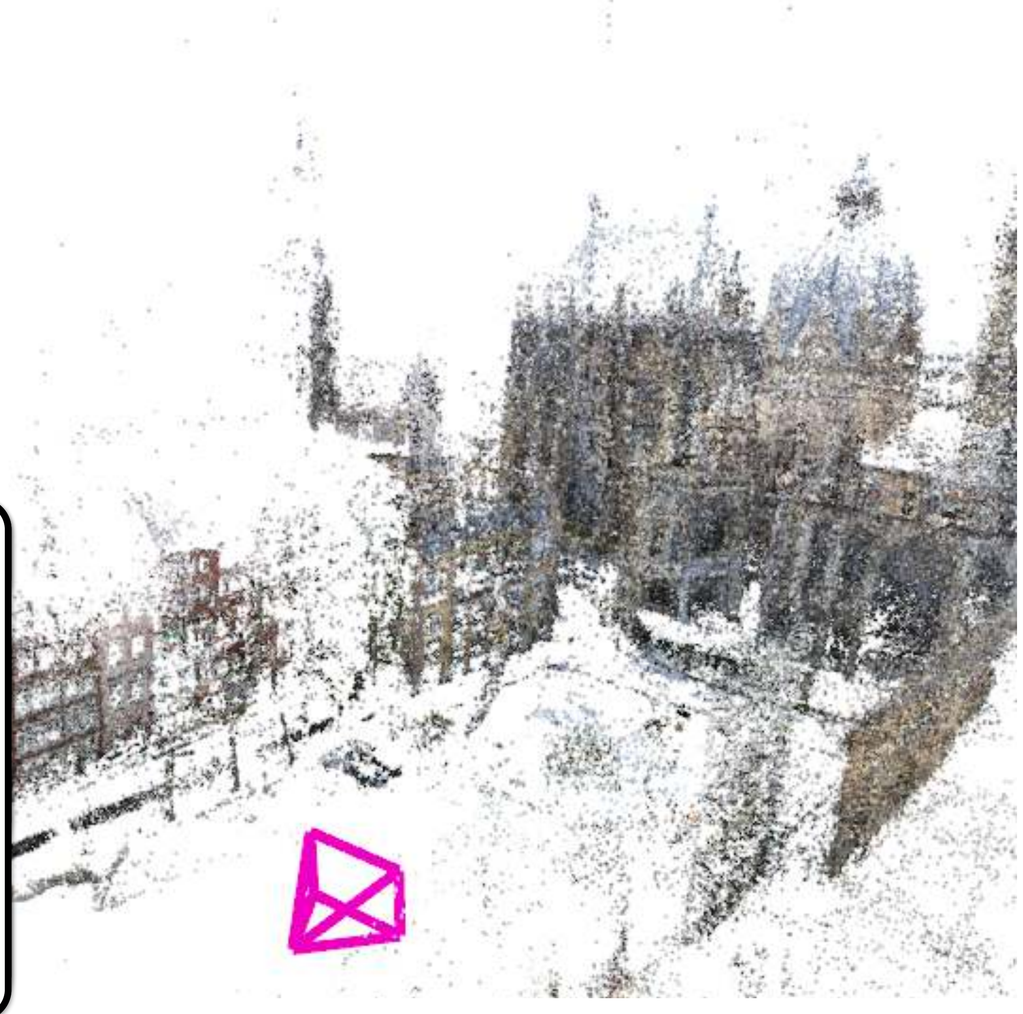
Pose Regression

“Old School” Localization

Feature Detection	1999 (SIFT)
Feature Description	
Descriptor Matching for 2D-3D Matching	1975 (kd-trees)
Estimate Camera Pose (RANSAC + n-point-pose solver)	1841 (P3P), 1981 (RANSAC)

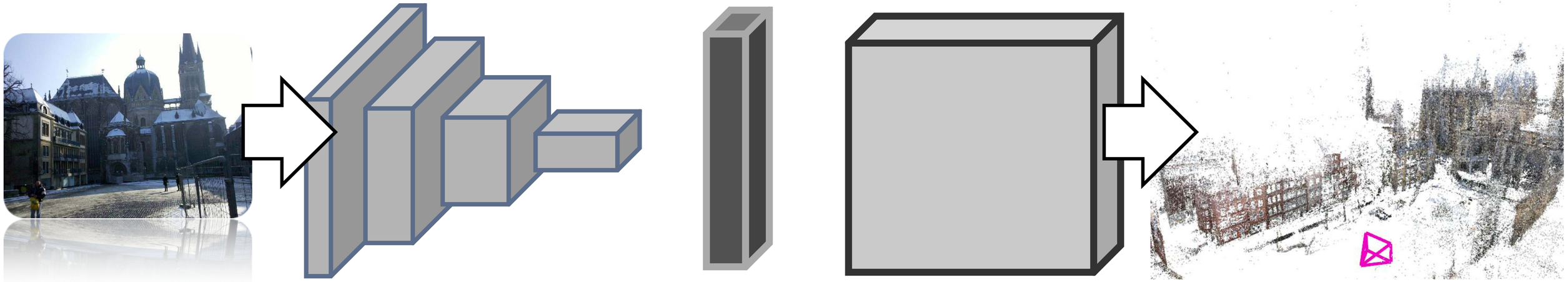
“New School” Localization

[Kendall, Grimes, Cipolla, PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization, ICCV 2015]

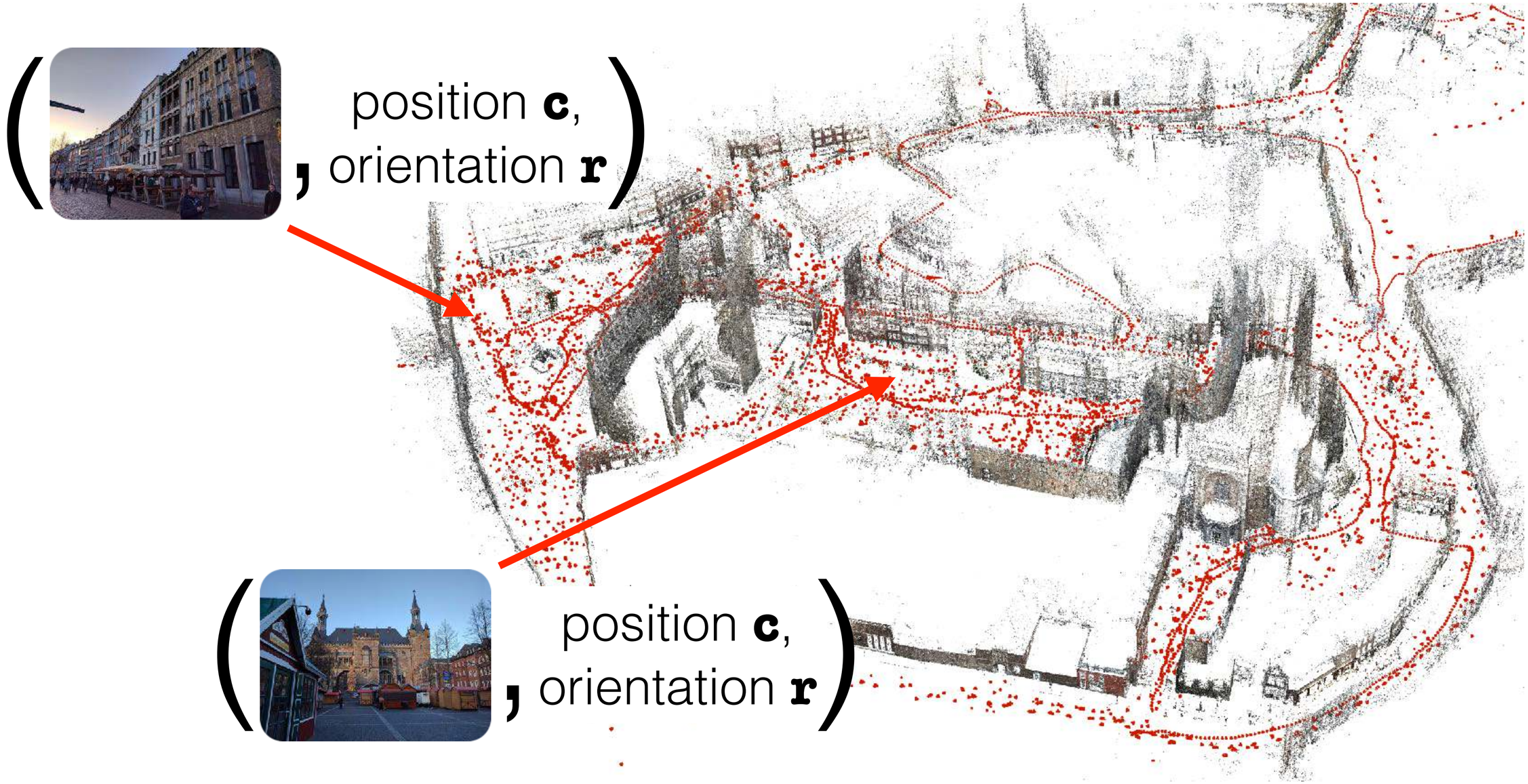


Convolutional Neural Network

Camera Pose Regression



Training Data



Progress Since 2015

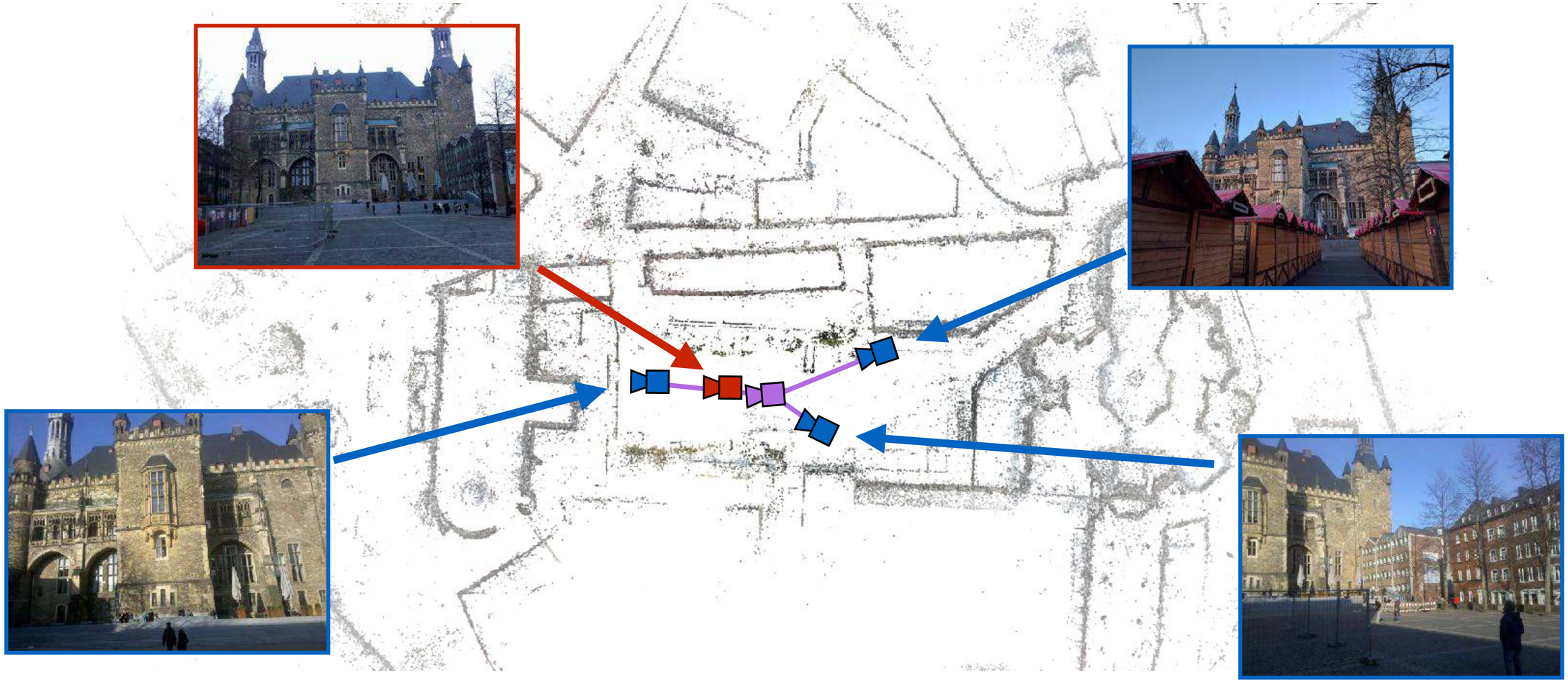
	Cambridge Landmarks					7 Scenes						
	Kings	Old	Shop	St. Mary's	Street	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs
PoseNet (PN) [30]	1.92/5.40	2.31/5.38	1.46/8.08	2.65/8.48		0.32/8.12	0.47/14.4	0.29/12.0	0.48/7.68	0.47/8.42	0.59/8.64	0.47/13.8
PN learned weights [29]	0.99/1.06	2.17/2.94	1.05/3.97	1.49/3.43	20.7/25.7	0.14/4.50	0.27/11.8	0.18/12.1	0.20/5.77	0.25/4.82	0.24/5.52	0.37/10.6
Bay. PN [28]	1.74/4.06	2.57/5.14	1.25/7.54	2.11/8.38		0.37/7.24	0.43/13.7	0.31/12.0	0.48/8.04	0.61/7.08	0.58/7.54	0.48/13.1
geo. PN [29]	0.88/1.04	3.20/3.29	0.88/3.78	1.57/3.32	20.3/25.5	0.13/4.48	0.27/11.3	0.17/13.0	0.19/5.55	0.26/4.75	0.23/5.35	0.35/12.4
LSTM PN [76]	0.99/3.65	1.51/4.29	1.18/7.44	1.52/6.68		0.24/5.77	0.34/11.9	0.21/13.7	0.30/8.08	0.33/7.00	0.37/8.83	0.40/13.7
GPoseNet [12]	1.61/2.29	2.62/3.89	1.14/5.73	2.93/6.46		0.20/7.11	0.38/12.3	0.21/13.8	0.28/ 8.83	0.37/6.94	0.35/8.15	0.37/12.5
SVS-Pose [50]	1.06/2.81	1.50/4.03	0.63/5.73	2.11/8.11								
Hourglass PN [44]						0.15/6.17	0.27/10.8	0.19/11.6	0.21/8.48	0.25/7.01	0.27/10.2	0.29/12.5
BranchNet [78]						0.18/5.17	0.34/8.99	0.20/14.2	0.30/7.05	0.27/5.10	0.33/7.40	0.38/10.3
MapNet [11]	1.07/1.89	1.94/3.91	1.49/4.22	2.00/4.53		0.08/3.25	0.27/11.7	0.18/13.3	0.17/5.15	0.22/4.02	0.23/4.93	0.30/12.1
MapNet+ [11]						0.10/3.17	0.20/9.04	0.13/11.1	0.18/5.38	0.19/3.92	0.20/5.01	0.30/13.4
MapNet+PGO [11]						0.09/3.24	0.20/9.29	0.12/8.45	0.19/5.42	0.19/3.96	0.20/4.94	0.27/10.6

Is this good or bad? **Baseline** needed!

[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Image Retrieval as a Baseline

- **Approximate** pose of **test** image via **training** image poses



[Torii, Sivic, Pajdla, Visual localization by linear combination of image descriptors, ICCV Workshops 2011]

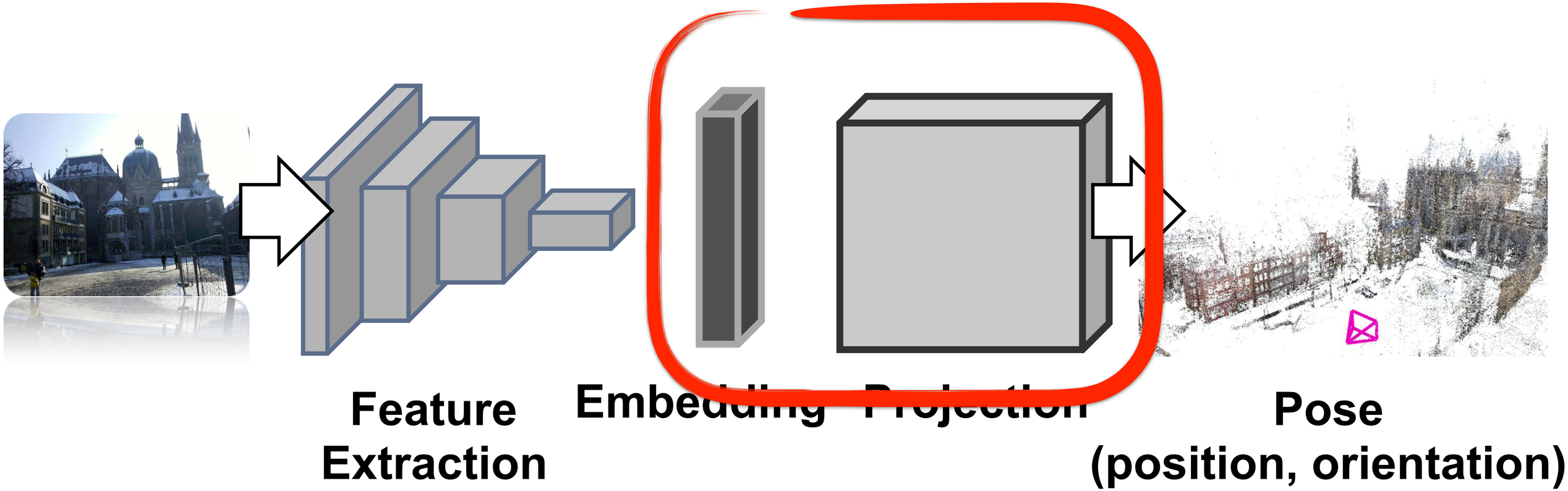
Pose Regression vs. Image Retrieval

	Cambridge Landmarks					7 Scenes						
	Kings	Old	Shop	St. Mary's	Street	Chess	Fire	Heads	Office	Pumpkin	Kitchen	Stairs
PoseNet (PN) [30]	1.92/5.40	2.31/5.38	1.46/8.08	2.65/8.48		0.32/8.12	0.47/14.4	0.29/12.0	0.48/7.68	0.47/8.42	0.59/8.64	0.47/13.8
PN learned weights [29]	0.99/1.06	2.17/2.94	1.05/3.97	1.49/3.43	20.7/25.7	0.14/4.50	0.27/11.8	0.18/12.1	0.20/5.77	0.25/4.82	0.24/5.52	0.37/10.6
Bay. PN [28]	1.74/4.06	2.57/5.14	1.25/7.54	2.11/8.38		0.37/7.24	0.43/13.7	0.31/12.0	0.48/8.04	0.61/7.08	0.58/7.54	0.48/13.1
geo. PN [29]	0.88/1.04	3.20/3.29	0.88/3.78	1.57/3.32	20.3/25.5	0.13/4.48	0.27/11.3	0.17/13.0	0.19/5.55	0.26/4.75	0.23/5.35	0.35/12.4
LSTM PN [76]	0.99/3.65	1.51/4.29	1.18/7.44	1.52/6.68		0.24/5.77	0.34/11.9	0.21/13.7	0.30/8.08	0.33/7.00	0.37/8.83	0.40/13.7
GPoseNet [12]	1.61/2.29	2.62/3.89	1.14/5.73	2.93/6.46		0.20/7.11	0.38/12.3	0.21/13.8	0.28/ 8.83	0.37/6.94	0.35/8.15	0.37/12.5
SVS-Pose [50]	1.06/2.81	1.50/4.03	0.63/5.73	2.11/8.11								
Hourglass PN [44]						0.15/6.17	0.27/10.8	0.19/11.6	0.21/8.48	0.25/7.01	0.27/10.2	0.29/12.5
BranchNet [78]						0.18/5.17	0.34/8.99	0.20/14.2	0.30/7.05	0.27/5.10	0.33/7.40	0.38/10.3
MapNet [11]	1.07/1.89	1.94/3.91	1.49/4.22	2.00/4.53		0.08/3.25	0.27/11.7	0.18/13.3	0.17/5.15	0.22/4.02	0.23/4.93	0.30/12.1
MapNet+ [11]						0.10/3.17	0.20/9.04	0.13/11.1	0.18/5.38	0.19/3.92	0.20/5.01	0.30/13.4
MapNet+PGO [11]						0.09/3.24	0.20/9.29	0.12/8.45	0.19/5.42	0.19/3.96	0.20/4.94	0.27/10.6
DenseVLAD [71]	2.80/5.72	4.01/7.13	1.11/7.61	2.31/8.00	5.16/23.5	0.21/12.5	0.33/13.8	0.15/14.9	0.28/11.2	0.31/11.3	0.30/12.3	0.25/15.8
DenseVLAD + Inter.	1.48/4.45	2.68/4.63	0.90/4.32	1.62/6.06	15.4/25.7	0.18/10.0	0.33/12.4	0.14/14.3	0.25/10.1	0.26/9.42	0.27/11.1	0.24/14.7

Not better than image retrieval!

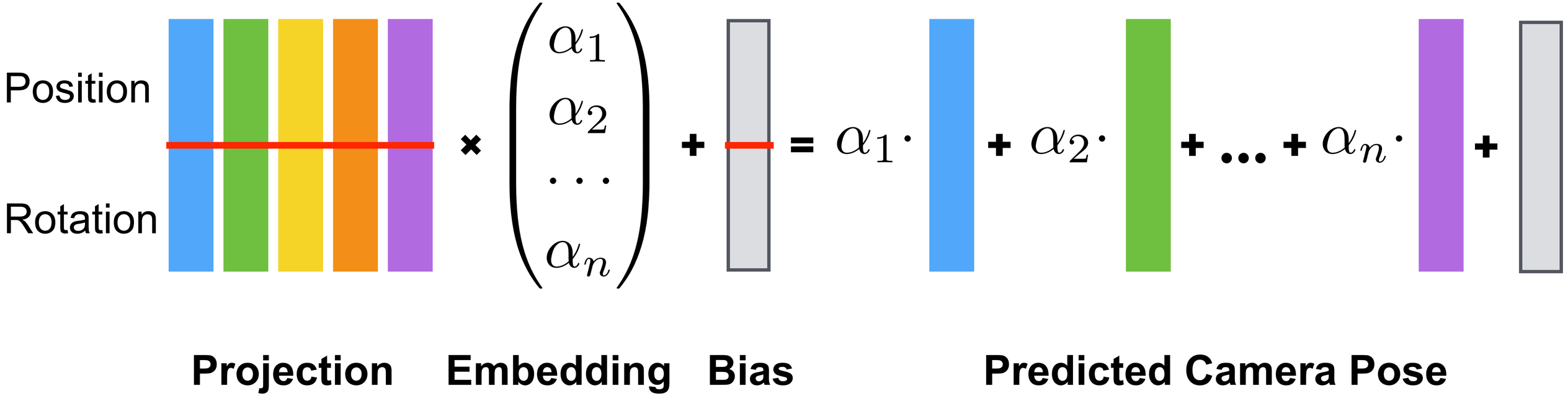
[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Camera Pose Regression



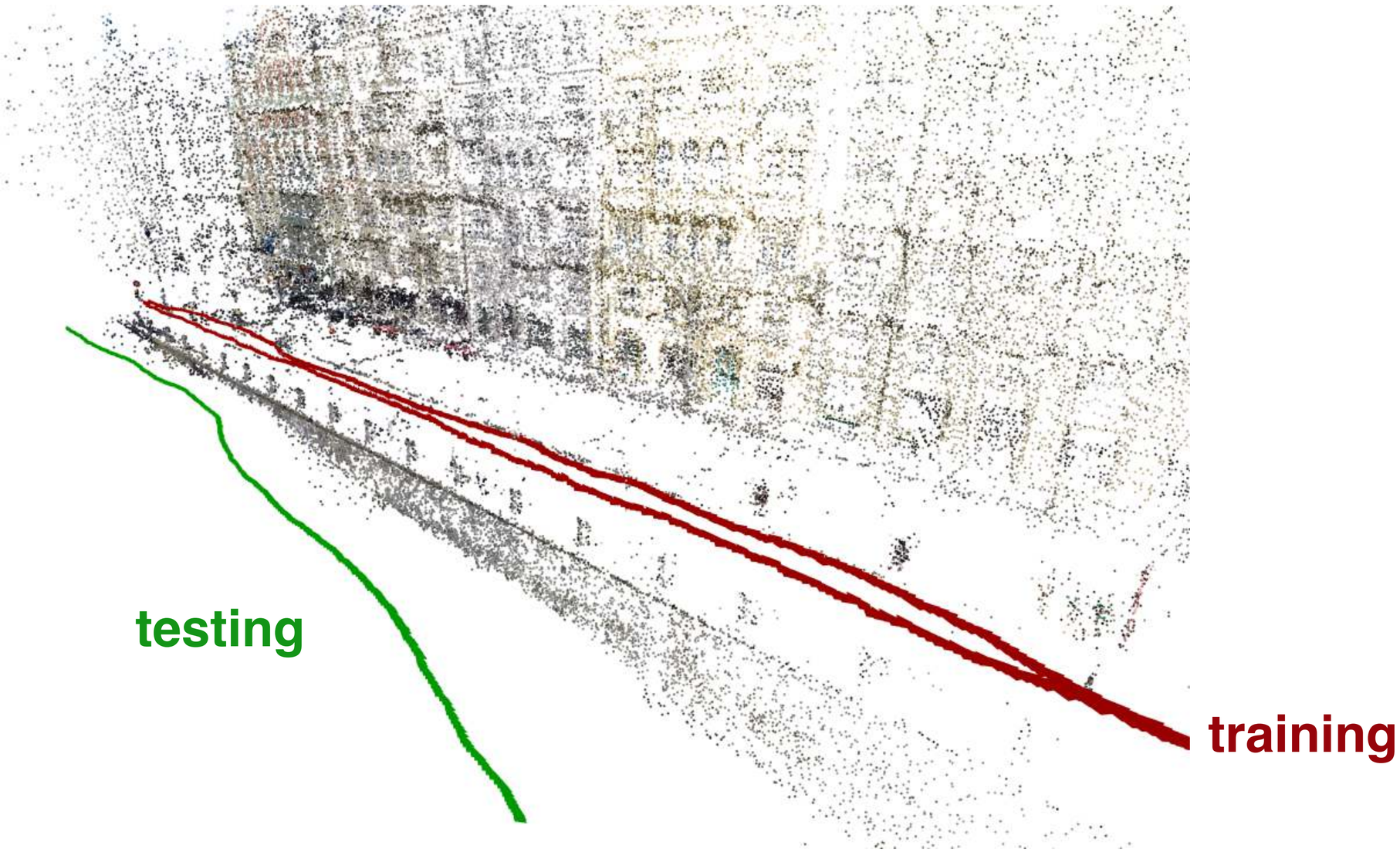
Looking Inside The Black Box

- Pose regression in **last FC layer** as **linear combination of base poses**:



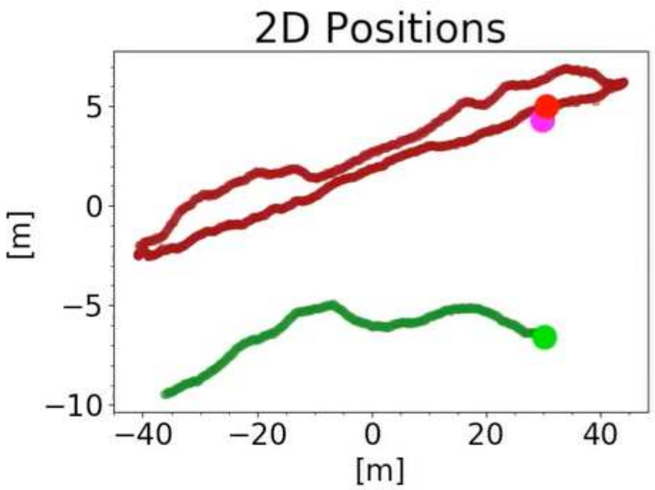
[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Camera Pose Regression Example



[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Camera Pose Regression Example



GT training poses

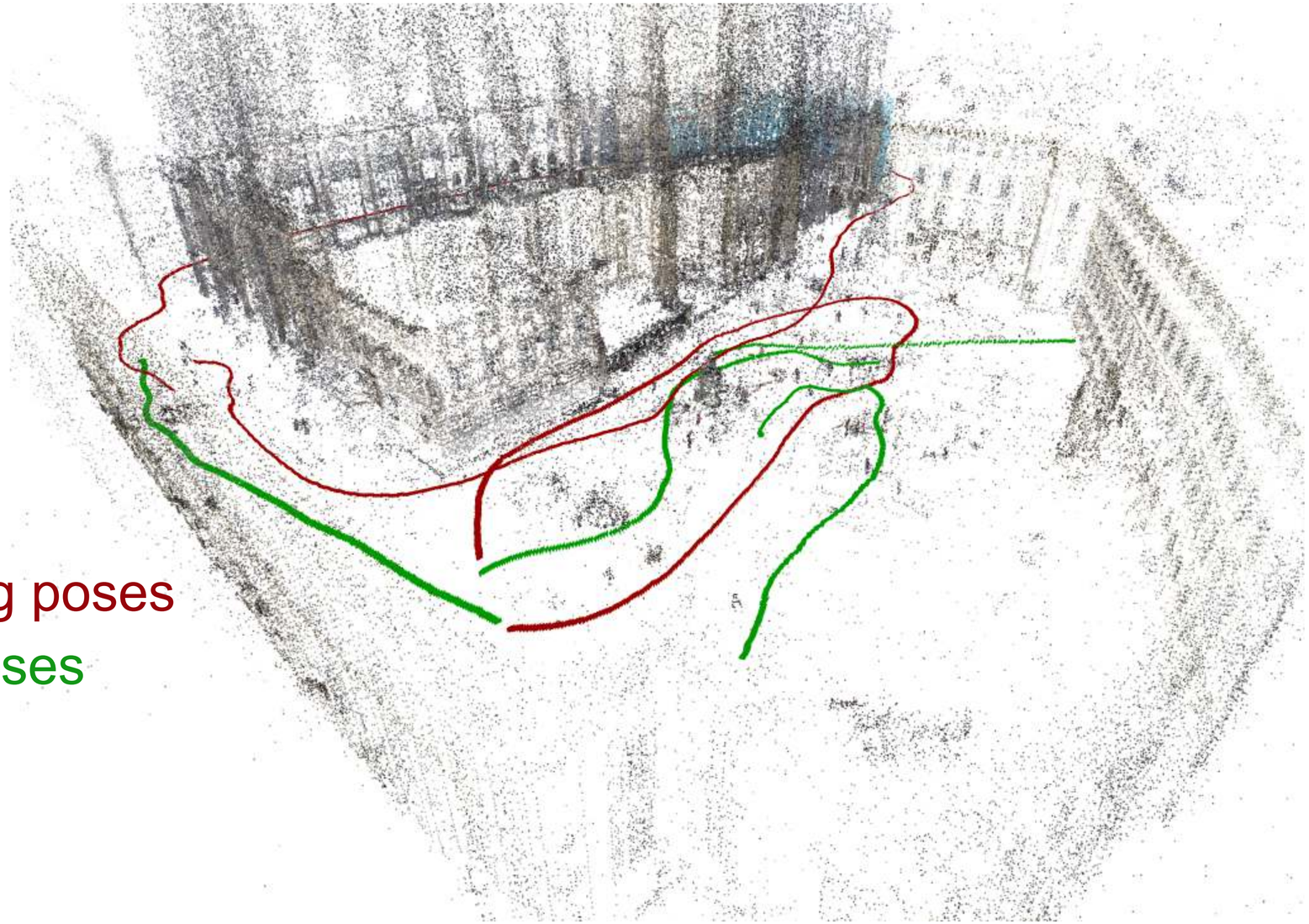
GT test poses

Predicted test pose

Pose of most similar training image

[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Camera Pose Regression Example

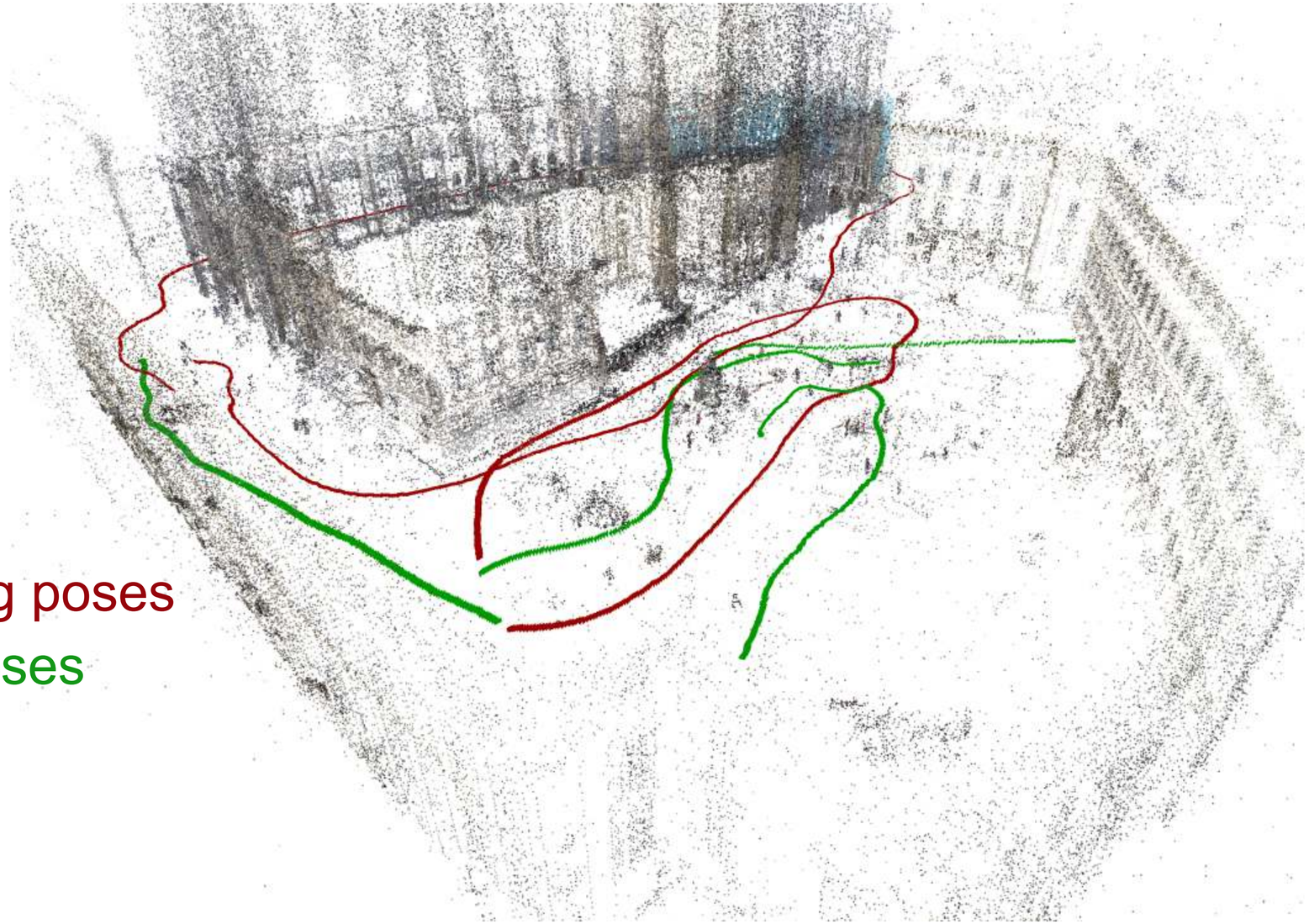


GT training poses

GT test poses

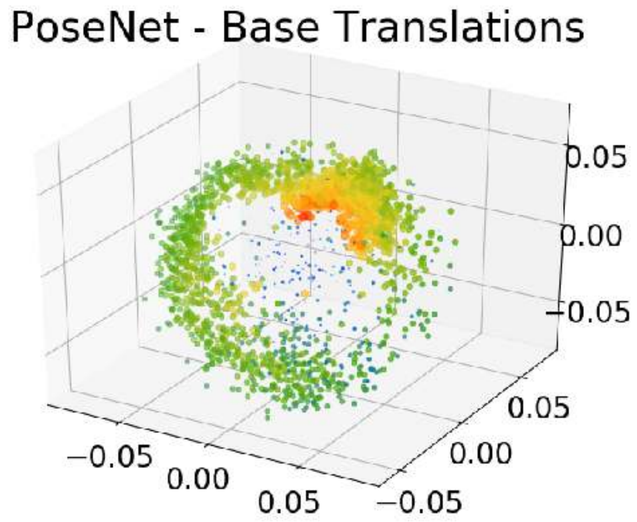
[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

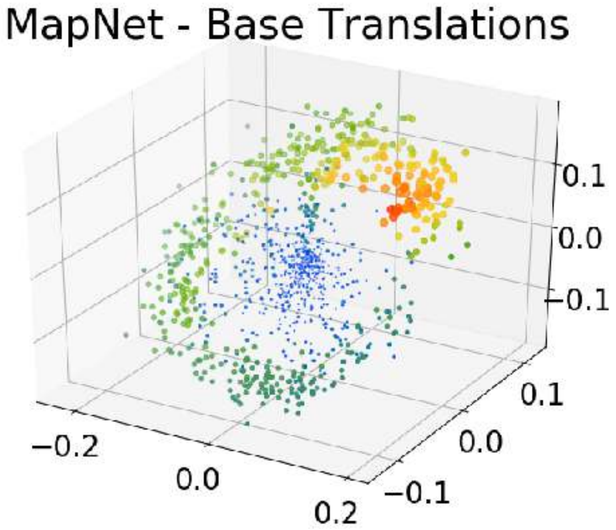
Camera Pose Regression Example



GT training poses

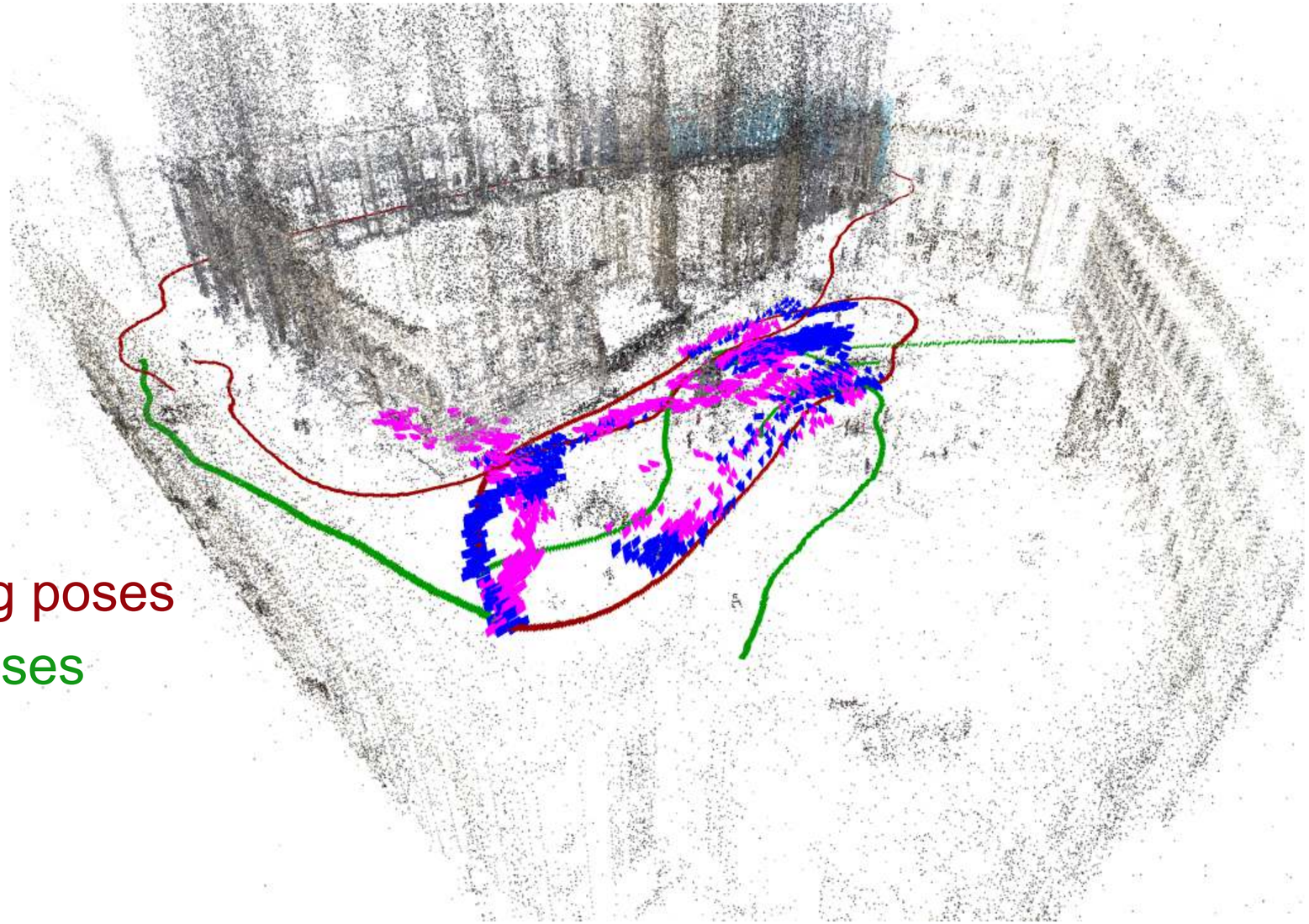
GT test poses





[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Camera Pose Regression Example

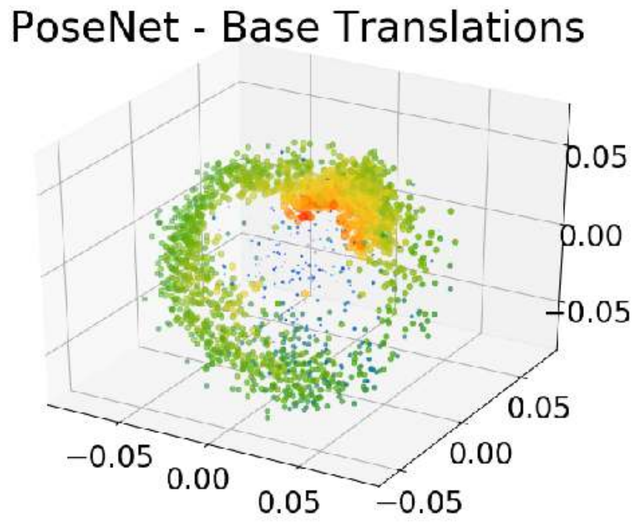


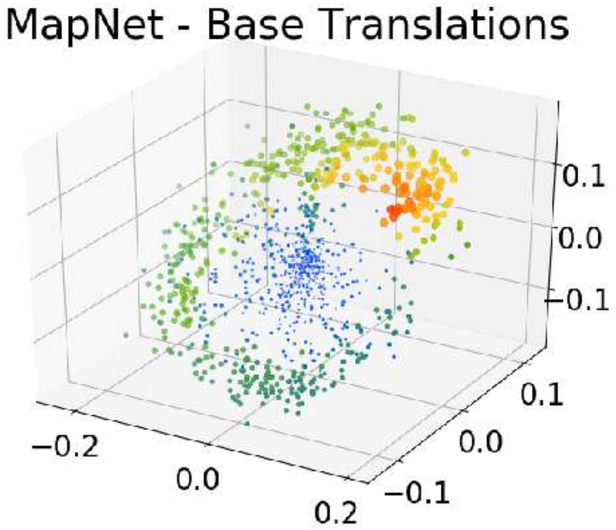
GT training poses

GT test poses

MapNet

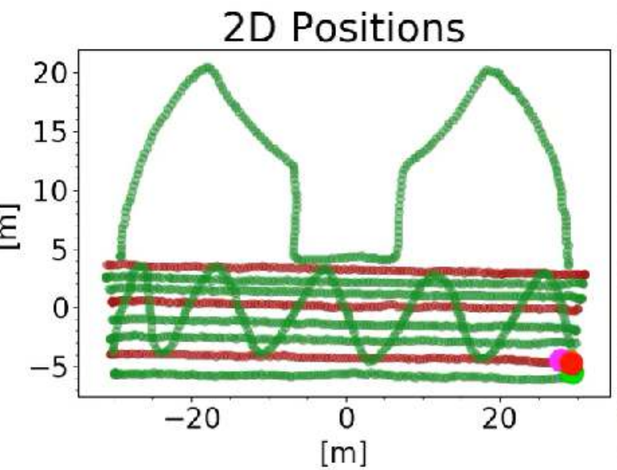
PoseNet

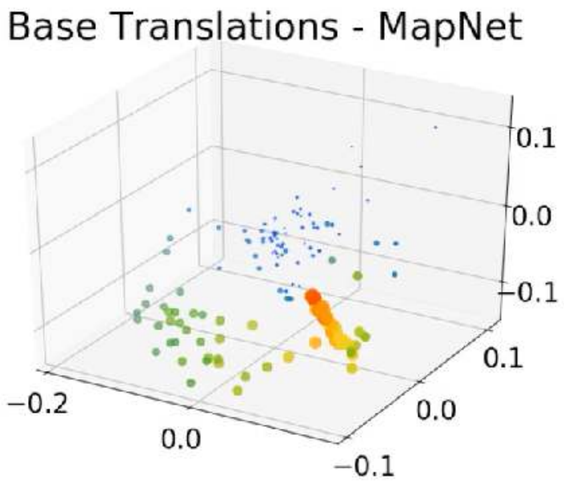




[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Camera Pose Regression Example





GT training poses

GT test poses

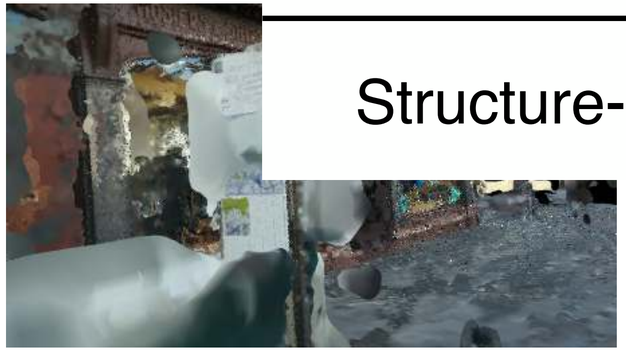
Predicted test pose

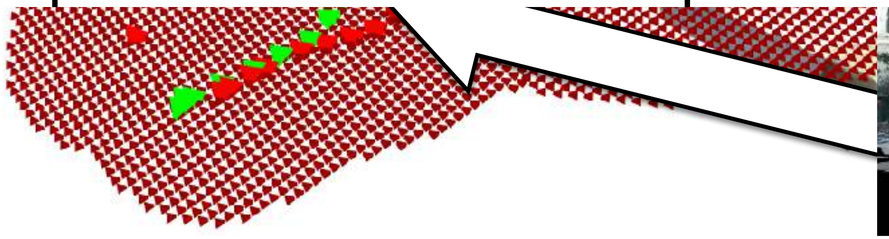
Pose of most similar training image

[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Training from More Data

	203 Training images	9,412 Training images
MapNet	1.07m / 4.70deg	0.33m / 1.46deg
Image Retrieval	0.89m / 5.71deg	0.38m / 6.41deg
Structure-Based	0.01m / 0.04deg	



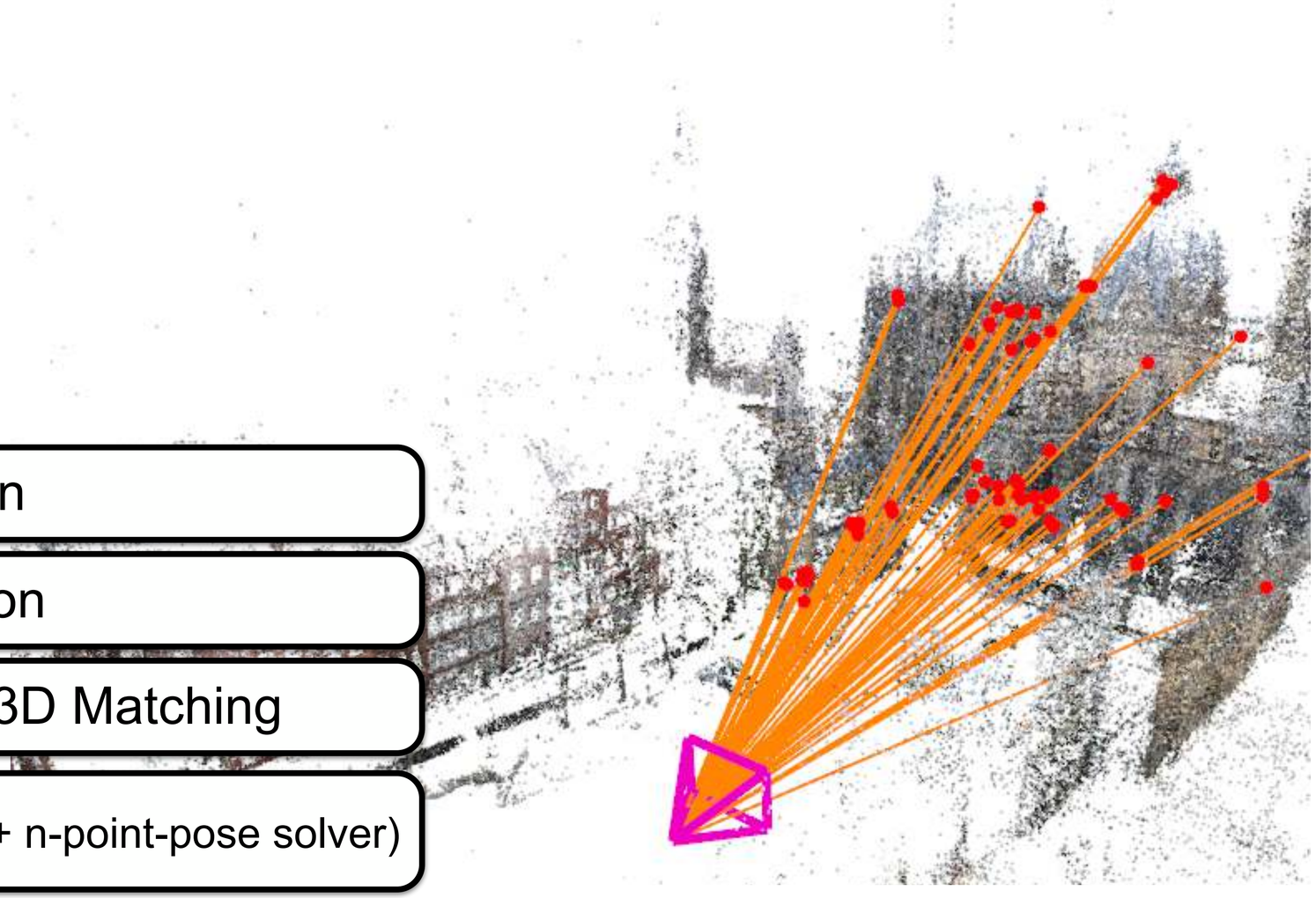


[Sattler, Zhou, Pollefeys, Leal-Taixé, Understanding the Limitations of CNN-based Absolute Camera Pose Regression, CVPR 2019]

Takeaways

- **Geometry matters**: Learning camera pose regression is hard, problems to generalize beyond training data
- Much **closer in performance to image retrieval than structure-based methods**
- **The need for baselines**: Don't exclude baselines just because they are simple
- **More research needed**: Pose regression algorithms do not work as well as I would expect

Local Feature-based Localization



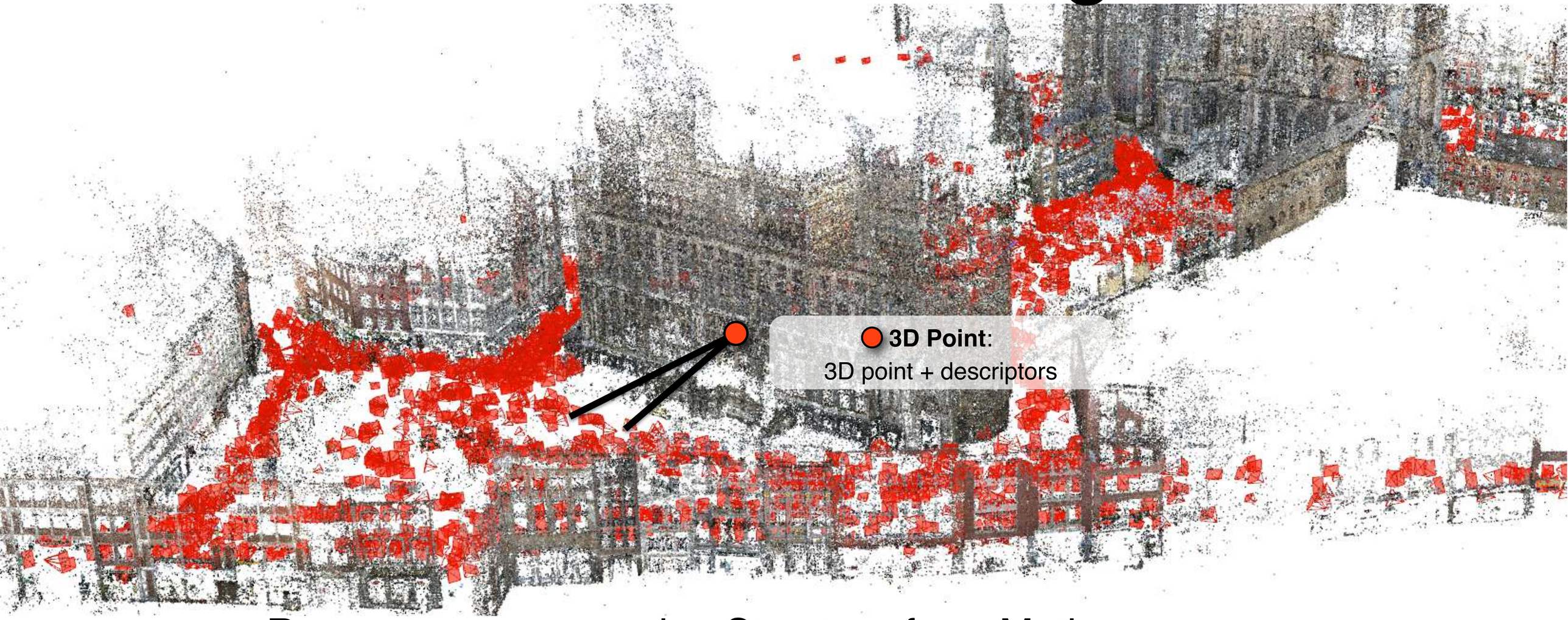
Feature Detection

Feature Description

Descriptor Matching for 2D-3D Matching

Estimate Camera Pose (RANSAC + n-point-pose solver)

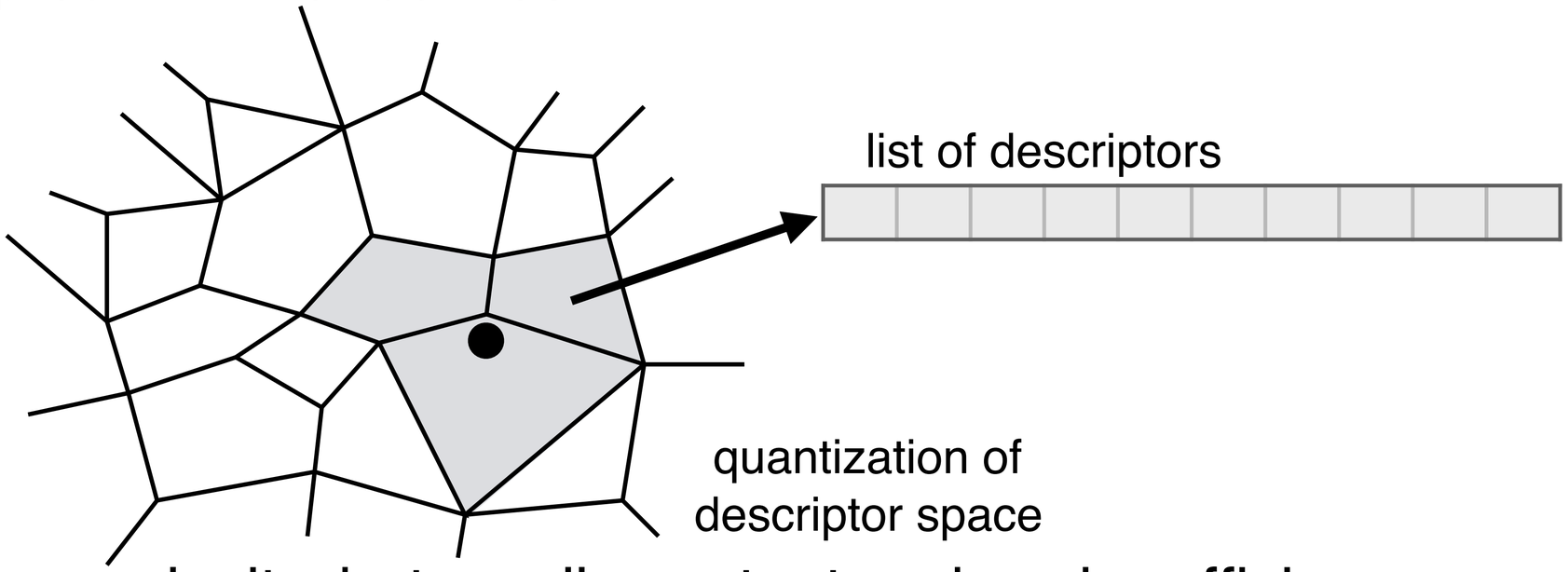
2D-3D Matching



- Reconstruct scene using Structure-from-Motion
- Associate each 3D point with local image descriptors (SIFT)

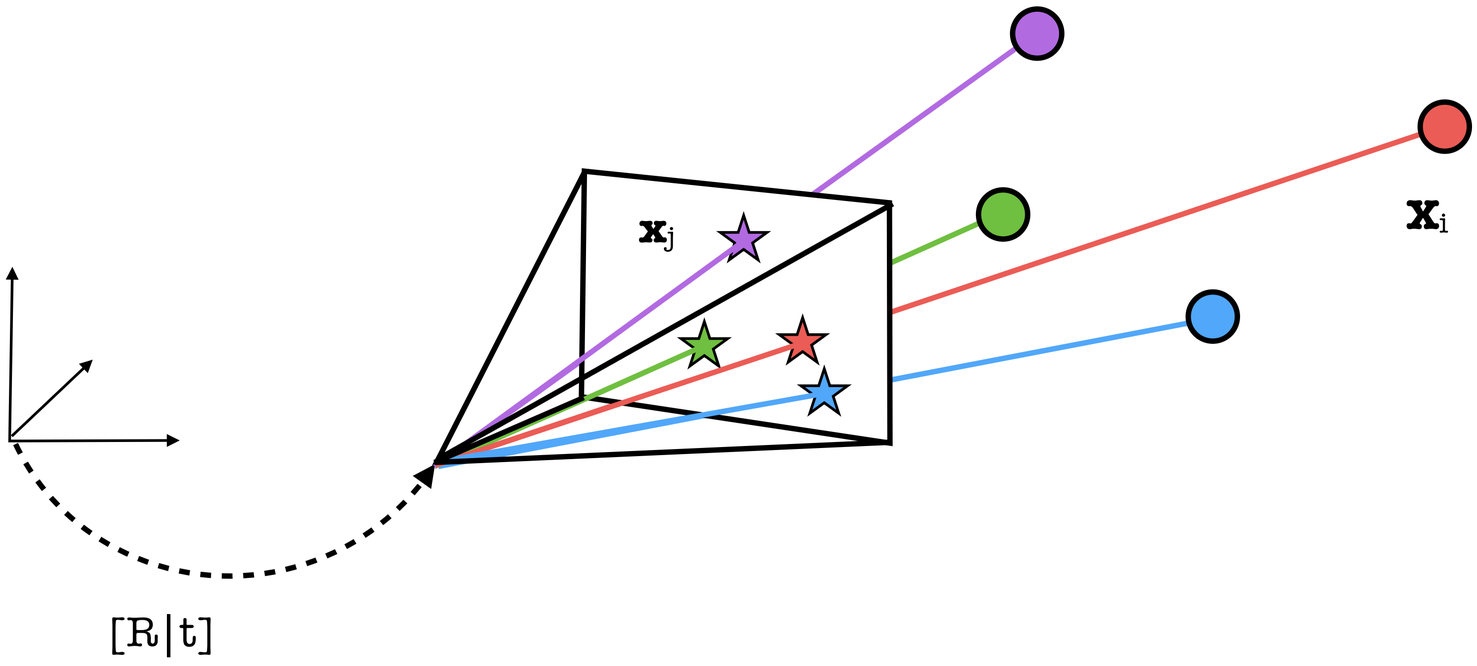
Nearest Neighbor Search

- 2D-3D matching via nearest neighbor search in descriptor space
- Approximate search due to curse of dimensionality
- Popular approach: Inverted files



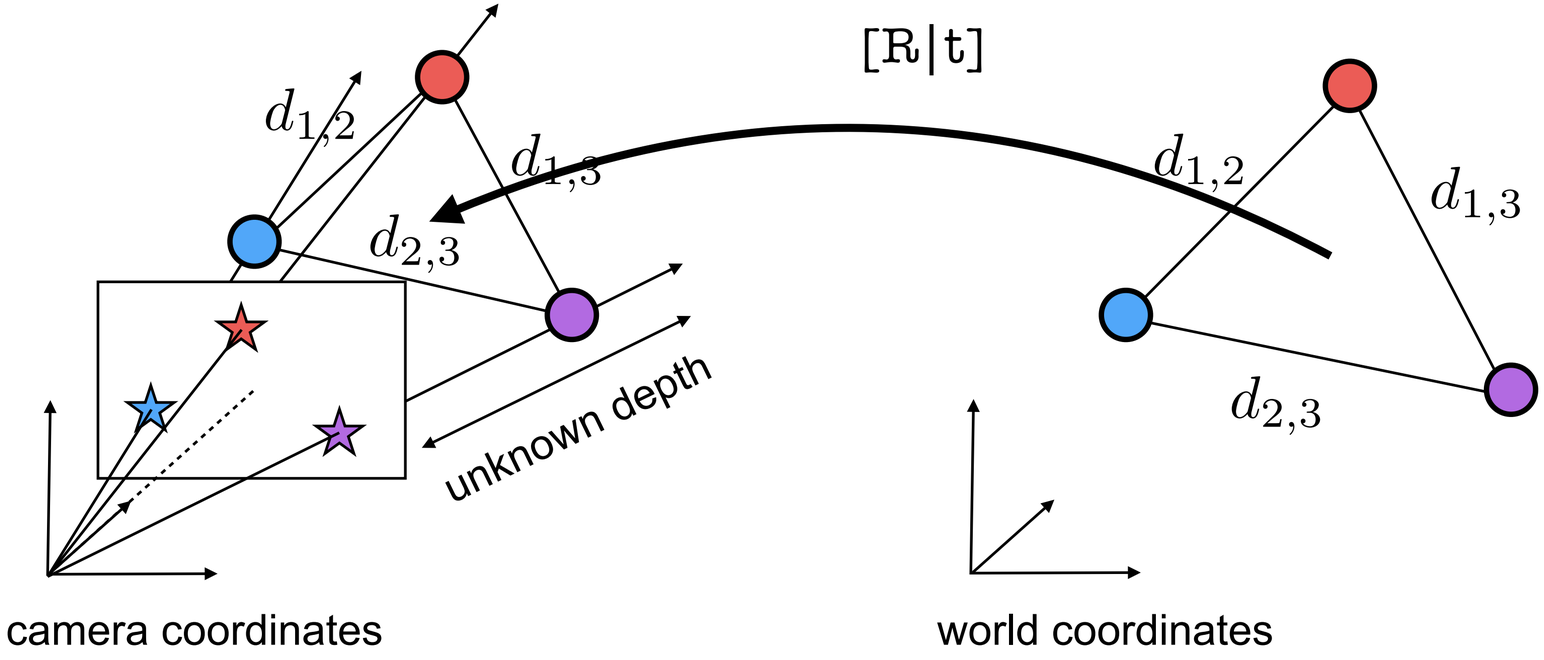
- Linear time complexity, but small constant and cache efficiency
- Very good software libraries:
  - FLANN [Muja, Lowe, Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration, VISAPP 2009] [code]
  - FAISS [Johnson, Douze, Jégou, Billion-scale similarity search with GPUs. arXiv:1702.08734] [code]

The n-Point Pose (PnP) Problem



- Given: *n* 2D-3D correspondences $(\mathbf{x}_i, \mathbf{X}_i)$
- Compute pose $[\mathtt{R}|\mathtt{t}]$ s.t. $\mathtt{K}[\mathtt{R}|\mathtt{t}]\mathbf{X}_i = \alpha_i\mathbf{x}_i$, $\alpha_i>0$
- Optionally: Also estimate internal calibration matrix $\mathtt{K}$, e.g., [Larsson, Kukelova, Zheng, Making minimal solvers for absolute pose estimation compact and robust, ICCV 2017][Bujnak, Kukelova, Pajdla, A general solution to the P4P problem for camera with unknown focal length, CVPR 2008][Larsson, Sattler, Kukelova, Pollefeys, Revisiting Radial Distortion Absolute Pose, ICCV 2019](code)

Calibrated Case: P3P



- Very efficient: Univariate 4th degree polynomial, ~2µs total [Kneip et al., CVPR'11] [code]
- Up to four solutions: Disambiguate using 4th point
- Problem older than computer vision: [Grunert, Das Pothenotische Problem in erweiterter Gestalt nest seine Anwendungen in der Geodäisie. **1841**]

Robust Estimation via RANSAC

While probability of missing correct model >η

Estimate model from *n* random data points

Estimate support (**#inliers** / **robust cost func.**) of model

If new best model

Perform Local Optimization (LO)

update best model, η

Return: Model with most inliers / lowest cost

[Chum, Matas, Optimal Randomized RANSAC. PAMI 2008]

[Lebeda, Matas, Chum, Fixing the Locally Optimized RANSAC. BMVC 2012] [code]

- See also USAC [Raguram et al., PAMI'13] [code] (good overview, nice implementation)
- Don't use standard RANSAC!

[Fischler & Bolles, Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. CACM 1981]

A Simple Localization System

Query Image

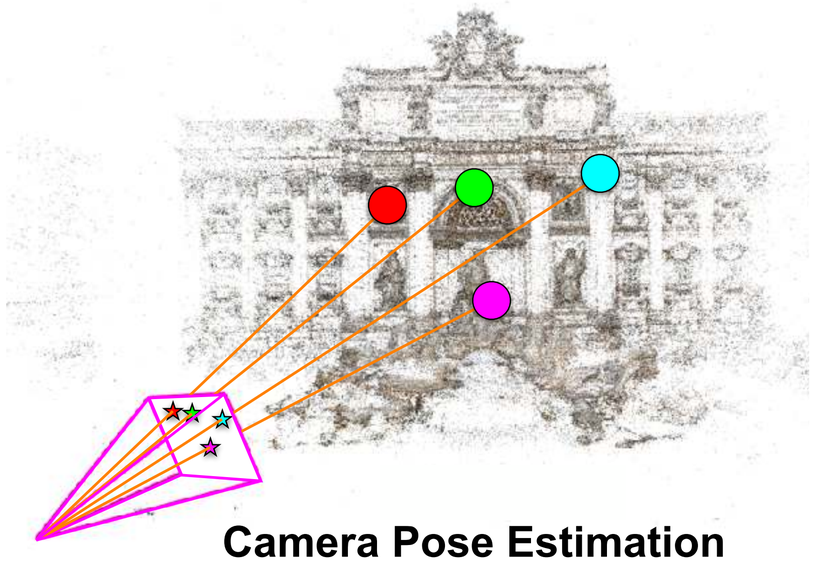
photo credit: Ming Li

kd-tree

2-nn search & Lowe's ratio test

$$\frac{||d - d_1||_2}{||d - d_2||_2} < 0.7$$

[Lowe, Distinctive image features from scale-invariant keypoints. IJCV 2004]



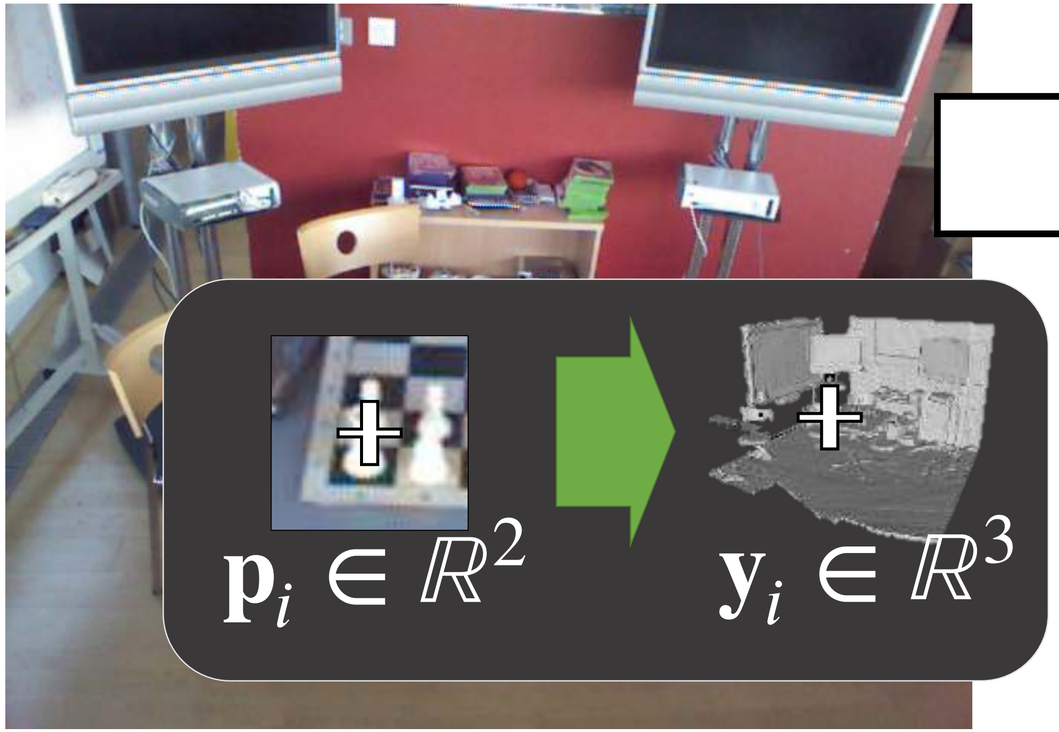
Camera Pose Estimation
LO-RANSAC +$\mathrm{T}_{1,1}$ + P6P

pose valid if ≥ 12 inliers

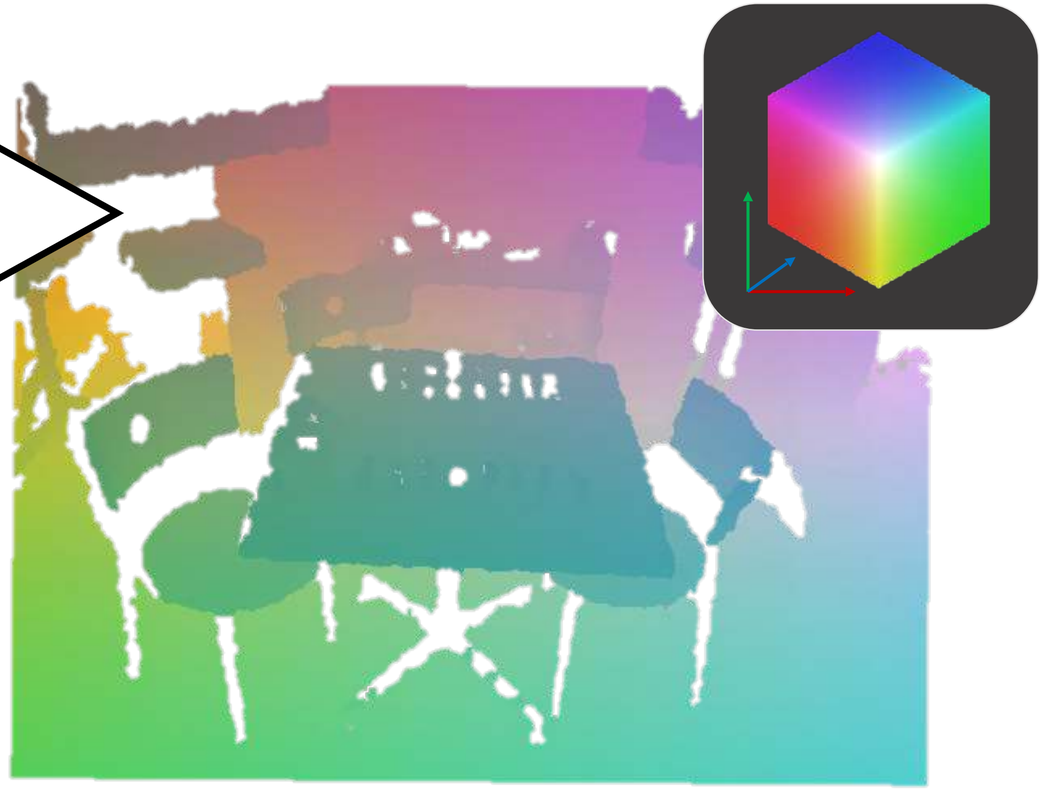
[Sattler, Leibe, Kobbelt, Fast Image-Based Localization using Direct 2D-to-3D Matching. ICCV 2011] [code]

Scene Coordinate Regression

Image → Feature Extraction → 2D-3D Matching → RANSAC → 6DOF Pose



Image

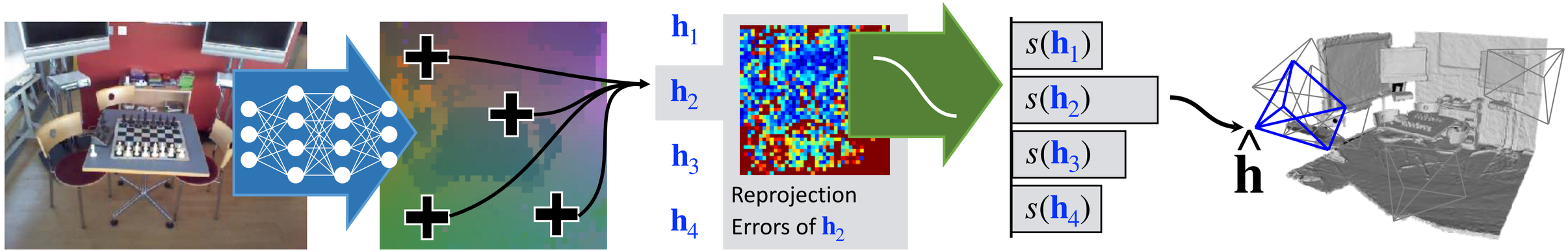


3D points in global coordinates

[Shotton et al., Scene Coordinate Regression Forests for Camera Relocalization in RGB-D Images, CVPR 2013]

slide credit: Eric Brachmann

Differentiable RANSAC (DSAC)



Input RGB | Scene Coordinate Regression (w) | Hypothesis Sampling | Soft Scoring | Hypothesis Selection | Result

argmax Selection	Soft argmax Selection	**Probabilistic Selection**
$\mathbf{h}_{\mathrm{AM}} = \underset{\mathbf{h}_j}{\operatorname{argmax}}\, s(\mathbf{h}_j)$	$\mathbf{h}_{\mathrm{SoftAM}} = \sum_j \frac{\exp(s(\mathbf{h}_j))\mathbf{h}_j}{\sum_k \exp(s(\mathbf{h}_k))}$	$\mathbf{h}_{\mathrm{DSAC}} = \mathbf{h}_j,\ \text{where } j \sim \frac{\exp(s(\mathbf{h}_j))}{\sum_k \exp(s(\mathbf{h}_k))}$
hard decision	soft decision	hard decision
non-differentiable	differentiable	differentiable

slide credit: Eric Brachmann

[Brachmann, Krull Nowozin, Shotton, Michel, Gumhold, Rother, DSAC - Differentiable RANSAC for Camera Localization, CVPR 2017]
[Brachmann, Rother, Learning Less is More - 6D Camera Localization via 3D Surface Regression, CVPR 2018]

Comparison

Measure: Cambridge: Median position [m] / orientation [deg] error
Aachen: % queries localized within pose error thresholds

	Great Court	King's College	Old Hospital	Shop Facade	St. Mary's Church
Active Search		42cm	44cm	12cm	19cm
Active Search [1,2] (v1.1 using RANSACLib)	**32.6**cm	13.8cm	19.5cm	**4.9**cm	**8.7**cm
CNN-based Localization [3,4,5]	33.0cm	**12.6**cm	19.6cm	5.4cm	9.8cm

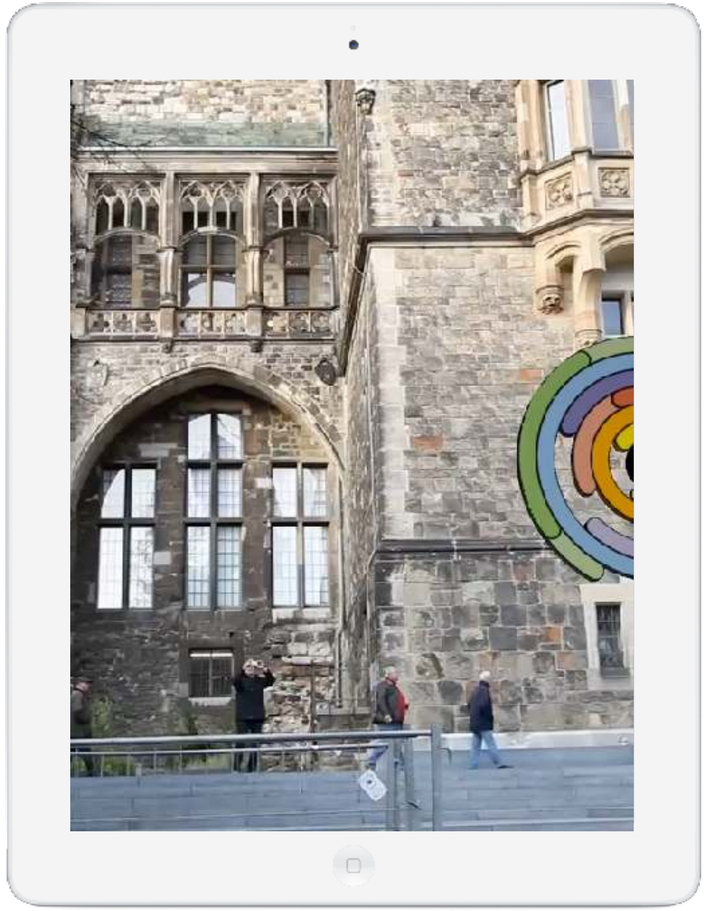
Cambridge Landmarks [6]
(outdoors)

[1] [Sattler, Leibe, Kobbelt, Improving Image-Based Localization by Active Correspondence Search, ECCV 2012]
[2] [Sattler, Leibe, Kobbelt, Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization, PAMI 2017]
[3] [Brachmann, Rother, Learning Less is More – 6D Camera Localization via 3D Surface Regression, CVPR 2018]
[4] [Brachmann, Rother, Neural-Guided RANSAC: Learning Where to Sample Model Hypotheses, ICCV 2019]
[5] [Brachmann, Rother, Expert Sample Consensus Applied to Camera Re-Localization, ICCV 2019]
[6] [Kendall et al., PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization, ICCV 2015]
[7] [Sattler, Weyand, Leibe, Kobbelt, Image Retrieval for Image-Based Localization Revisited, BMVC 2012]
[8] [Sattler, Maddern, Toft, Torii, Hammarstrand, Stenborg, Safari, Okutomi, Pollefeys, Sivic, Kahl, Pajdla, Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions, CVPR 2018]

Takeaways

- (SIFT) feature-based methods still competitive on easier datasets
- State-of-the-art on medium and large-scale datasets

Mobile Localization



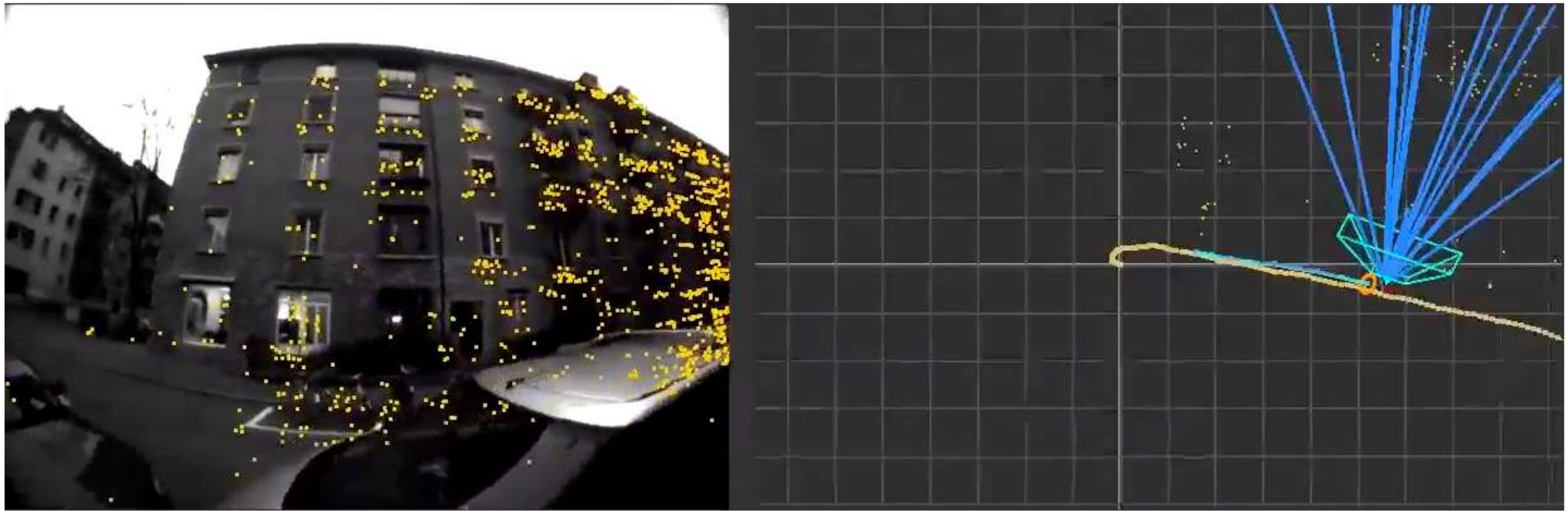
Challenges

- Hardware limitations
  - CPU/GPU
  - storage
  - energy consumption
- Real-time localization
- Augmented / Virtual Reality:
  - Temporal consistent pose estimates
  - No jitter from frame-to-frame

Video credit: Sven Middelberg [Middelberg, Sattler, Untzelmann, Kobbelt. Scalable 6-DOF Localization on Mobile Devices. ECCV 2014]

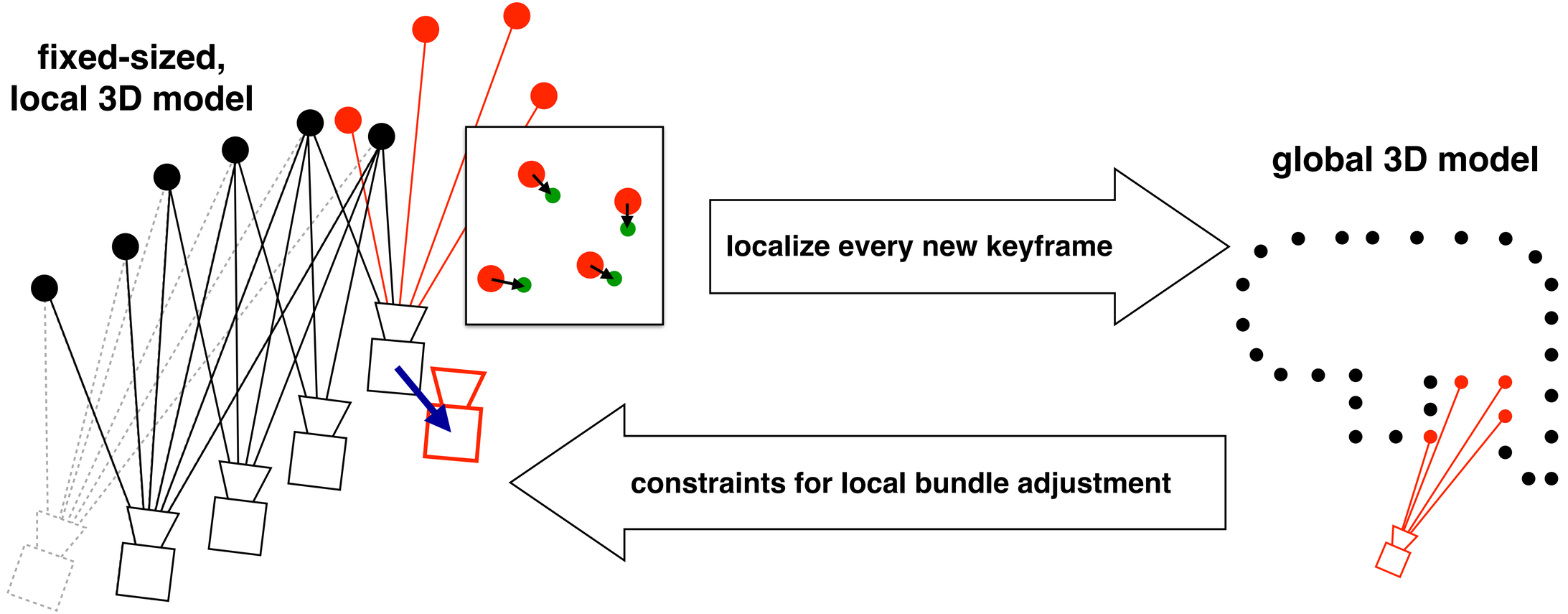
Mobile Localization

Key idea: Decouple real-time pose tracking & non-real-time global localization



[Lynen, Sattler, Bosse, Hesch, Pollefeys, Siegwart, Get Out of My Lab: Large-scale Real-Time Visual-Inertial Localization. RSS 2015]

Algorithm Overview

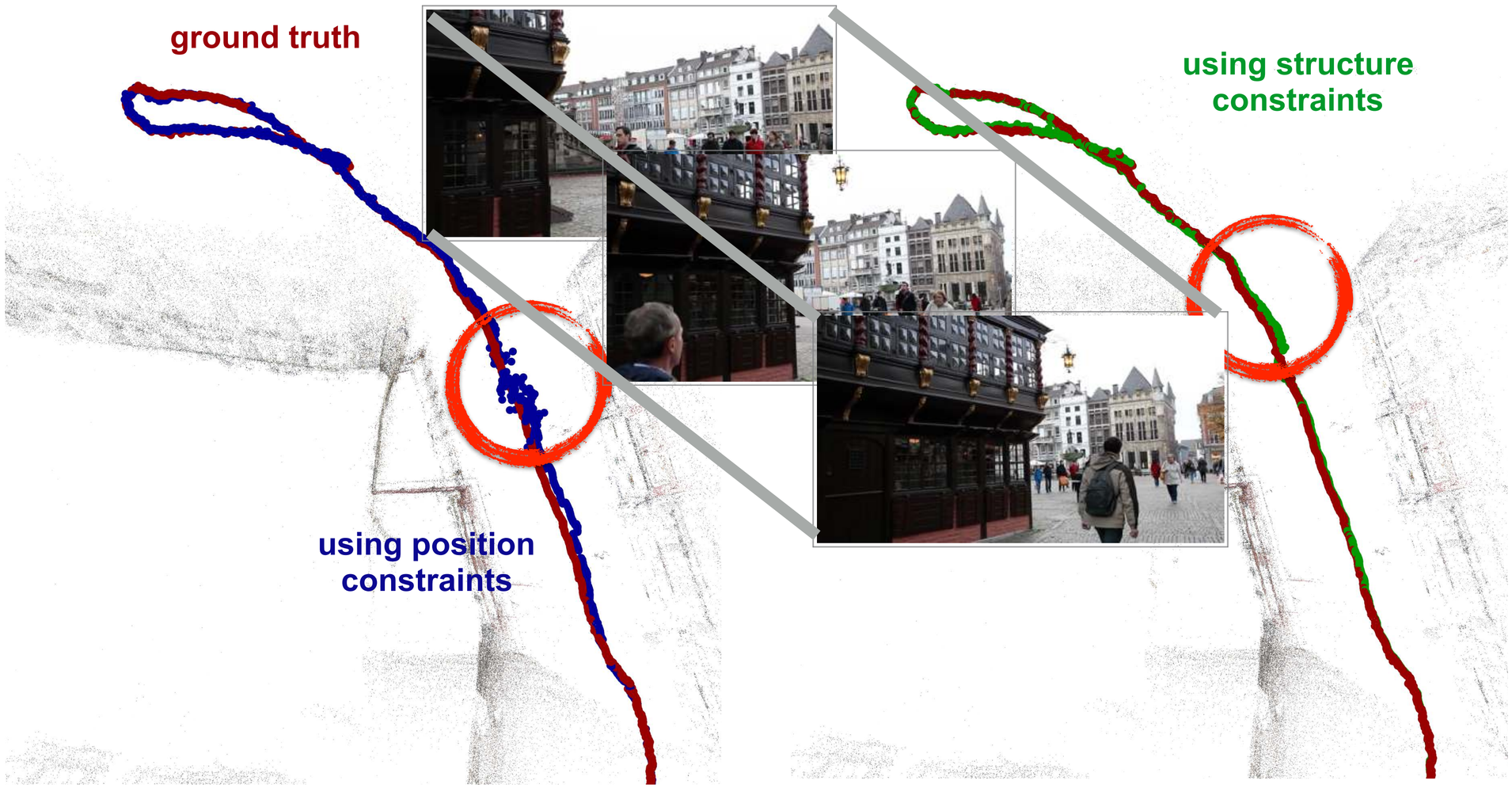


local pose tracking via SLAM

global localization

[Middelberg, Sattler, Untzelmann, Kobbelt. Scalable 6-DOF Localization on Mobile Devices. ECCV 2014]

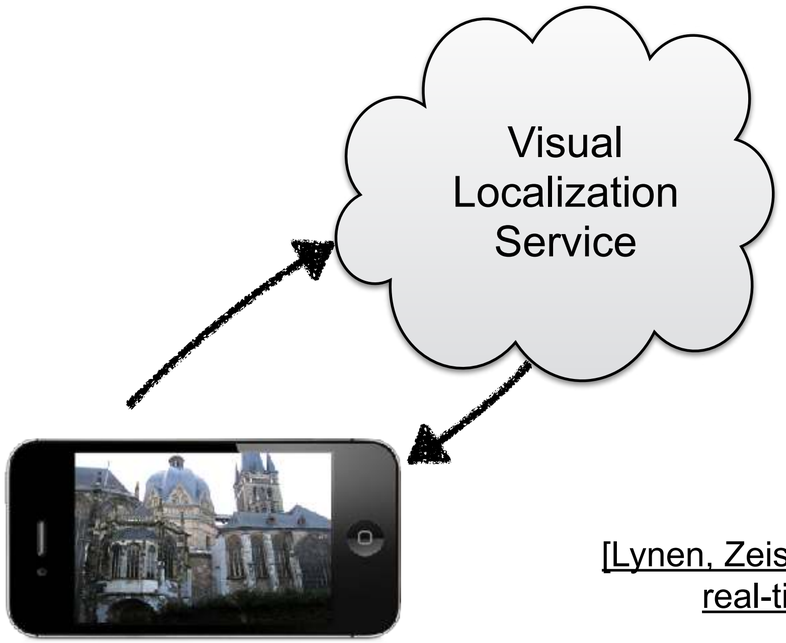
Pose Constraints vs. Structure Constraints



[Middelberg, Sattler, Untzelmann, Kobbelt. Scalable 6-DOF Localization on Mobile Devices. ECCV 2014]

Large-Scale Mobile Localization

- Limited computational power & memory capabilities
- Decoupling alone does not solve this problem



Combination of both

[Lynen, Zeisl, Aiger, Bosse, Hesch, Pollefeys, Siegwart, Sattler, Large-scale, real-time visual-inertial localization revisited, arXiv:1907.00338]

Server-Client Architecture

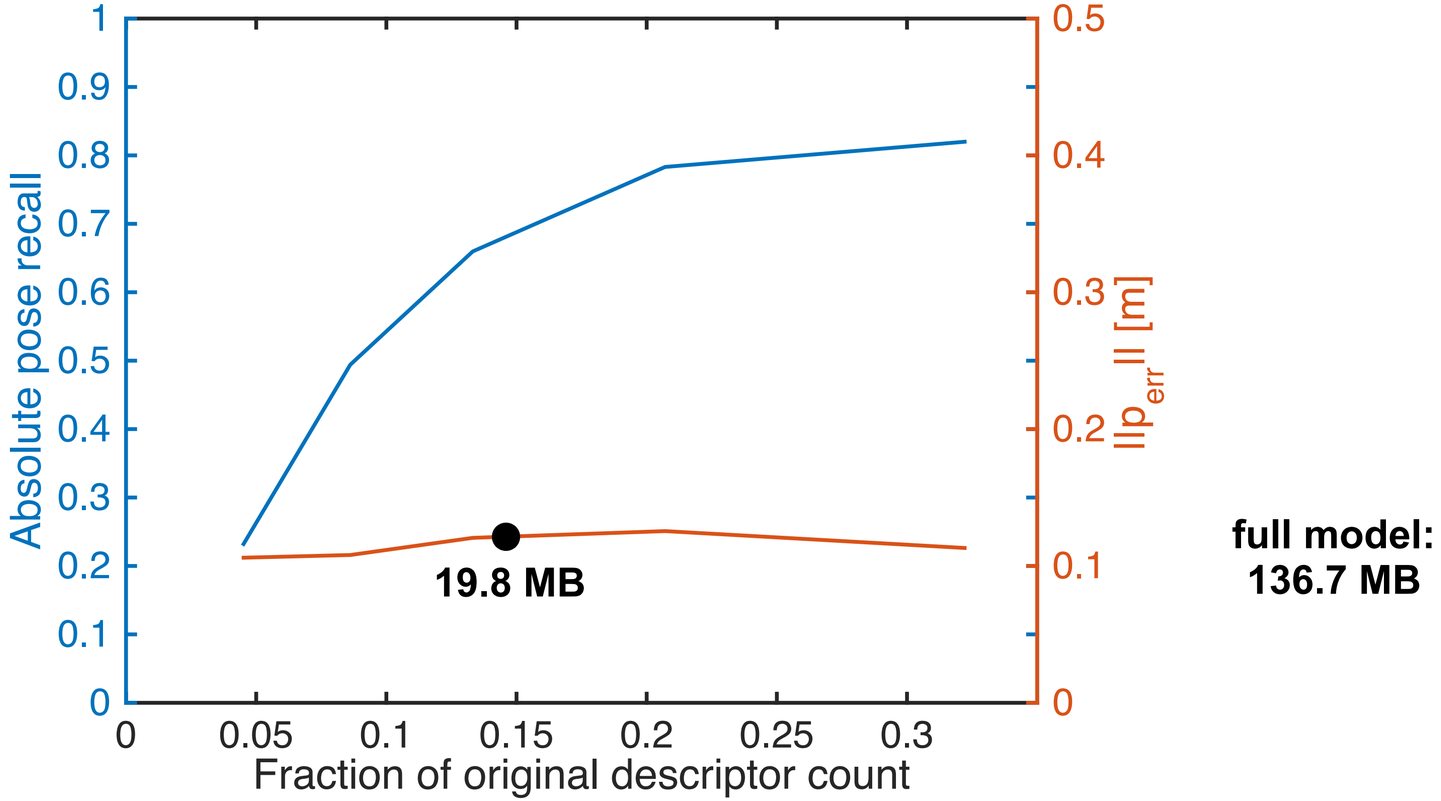
[Middelberg, Sattler, Untzelmann, Kobbelt. Scalable 6-DOF Localization on Mobile Devices. ECCV 2014]

Map Compression

[Lynen, Sattler, Bosse, Hesch, Pollefeys, Siegwart, Get Out of My Lab: Large-scale Real-Time Visual-Inertial Localization. RSS 2015]

Map Compression

- Use a subset of all 3D points in model [Li et al., Location Recognition using Prioritized Feature Matching. ECCV 2010]



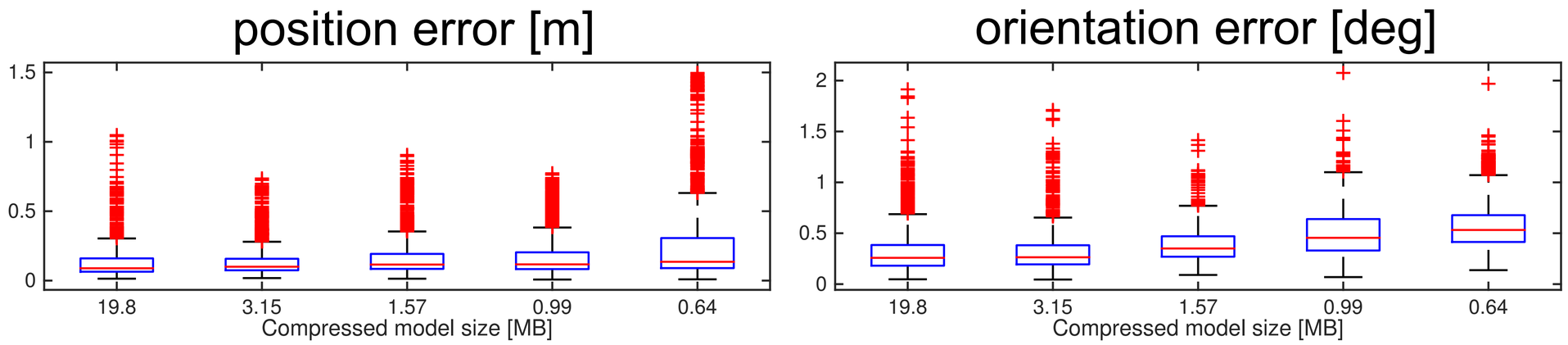
[Lynen, Sattler, Bosse, Hesch, Pollefeys, Siegwart, Get Out of My Lab: Large-scale Real-Time Visual-Inertial Localization. RSS 2015]

Map Compression

[Lynen, Sattler, Bosse, Hesch, Pollefeys, Siegwart, Get Out of My Lab: Large-scale Real-Time Visual-Inertial Localization. RSS 2015]

Map Compression

- Further compression by descriptor quantization [Jégou et al., PAMI'11] [code][Ge et al., CVPR'13] [Norouzi & Fleet, CVPR'13] [Kalantidis & Avrithis, CVPR'14]
- Still good accuracy with 4 MB or less (for 1 km trajectory)

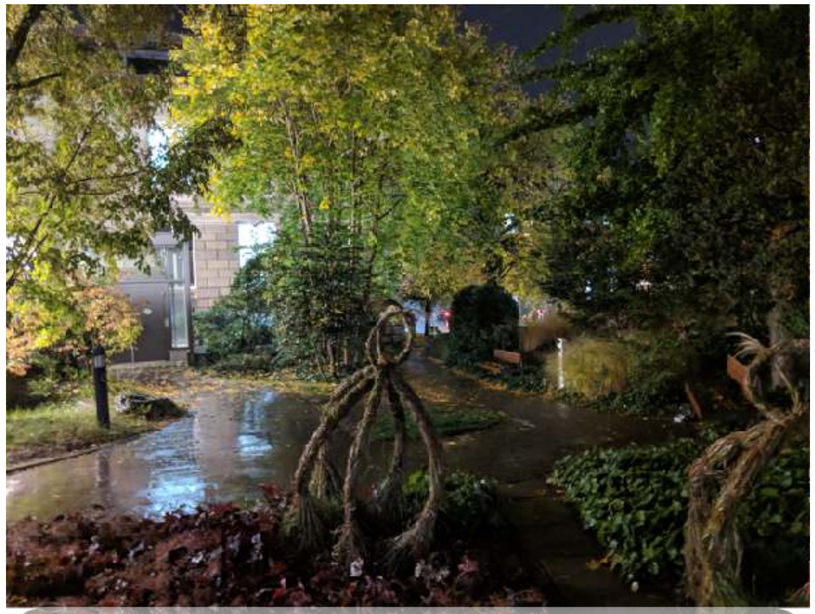
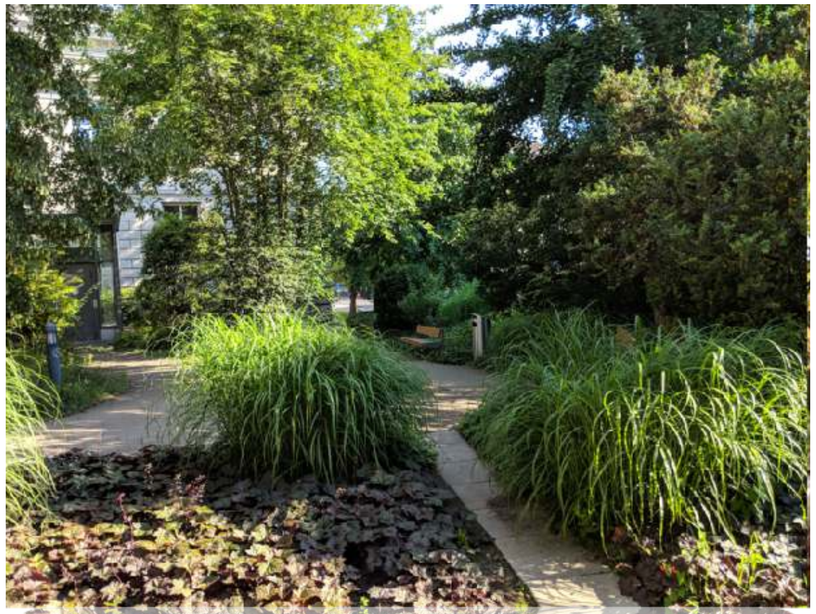


[Lynen, Sattler, Bosse, Hesch, Pollefeys, Siegwart, Get Out of My Lab: Large-scale Real-Time Visual-Inertial Localization. RSS 2015]

Takeaways

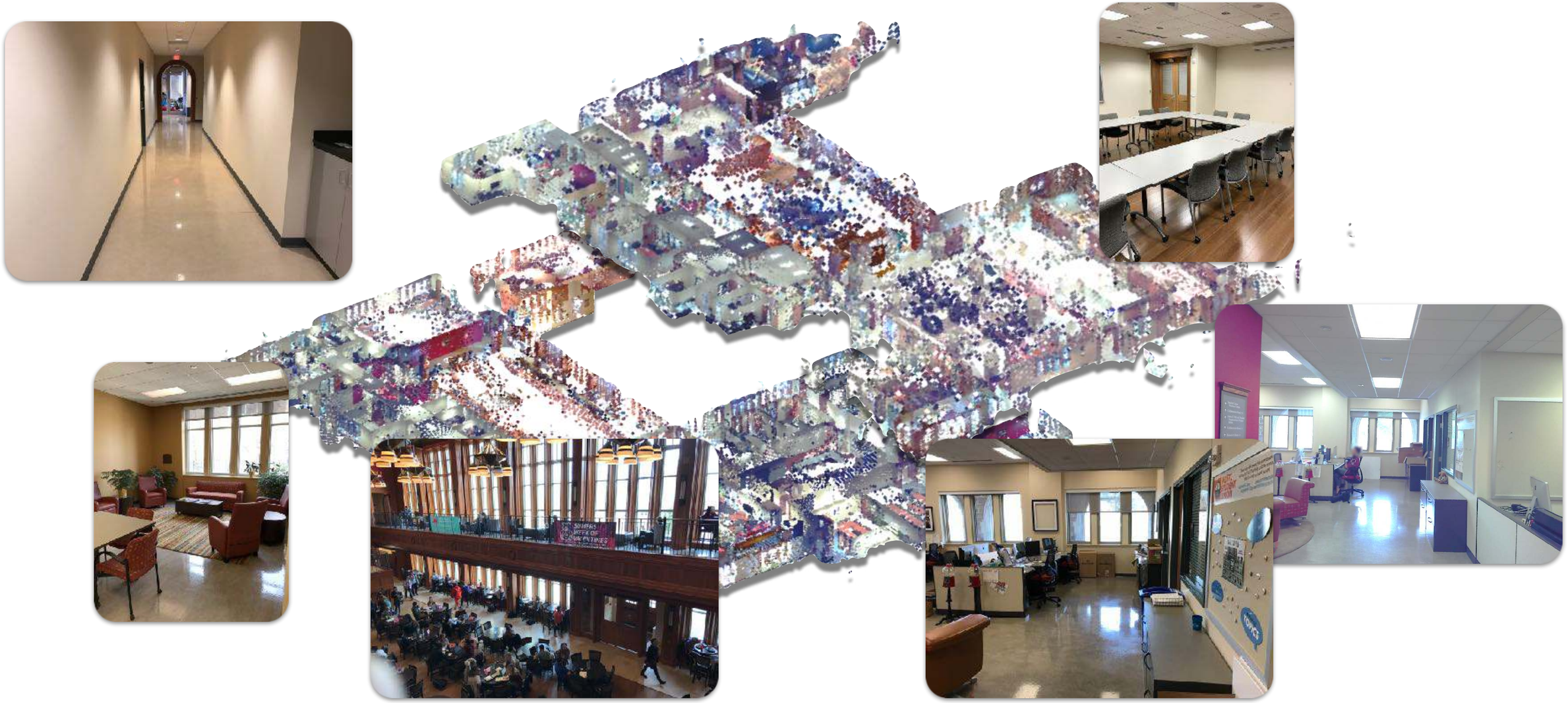
- **Combination** of **localization** (low frame rate, no drift) and **SLAM** / **Visual (Inertial) Odometry** (high frame rate, drift)
- **Map compression** necessary for memory and run-time efficiency
  - **Structure** compression: Select subset of 3D map points
  - **Appearance** compression: Descriptor quantization

Long-Term Visual Localization

World is **not static**, **appearance and geometry change**!
How **robust** are visual localization algorithms?

InLoc Dataset



[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

Weakly Textured Scenes

[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

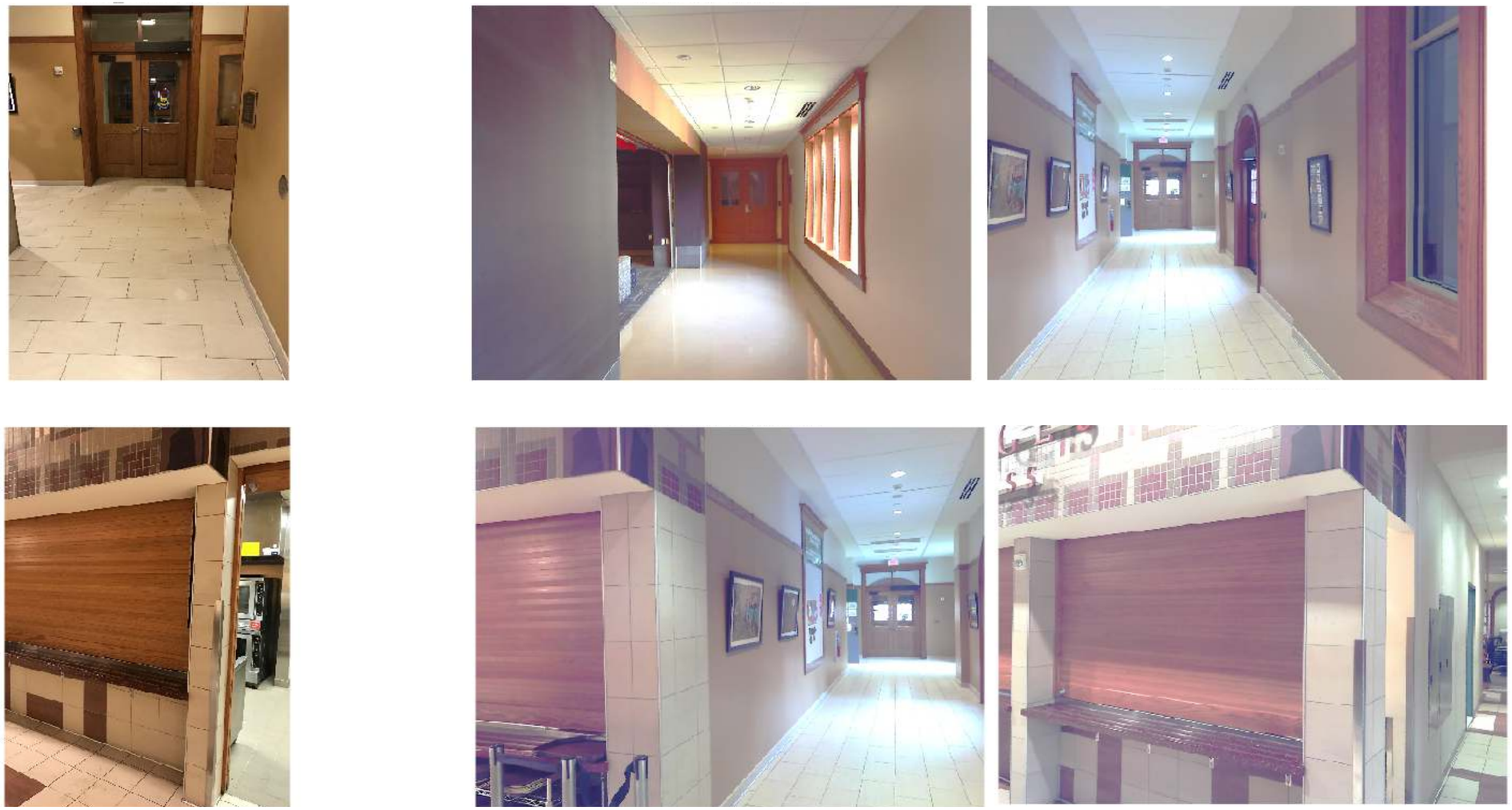
Complex Illumination

[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

Changes in Viewpoint & Time

[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

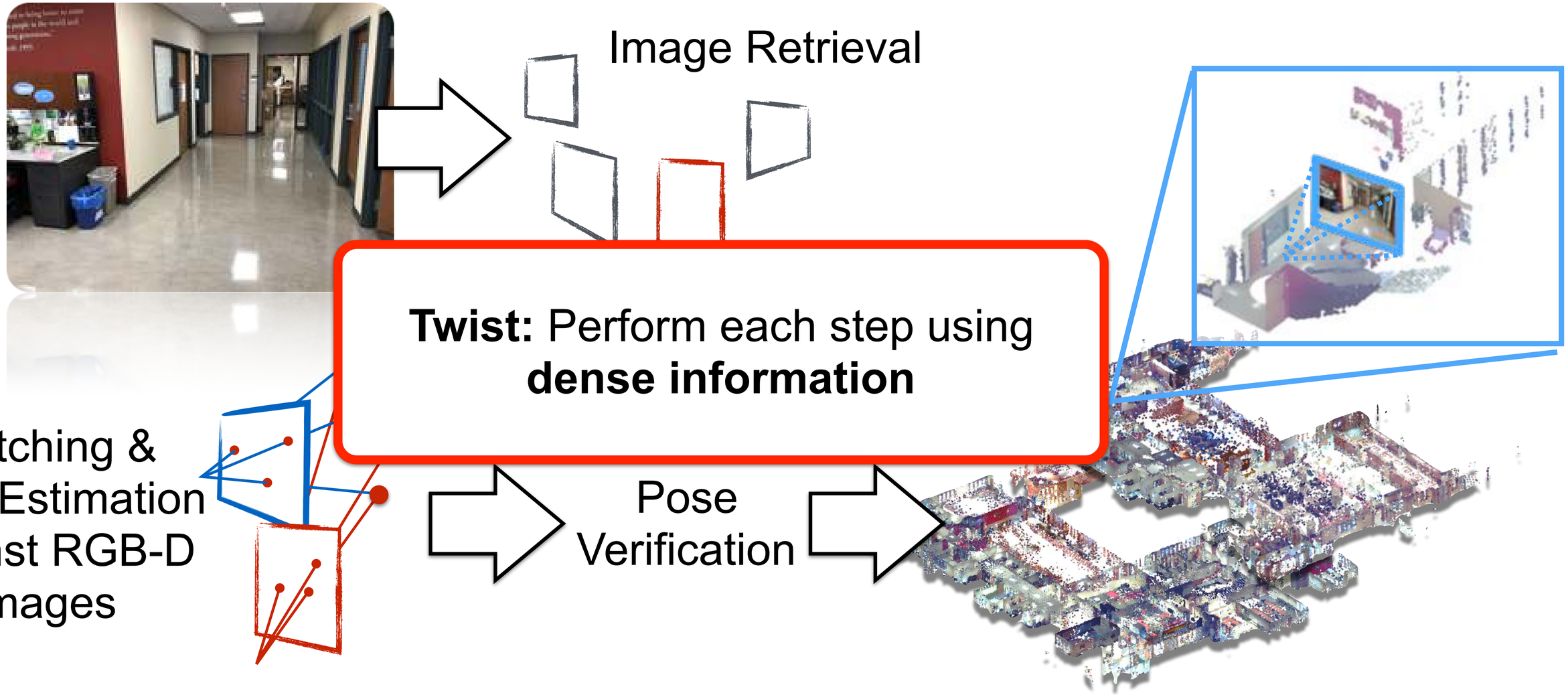
Repetitive Structures



slide credit: Hajime Taira

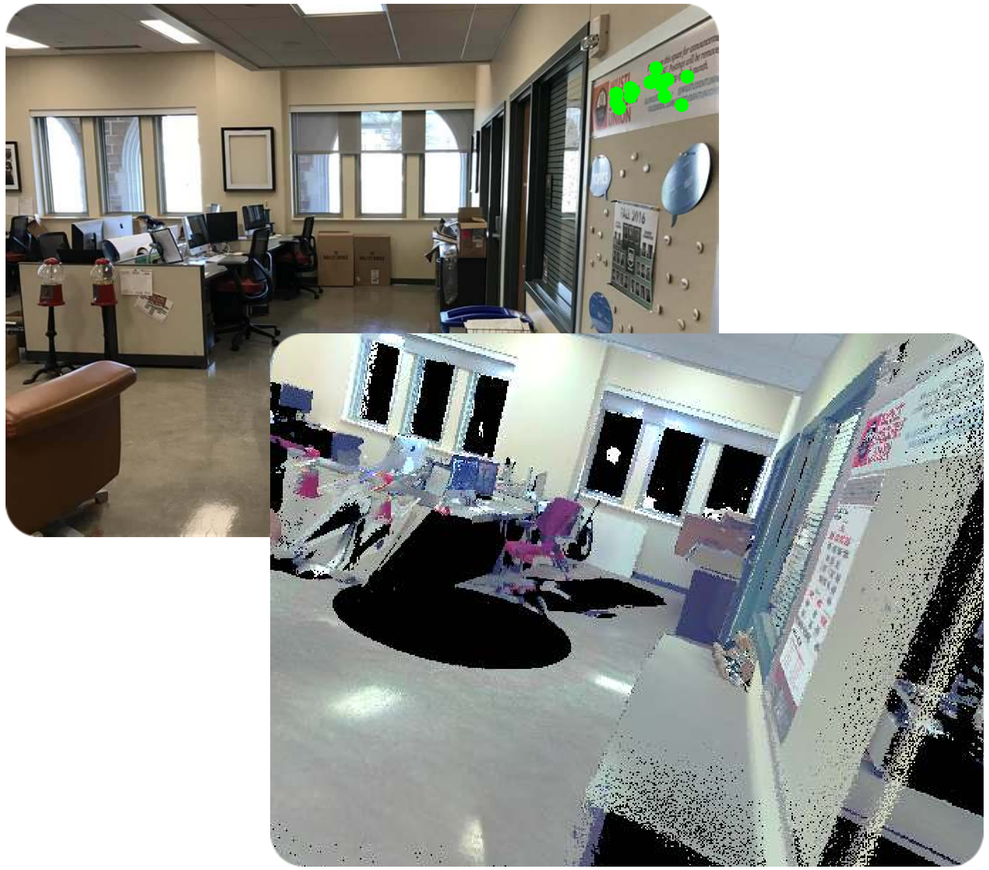
[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

Indoor Localization

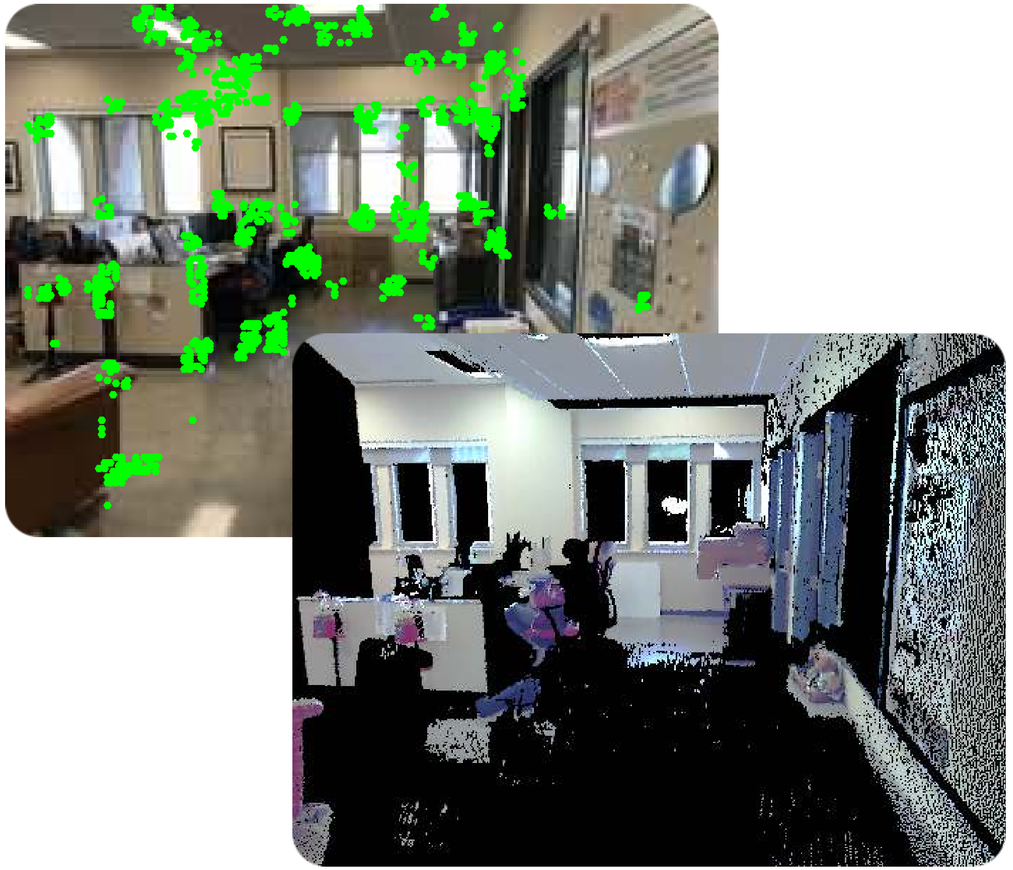


[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

Dense Feature Matching



Sparse Matching



Dense Matching

[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

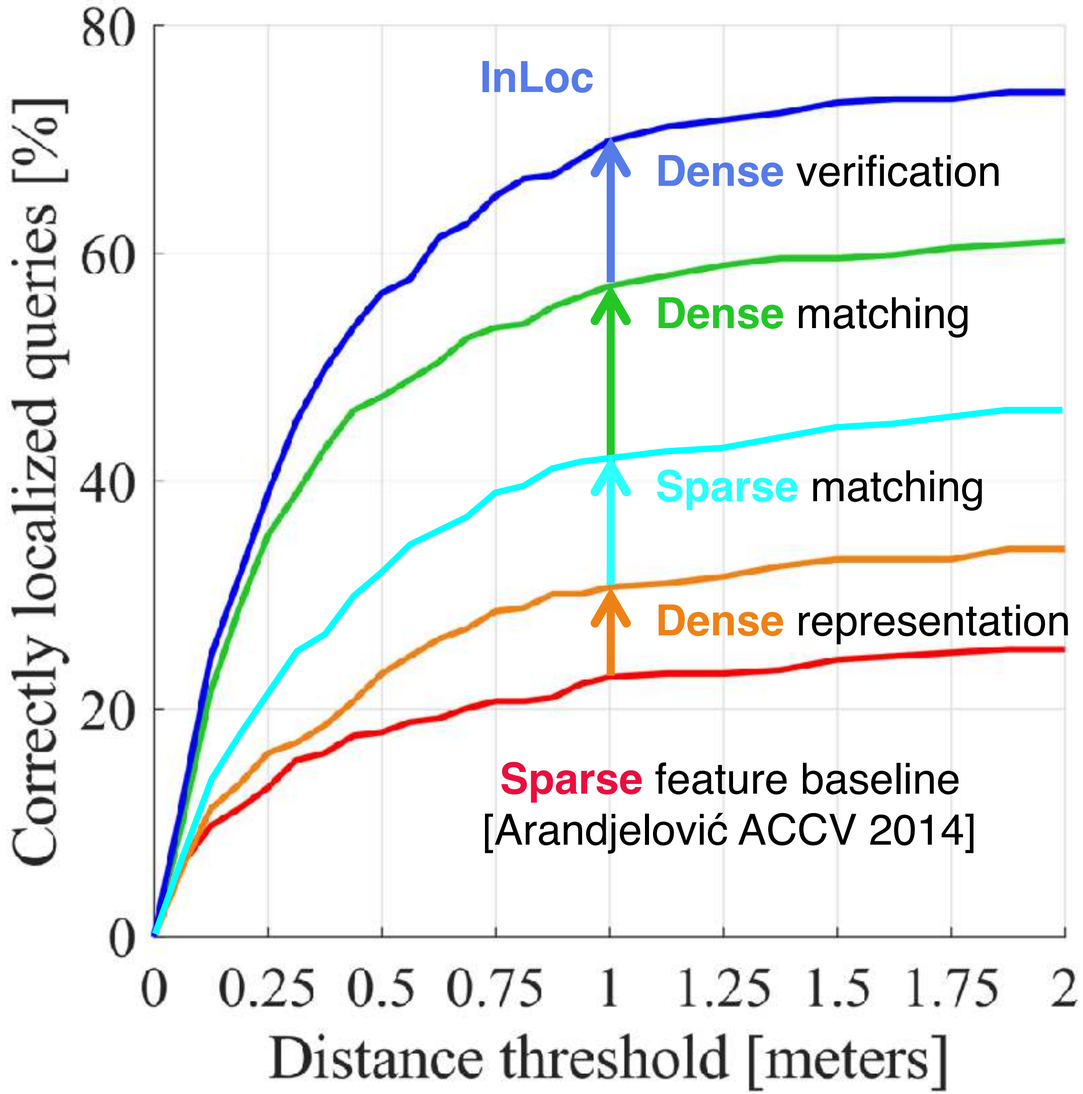
Dense Pose Verification

Candidate poses **Rendered views** **Error map**

slide credit: Hajime Taira

[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

“Dense” Indoor Localization



slide credit: Hajime Taira

[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

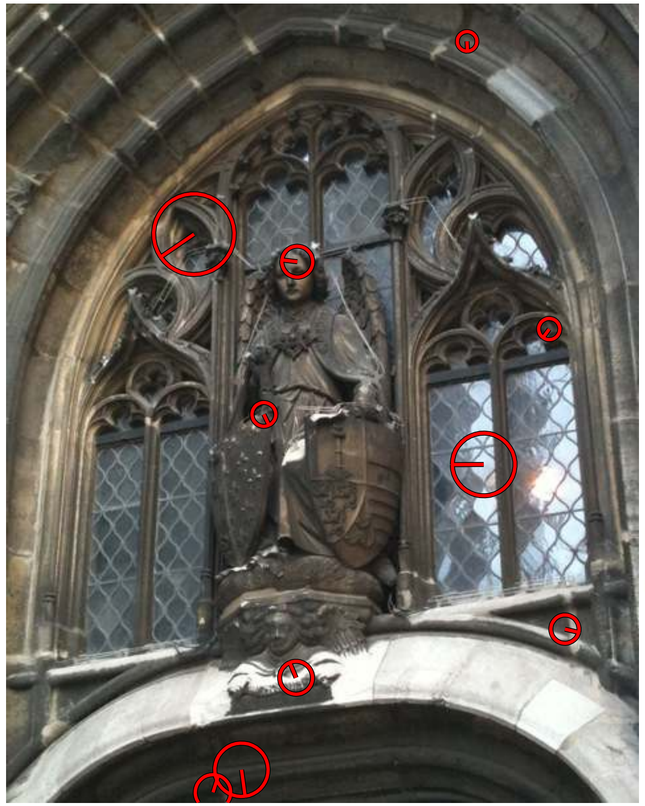
Takeaways

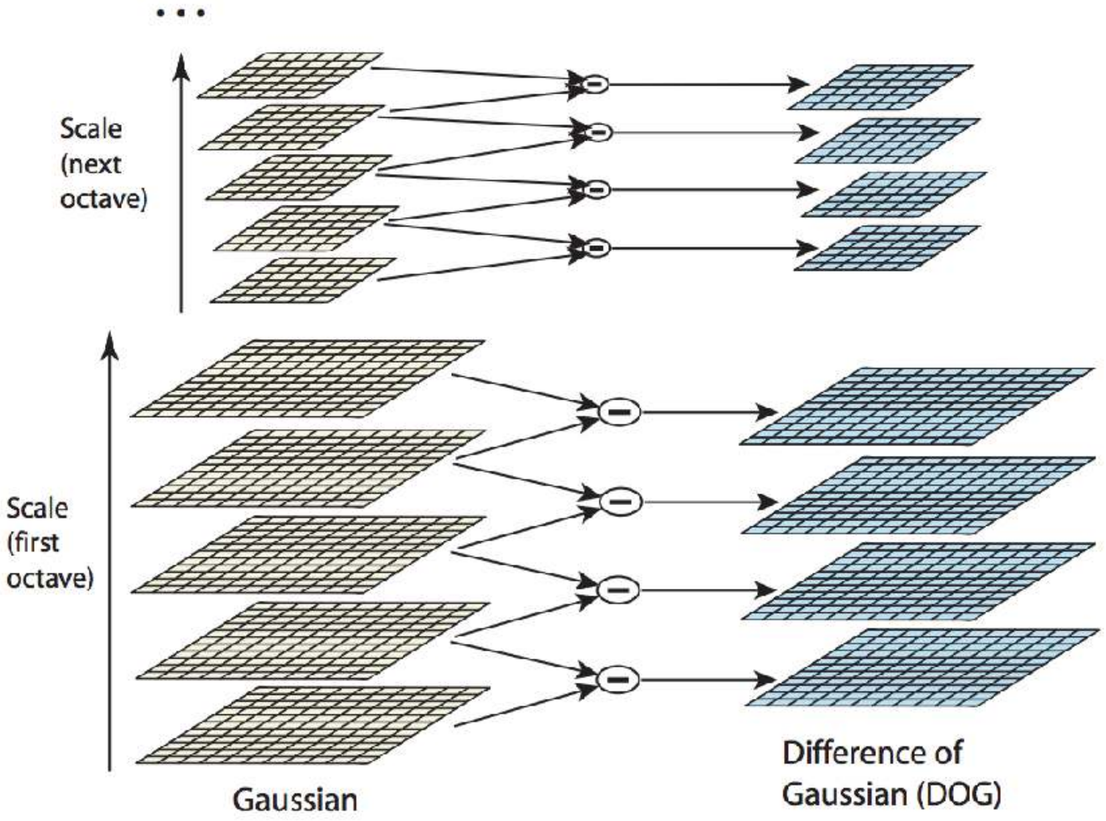
- **Higher-Level Features**: Model higher-level structures rather than low-level information (gradients, …)
- **Skip Keypoint Detector**: Densely extract and match features
- Code: http://www.ok.sc.e.titech.ac.jp/INLOC/
- **Limitations**: Run-time and memory
- **Do we really need dense features?**

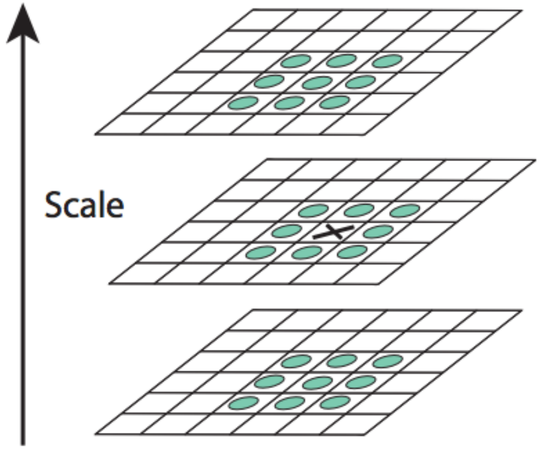
[Taira, Okutomi, Sattler, Cimpoi, Pollefeys, Sivic, Pajdla, Torii, InLoc: Indoor Visual Localization with Dense Matching and View Synthesis. CVPR 2018]

- Visual Localization
- **Learning Feature Detectors**

Recap: Local Features (SIFT)







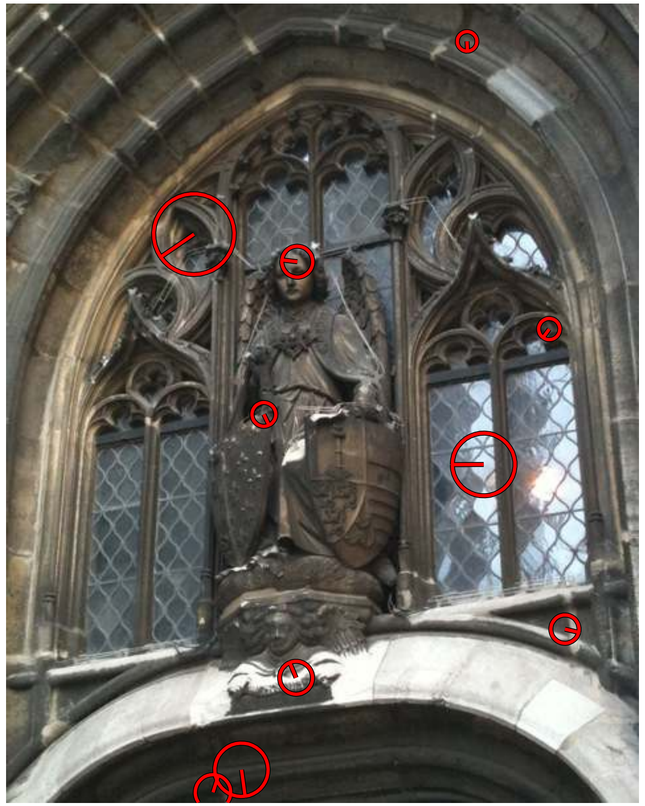
Keypoint Detection

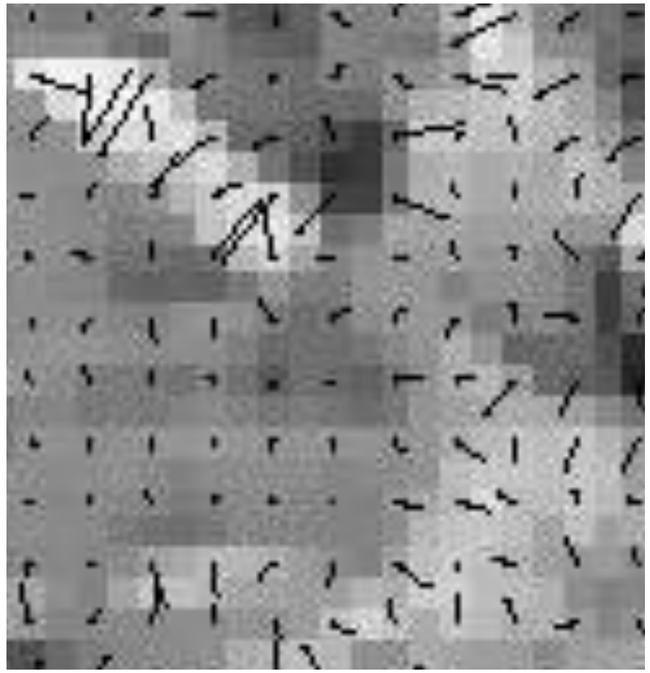
scale-space construction

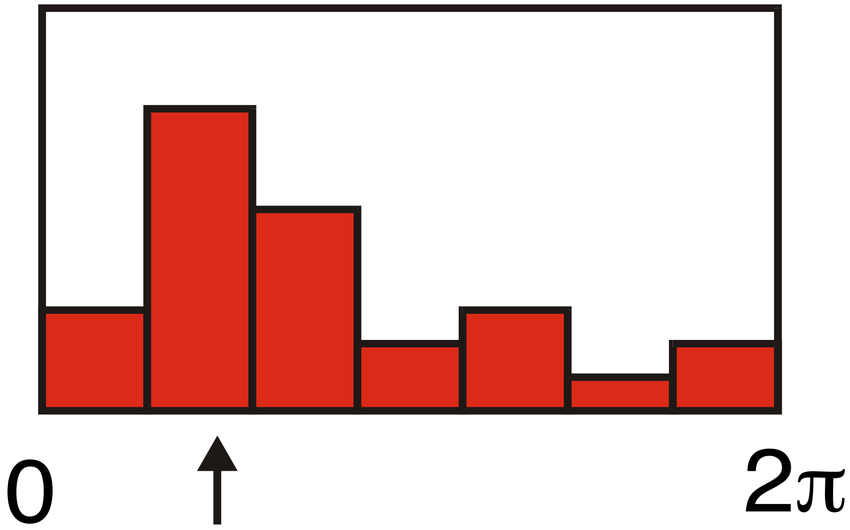
detect local extrema

[Lowe, Distinctive Image Features from Scale-Invariant Keypoints, IJCV 2004]

Recap: Local Features (SIFT)







find dominant gradient direction per region

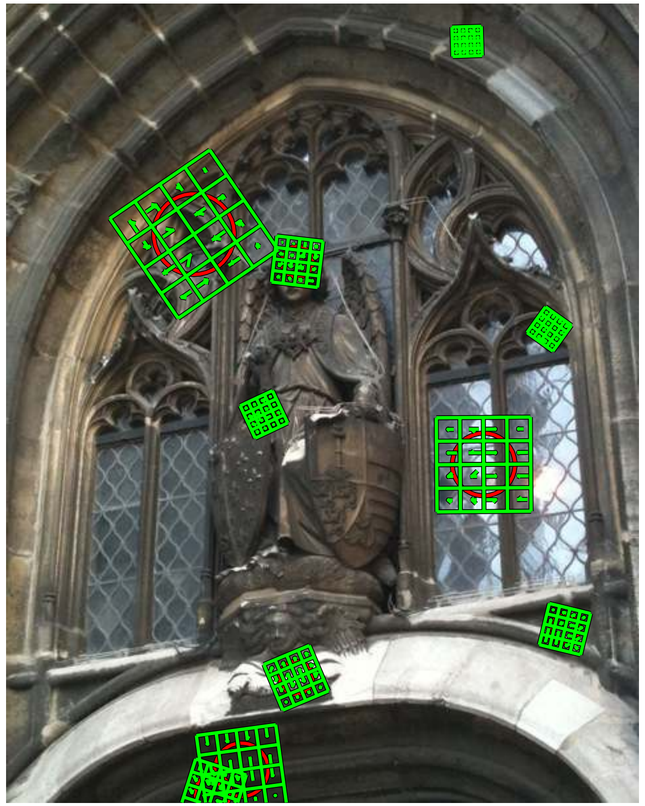
Keypoint Detection

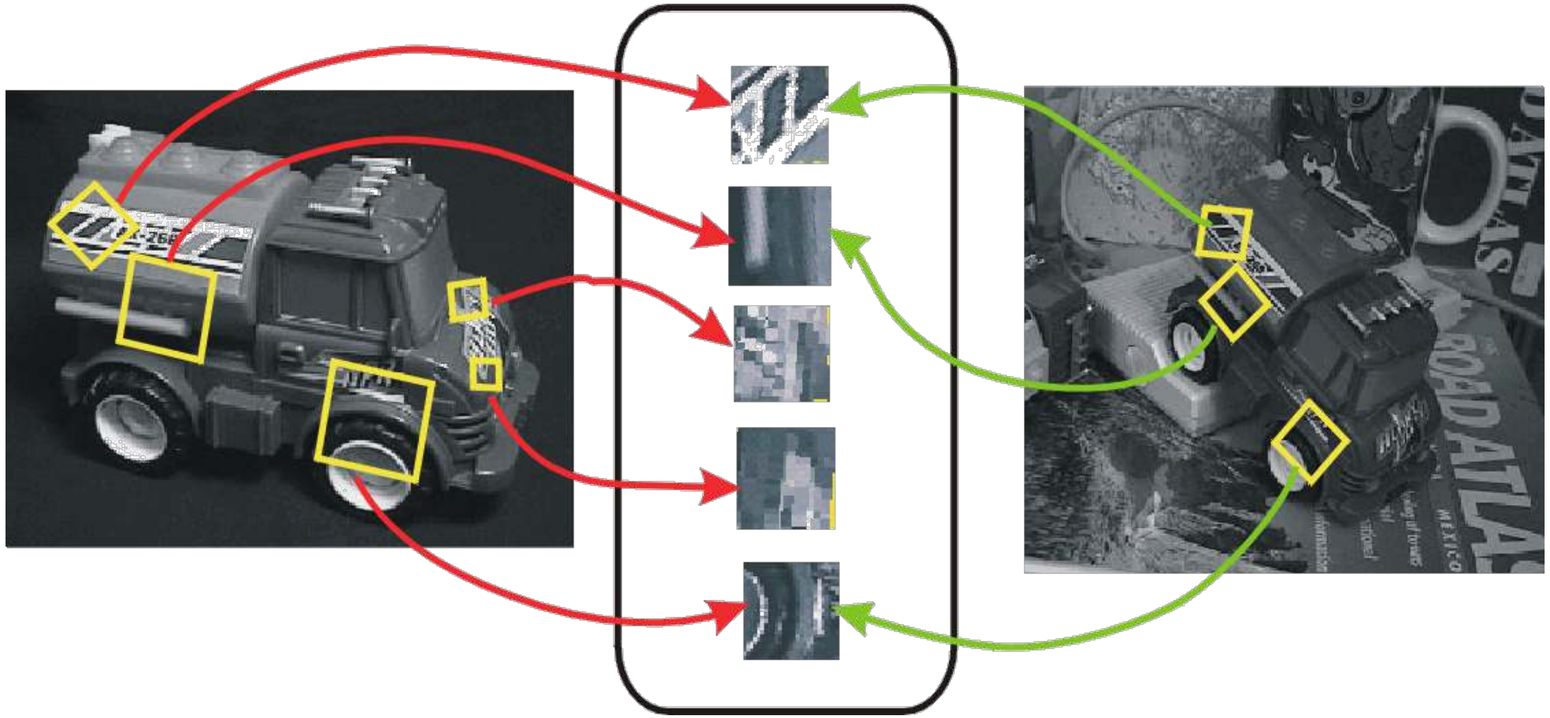
Orientation Assignment

[Lowe, Distinctive Image Features from Scale-Invariant Keypoints, IJCV 2004]

slide credit: Marc Pollefeys

Recap: Local Features (SIFT)





align patches to dominant direction

Keypoint Detection

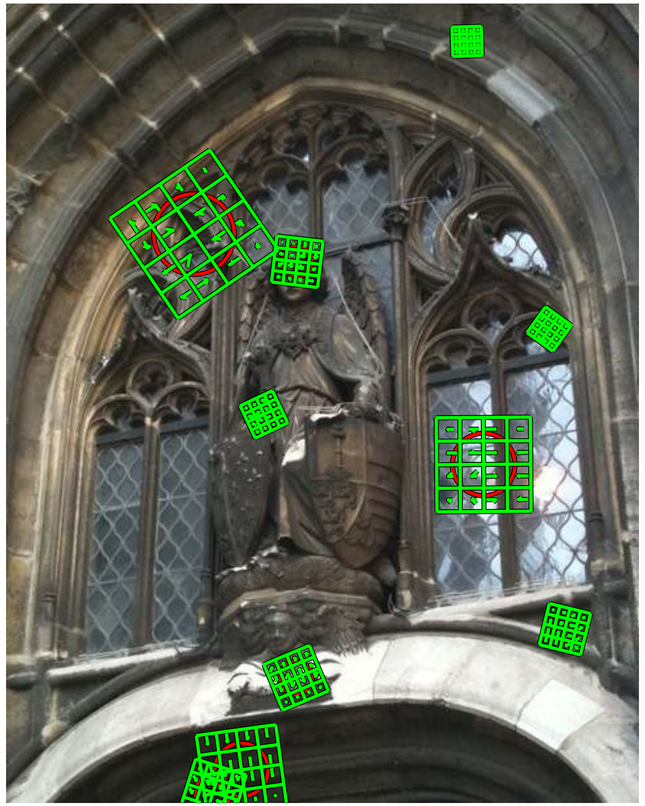
Orientation Assignment

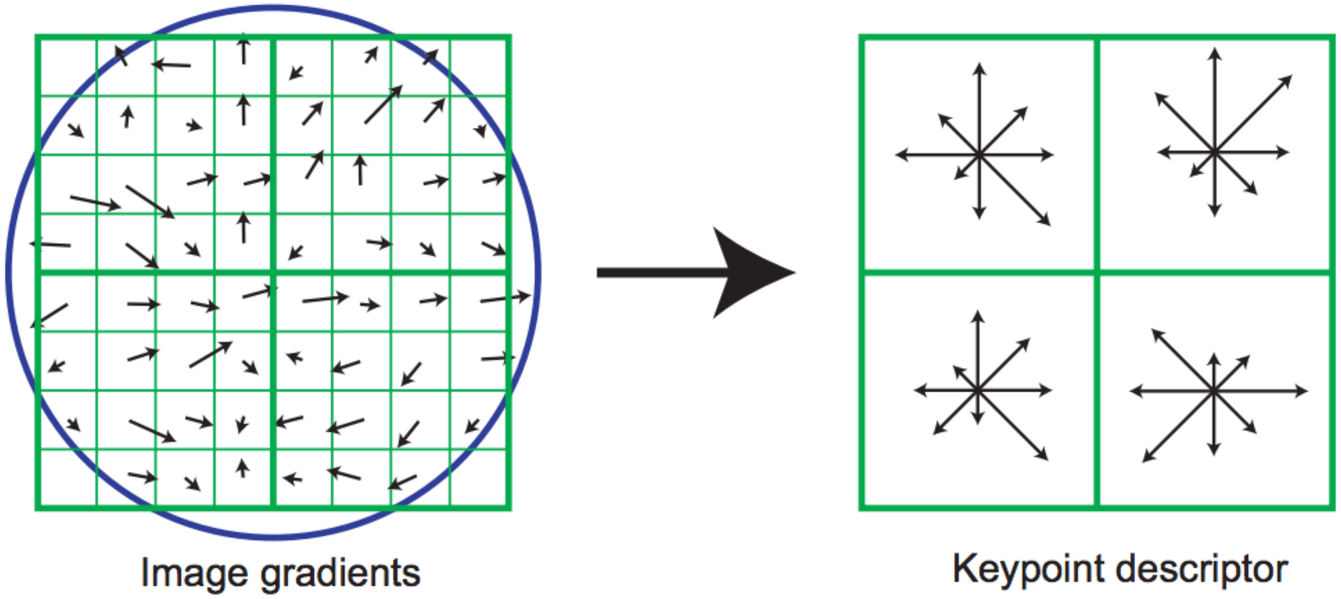
Descriptor Computation

[Lowe, Distinctive Image Features from Scale-Invariant Keypoints, IJCV 2004]

slide credit: Marc Pollefeys

Recap: Local Features (SIFT)





compute descriptor

Keypoint Detection

Orientation Assignment

Descriptor Computation

[Lowe, Distinctive Image Features from Scale-Invariant Keypoints, IJCV 2004]

Learning Features

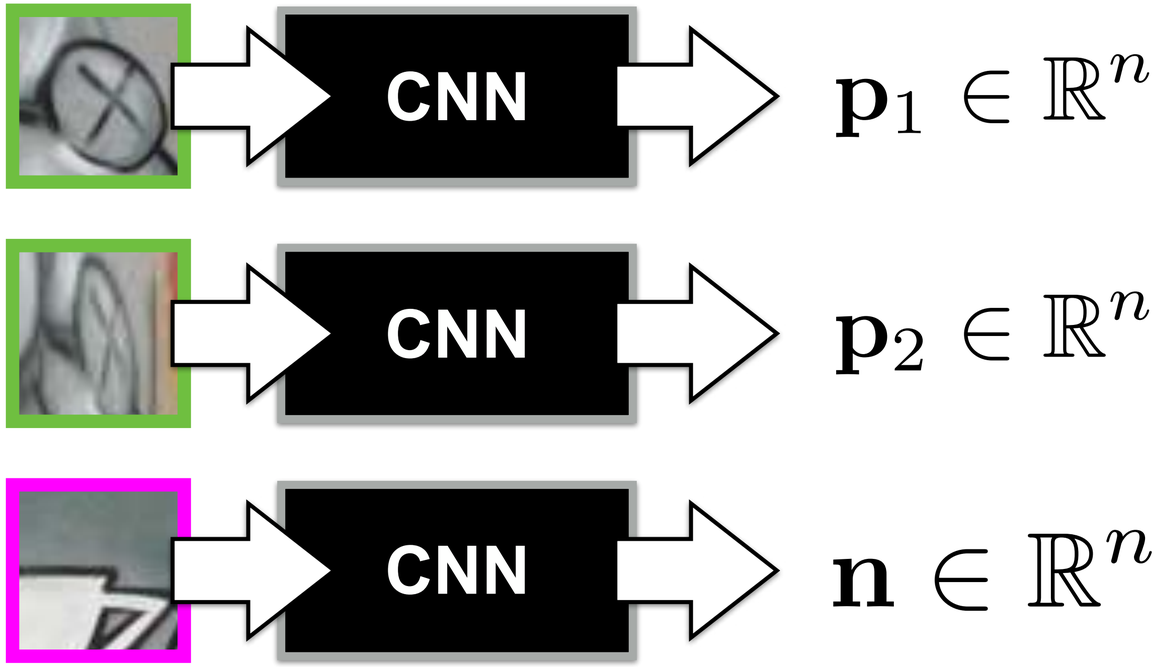
- SIFT is nearly 20 years old!
- Can we learn better local features via deep learning?
- How can we learn local features?

Learning Features

- **Learning Feature Descriptors**
- Learning Feature Detectors
- Joint Feature Detection and Description

Learning Feature Descriptors

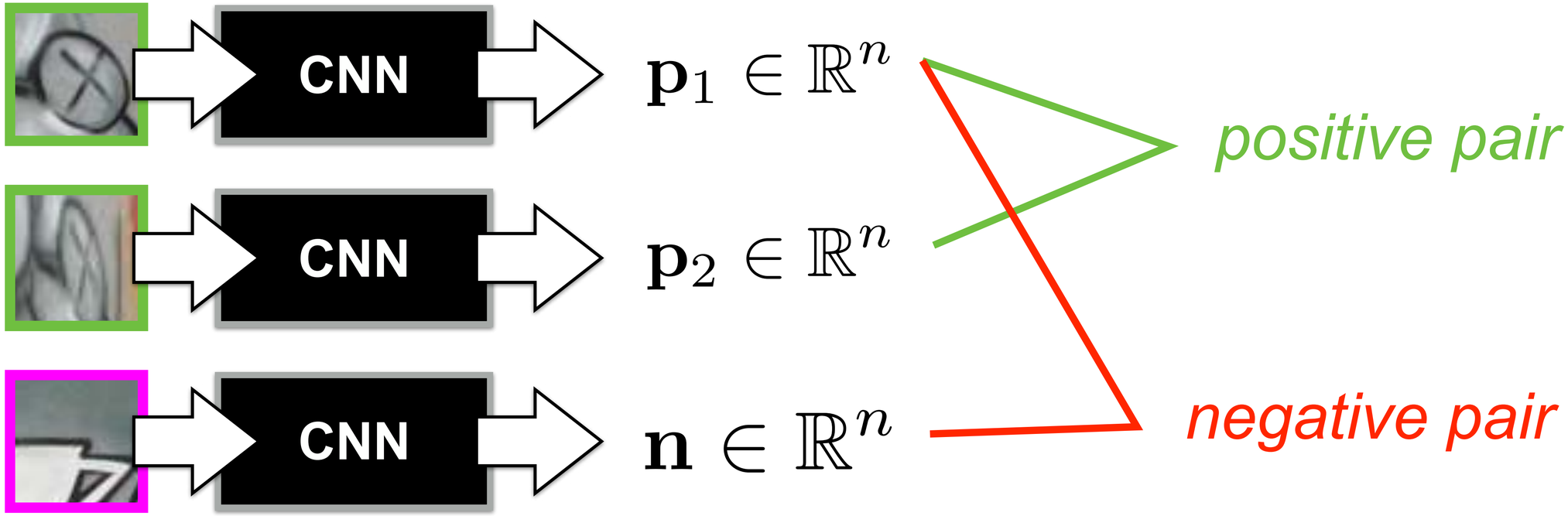
- Learn mapping from patch to descriptor in $\mathbb{R}^n$
- How define the training loss?



[Simo-Serra, Trulls, Ferraz, Kokkinos, Fua, Moreno-Noguer, Discriminative Learning of Deep Convolutional Feature Point Descriptors, ICCV 2015]

Learning Feature Descriptors

- Learn mapping from patch to descriptor in $\mathbb{R}^n$
- How define the training loss?



[Simo-Serra, Trulls, Ferraz, Kokkinos, Fua, Moreno-Noguer, Discriminative Learning of Deep Convolutional Feature Point Descriptors, ICCV 2015]

Learning with Pairs

- Learn mapping from patch to descriptor in $\mathbb{R}^n$
- Contrastive loss:

$$l(\mathbf{p}, \mathbf{p}') = \begin{cases} ||\mathbf{p} - \mathbf{p}'||_2 & \text{for positive pair} \\ \end{cases}$$

- Enforces global distance. Always possible?

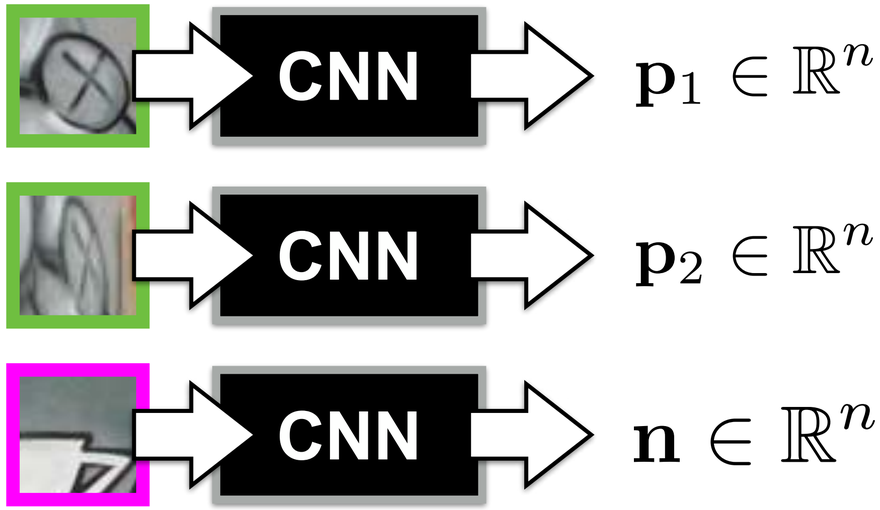
[Simo-Serra, Trulls, Ferraz, Kokkinos, Fua, Moreno-Noguer, Discriminative Learning of Deep Convolutional Feature Point Descriptors, ICCV 2015]

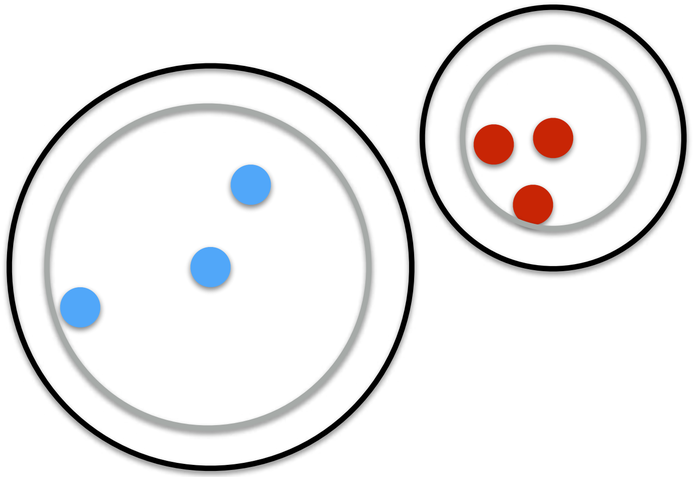
Learning with Triplets

- Learn mapping from patch to descriptor in $\mathbb{R}^n$
- Margin ranking loss:

margin

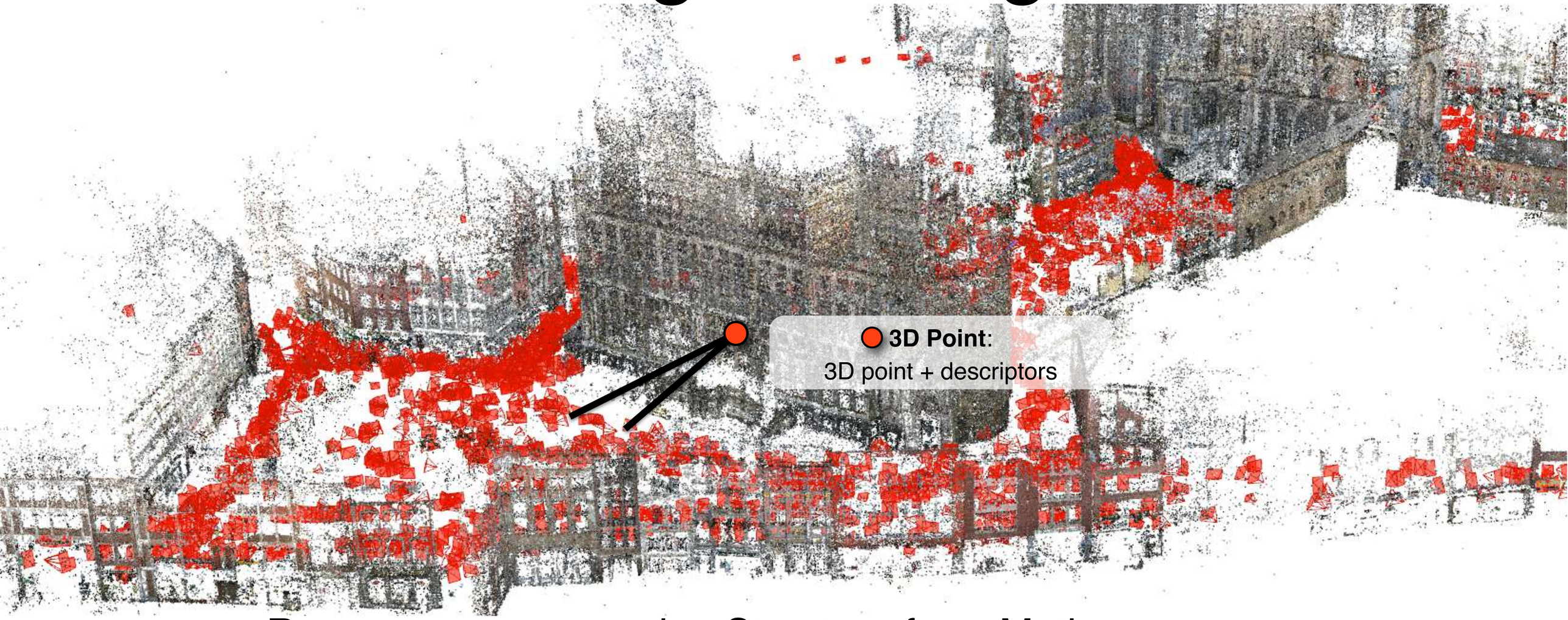
$$\max(0, \eta + ||\mathbf{p}_1 - \mathbf{p}_2||_2 - ||\mathbf{n} - \mathbf{p}_2||_2)$$





[Balntas, Riba, Ponsa, Mikolajczyk, Learning local feature descriptors with triplets and shallow convolutional neural networks, BMVC 2016]

Obtaining Training Data



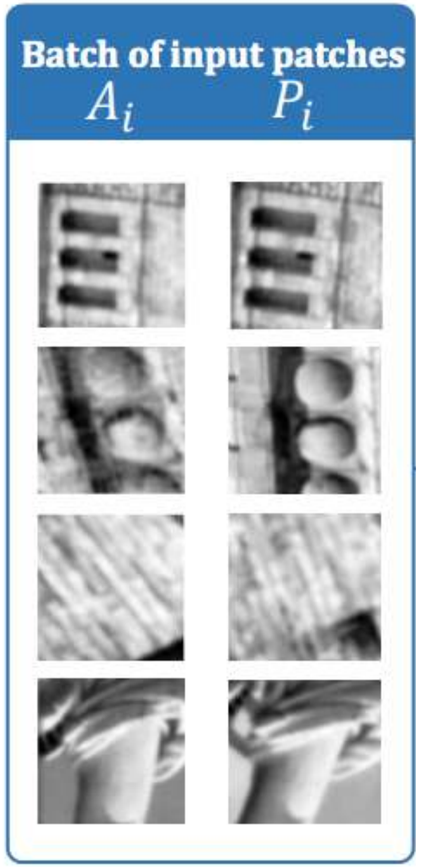
- Reconstruct scene using Structure-from-Motion
- Positive pairs: patches belonging to the same 3D point

Learning from Transitivity

- How can we learn **better** descriptors?

Selecting Pairs

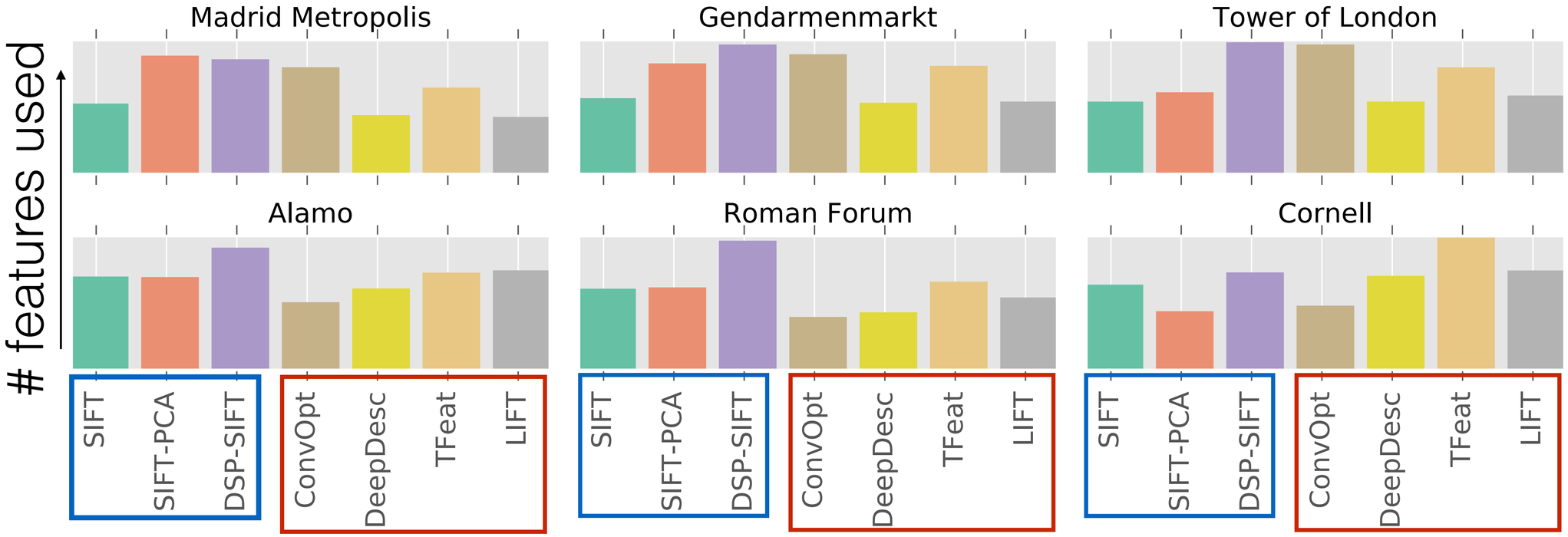
- How to select good pairs (positive & negative)?
- Sample hard cases per batch!



[Mishchuk, Mishkin, Radenovic, Matas, Working hard to know your neighbor's margins: Local descriptor learning loss, NIPS 2017]

Hand-Crafted vs. Learned Descriptors

- Comparing learned with hand-crafted descriptors (SIFT variants)
- Evaluated on Structure-from-Motion task
- Measure: Number of triangulated features (higher = better)



[Schönberger, Hardmeier, Sattler, Pollefeys, Evaluation of Hand-Crafted and Learned Local Features. CVPR 2017]

My Take

- Most learned descriptors are trained on rather small datasets
- As a result, they do not generalize well to new scenes too dissimilar to their training data
- Problem should resolve itself as more training data becomes available

Some Notes

- Descriptor learning predates deep learning era:
  - [Brown, Hua, Winder, Discriminative Learning of Local Image Descriptors, PAMI 2011] (conference paper published in 2007)
  - [Simonyan, Vedaldi, Zisserman, Learning local feature descriptors using convex optimisation, PAMI 2014] (conference paper published 2012)

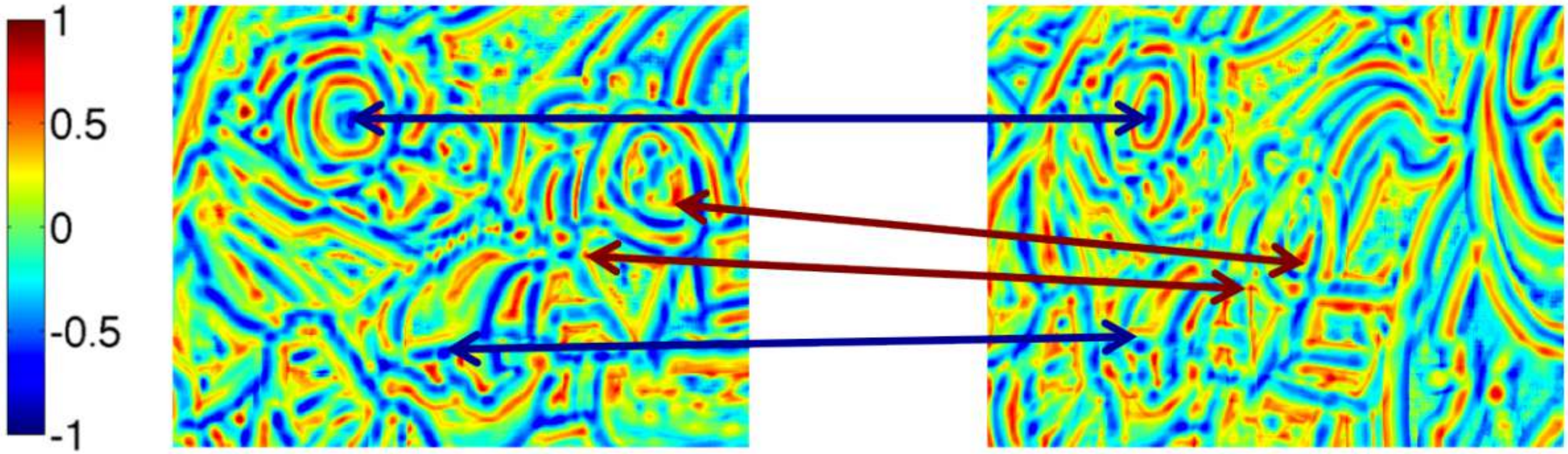
Learning Features

- Learning Feature Descriptors
- **Learning Feature Detectors**
- Joint Feature Detection and Description

Learning Feature Detectors

- What are properties of a good feature detector?
  - Repeatability, stability, viewpoint invariance
  - Fire at “interesting regions” suitable for matching
- How to model this mathematically?
- How to train a detector from scratch without any bias to existing solutions?

Learning Feature Detectors

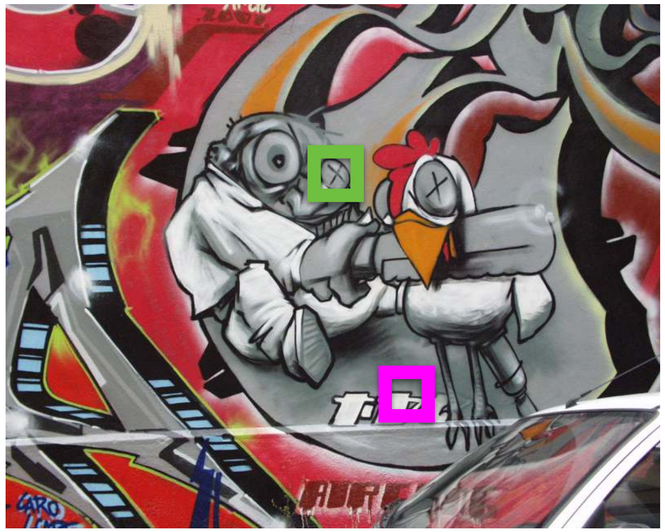


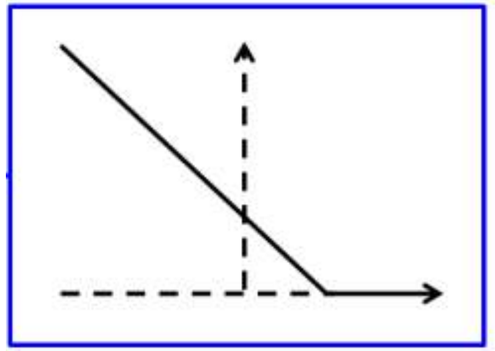
- Learn function $H(x|w)$: $\mathbb{R}^2 \rightarrow [-l, l]$ with parameters w
- Interesting points are close to $-l$ or l
- Repeatability = consistent ranking under transformations

[Schönberger, Hardmeier, Sattler, Pollefeys, Evaluation of Hand-Crafted and Learned Local Features. CVPR 2017]

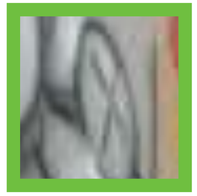
Learning to Rank

- Learn consistent ranking $H(x|w)$:





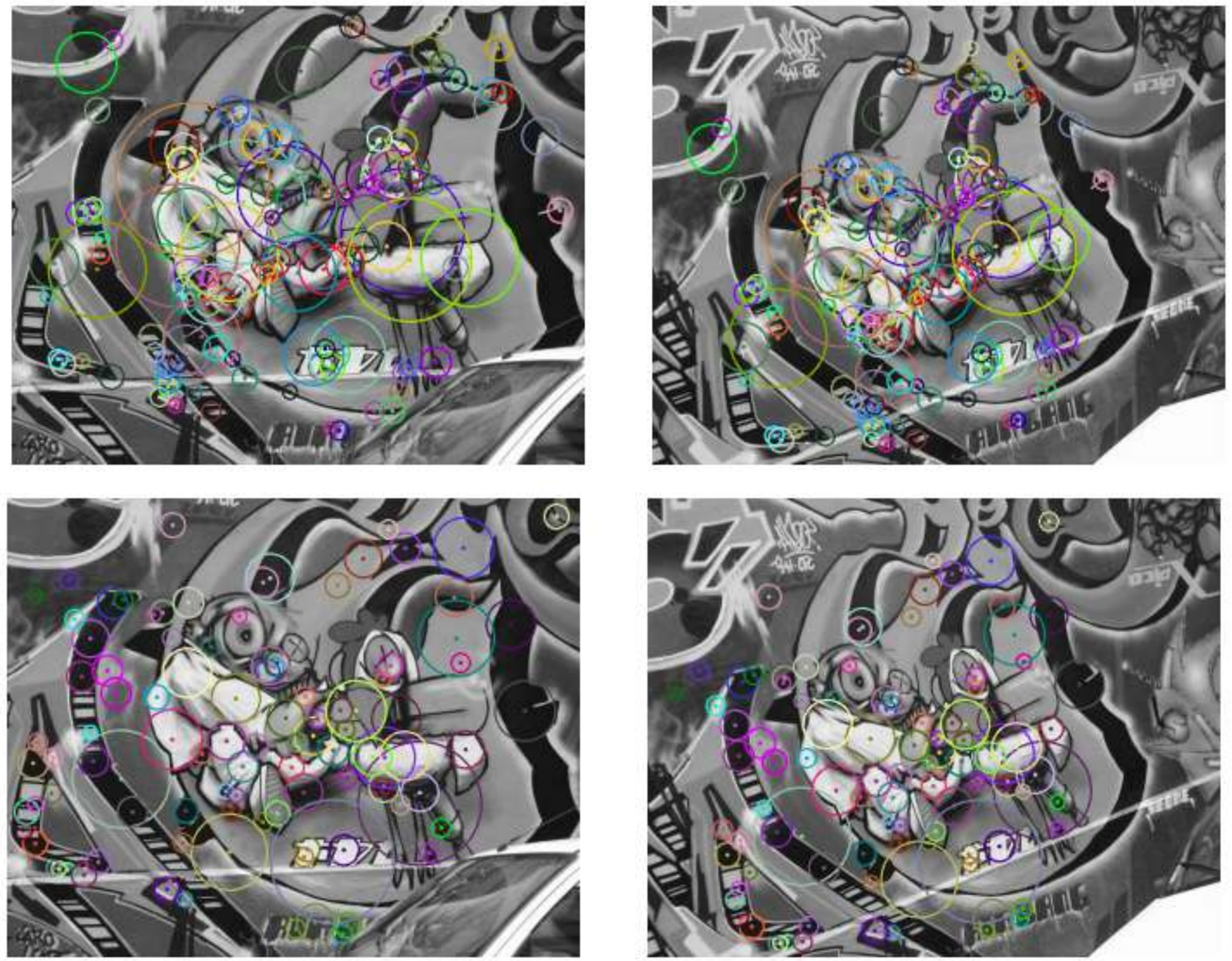
Hinge loss

$(H($ $|w) > H($ $|w)) \Leftrightarrow (H($  $|w) > H($ $|w)) > 0$

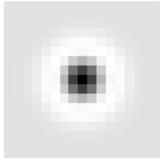
1

[Schönberger, Hardmeier, Sattler, Pollefeys, Evaluation of Hand-Crafted and Learned Local Features. CVPR 2017]

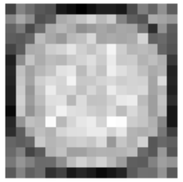
Detection Results



Difference-of-Gaussians (SIFT)

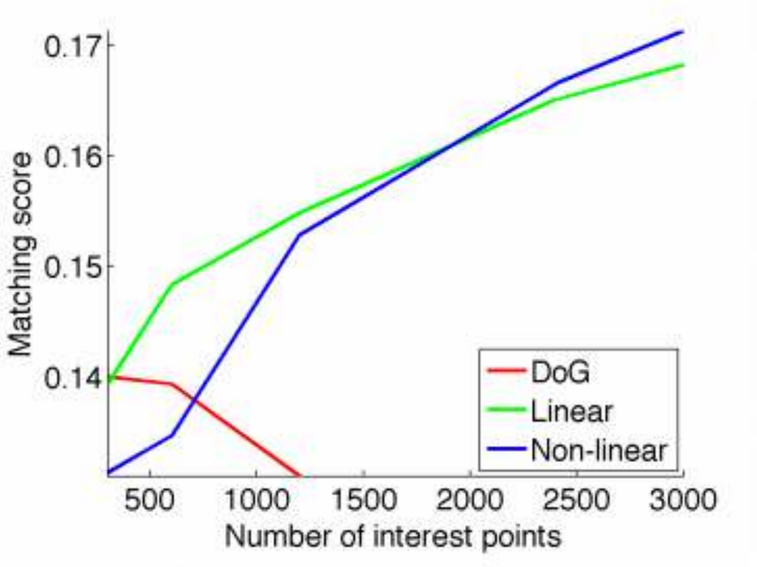


learned

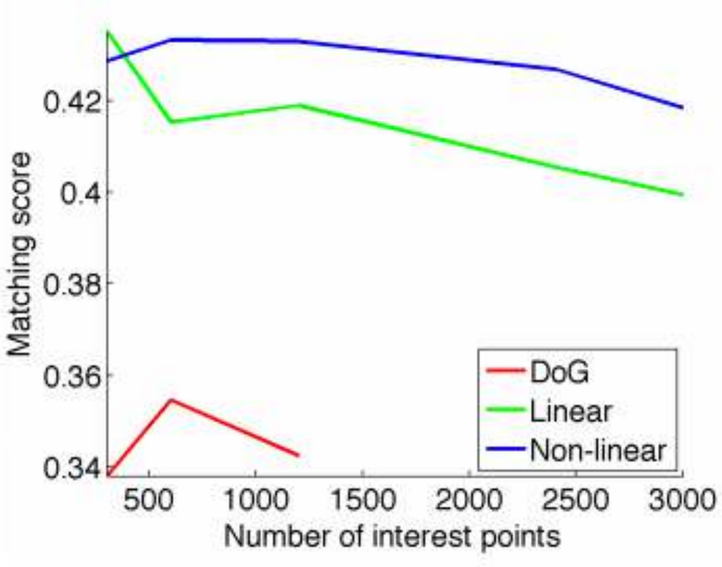


[Schönberger, Hardmeier, Sattler, Pollefeys, Evaluation of Hand-Crafted and Learned Local Features. CVPR 2017]

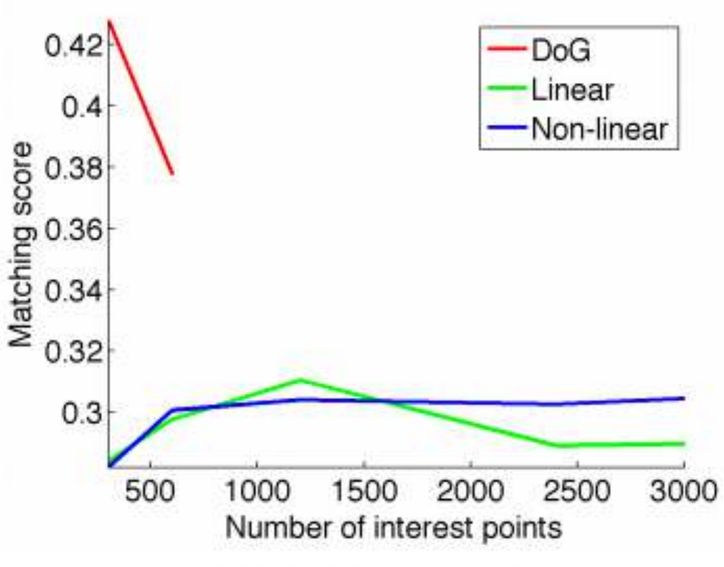
Matching Results



Wall (viewpoint)



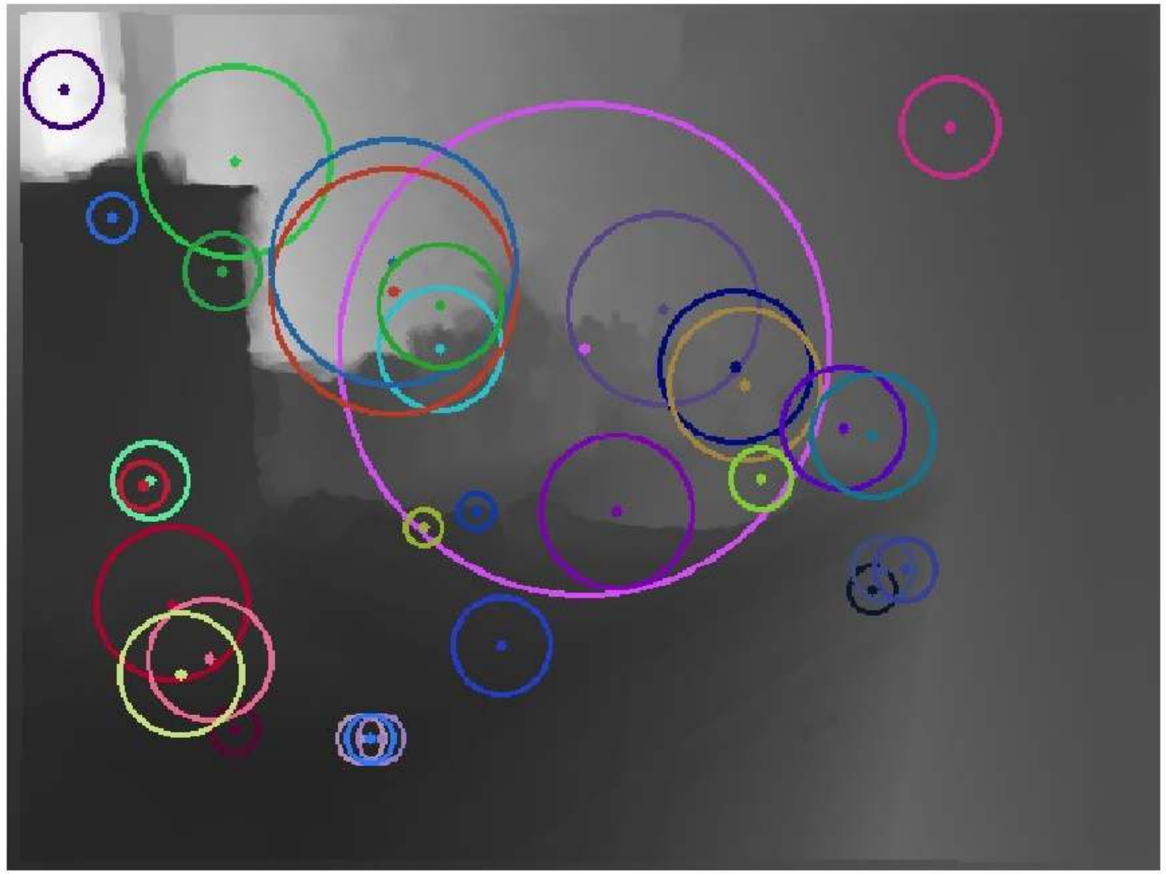
Leuven (illumination)



UBC (jpeg)

[Schönberger, Hardmeier, Sattler, Pollefeys, Evaluation of Hand-Crafted and Learned Local Features. CVPR 2017]

Multi-Modal Features



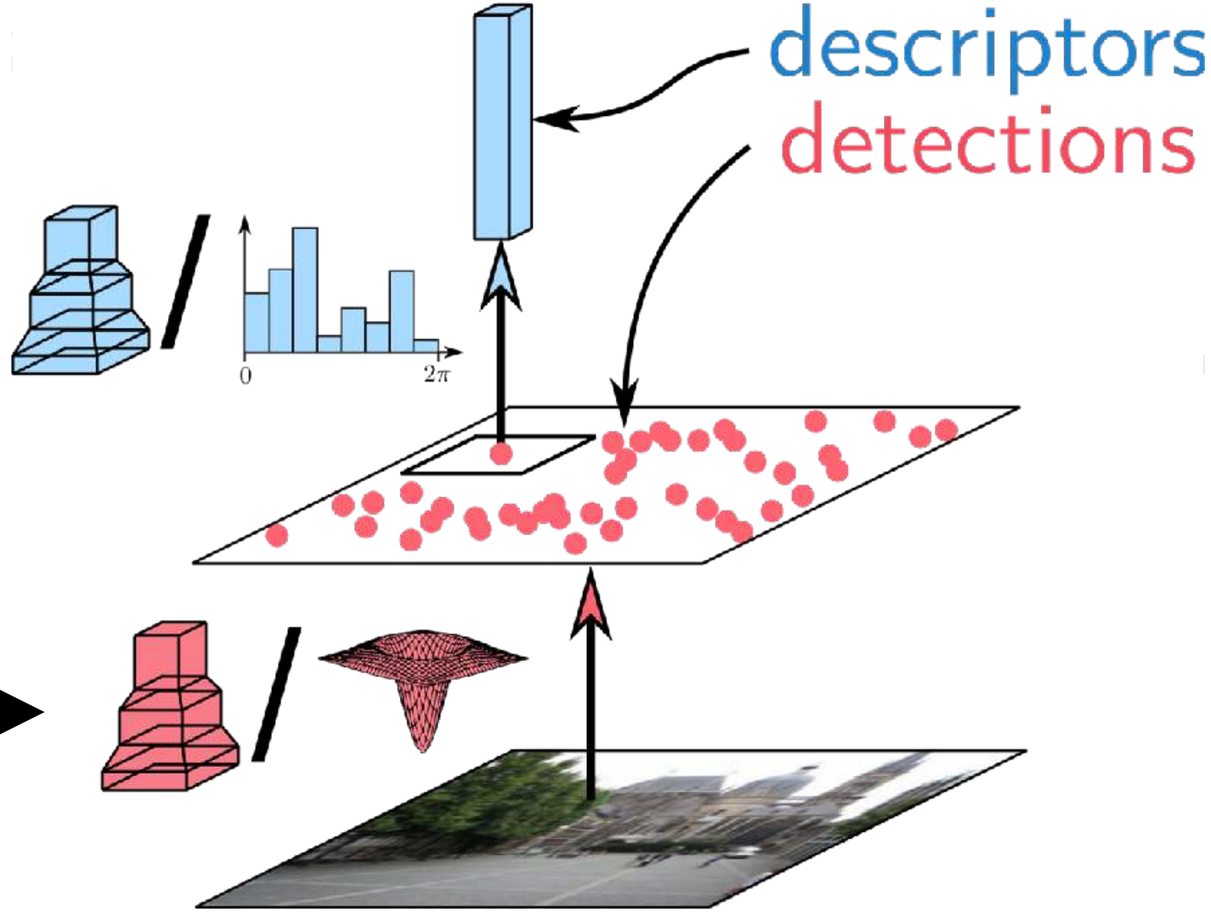
[Schönberger, Hardmeier, Sattler, Pollefeys, Evaluation of Hand-Crafted and Learned Local Features. CVPR 2017]

Learning Features

- Learning Feature Descriptors
- Learning Feature Detectors
- **Joint Feature Detection and Description**

Classical Local Features

Detect-then-Describe:



Efficient, looking at low-level structures / statistics

[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

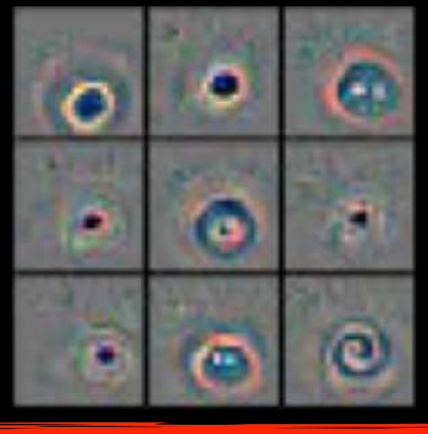
Limits of Local Features

slide credit: Hugo Germanin

CNNs as Object Detectors

Low-Level Features

Mid-Level Features



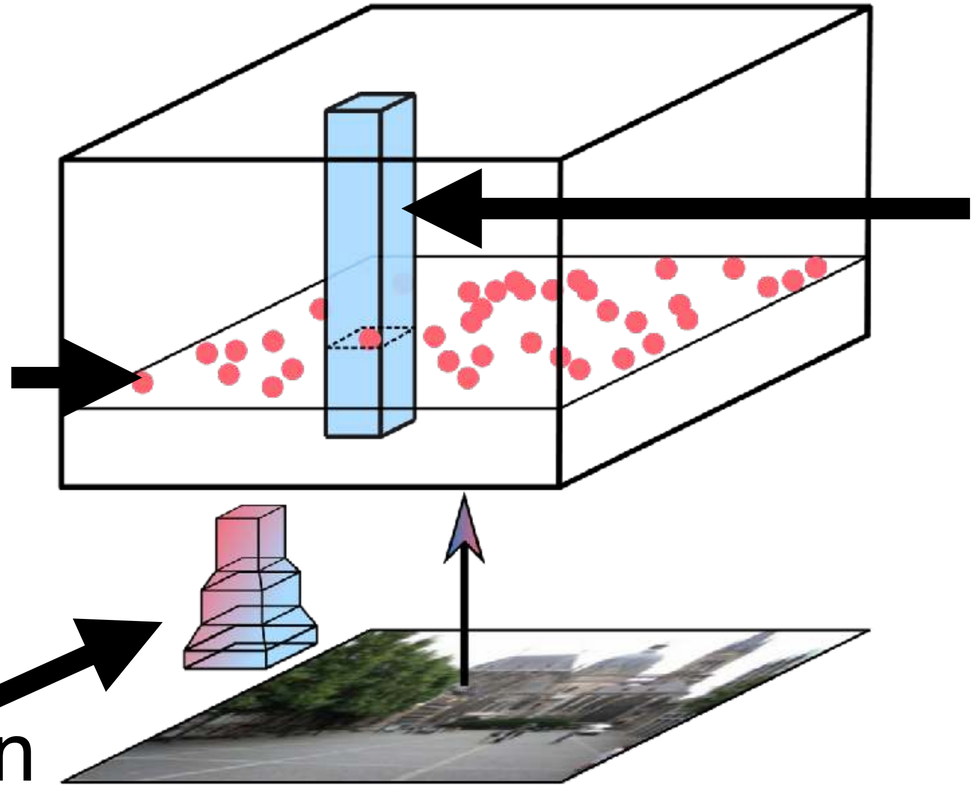
High-Level Features

[Zeiler & Fergus, Visualizing and Understanding Convolutional Networks, ECCV 2014]

Detect-And-Describe Approach

Descriptor ≈
“Objectness” scores

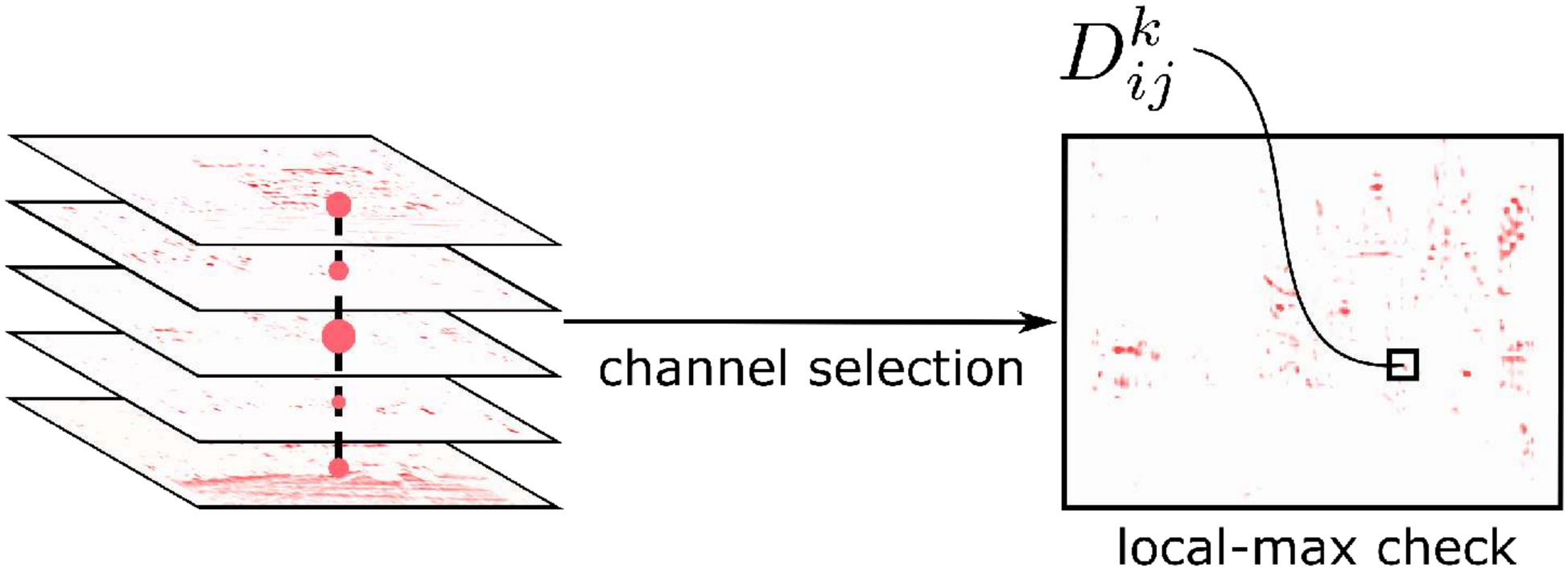
Local maxima ≈
Object detections



Same CNN for
detection & description

[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

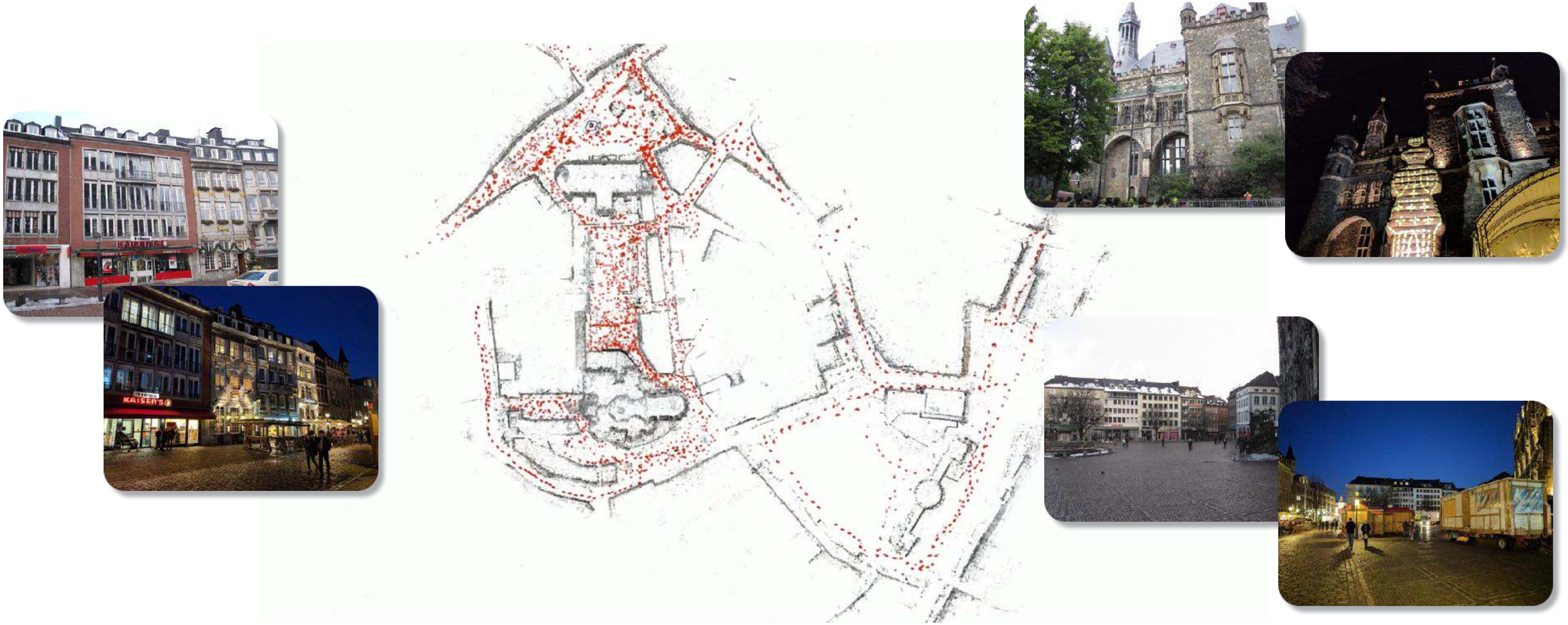
Hard Keypoint Detection



$$(i,j) \text{ is a detection} \iff D^k_{ij} \text{ is a local max. in } D^k \ ,$$
$$\text{with } k = \arg\max_t D^t_{ij} \ .$$

[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

Aachen Day-Night Dataset



[Sattler, Maddern, Toft, Torii, Hammarstrand, Stenborg, Safari, Okutomi, Pollefeys, Sivic, Kahl, Pajdla, Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions, CVPR 2018]

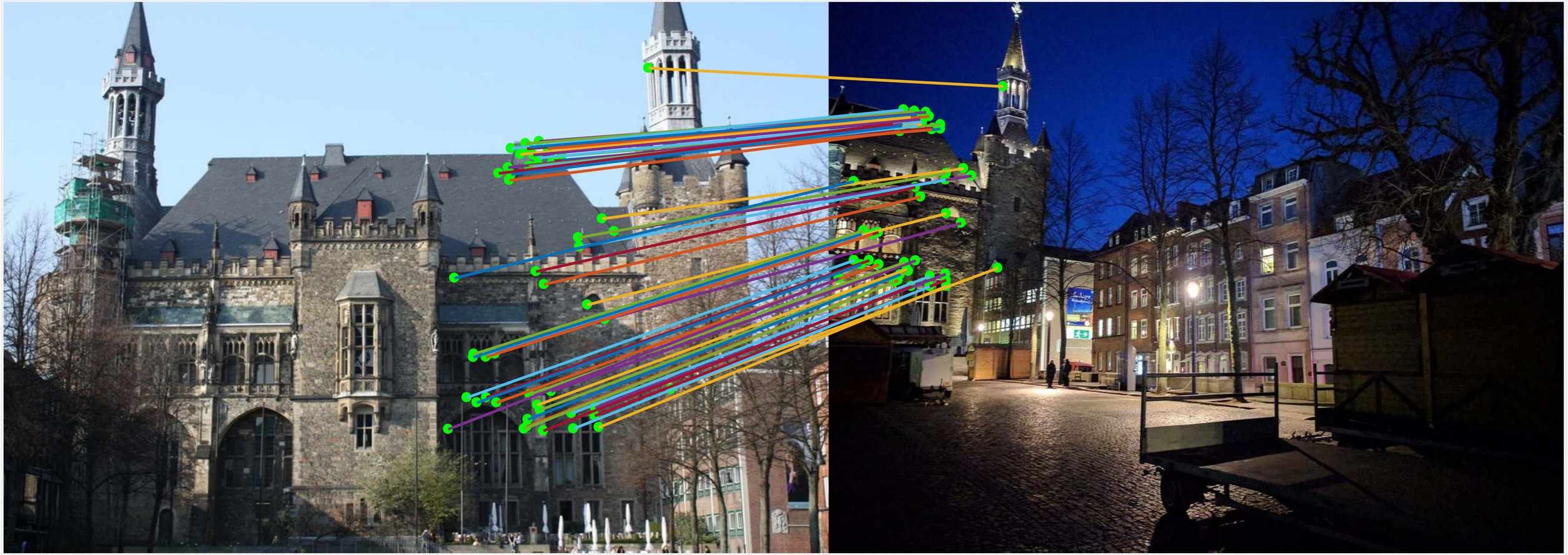
The Good

Aachen Day-Night, network pre-trained on ImageNet

Method	# Features	Correctly localized queries (%)			
		0.5m, 2°	1.0m, 5°	5.0m, 10°	10m, 25°
Upright RootSIFT	11.3K	36.7	54.1	72.5	81.6
DenseSfM	7.5K / 30K	39.8	60.2	84.7	99.0
HessAffNet + HardNet++	11.5K	39.8	61.2	77.6	88.8
SuperPoint	6.6K	42.8	57.1	75.5	86.7
DELF	11K	38.8	62.2	85.7	98.0
D2-Net SS (ours)	7K	41.8	66.3	85.7	98.0
D2-Net MS (ours)	11.4K	43.9	**67.3**	87.8	99.0
D2-Net SS Trained (ours)	14.5K	**44.9**	66.3	**88.8**	**100**
D2-Net MS Trained (ours)	19.3K	**44.9**	64.3	**88.8**	**100**

[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

The Good

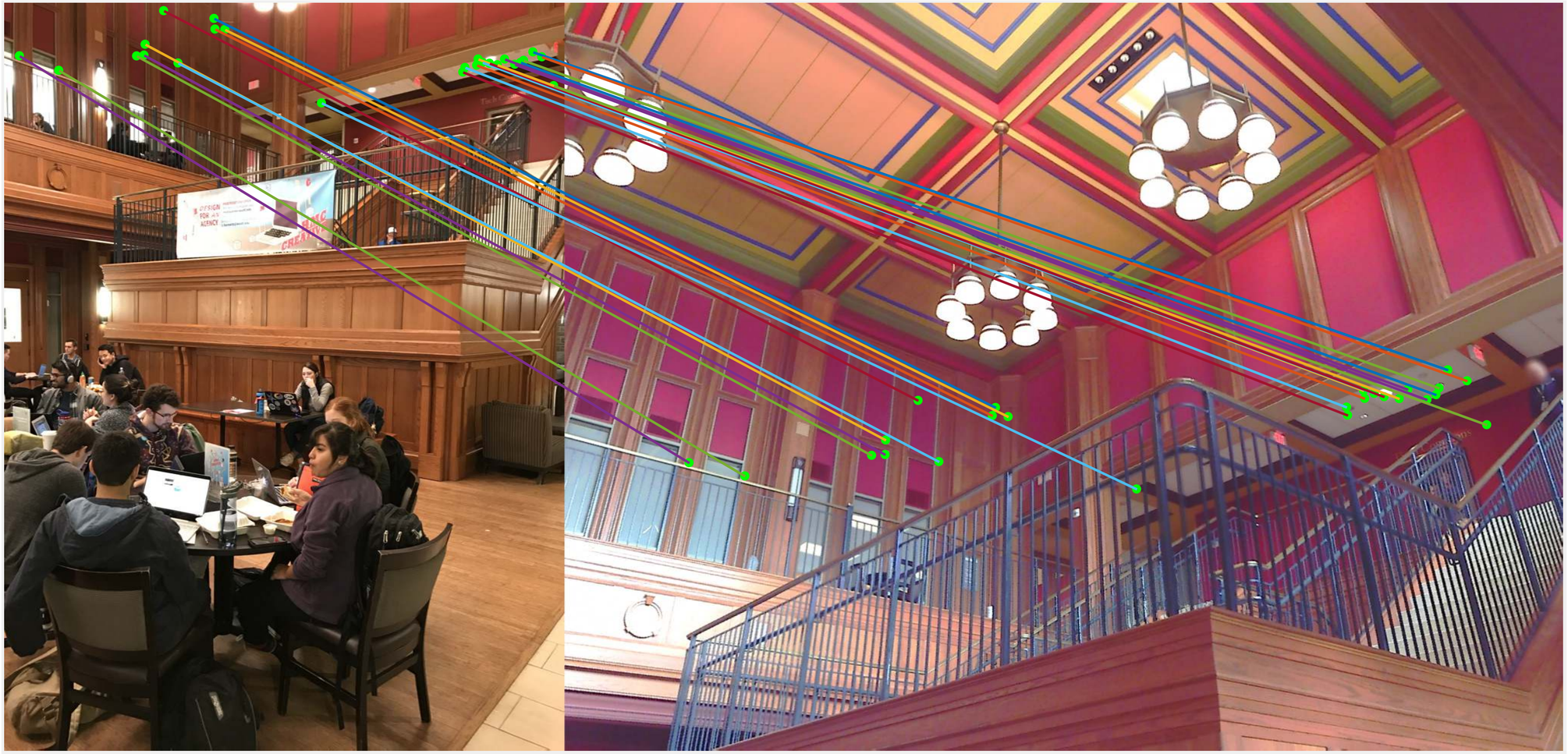


[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

The Good

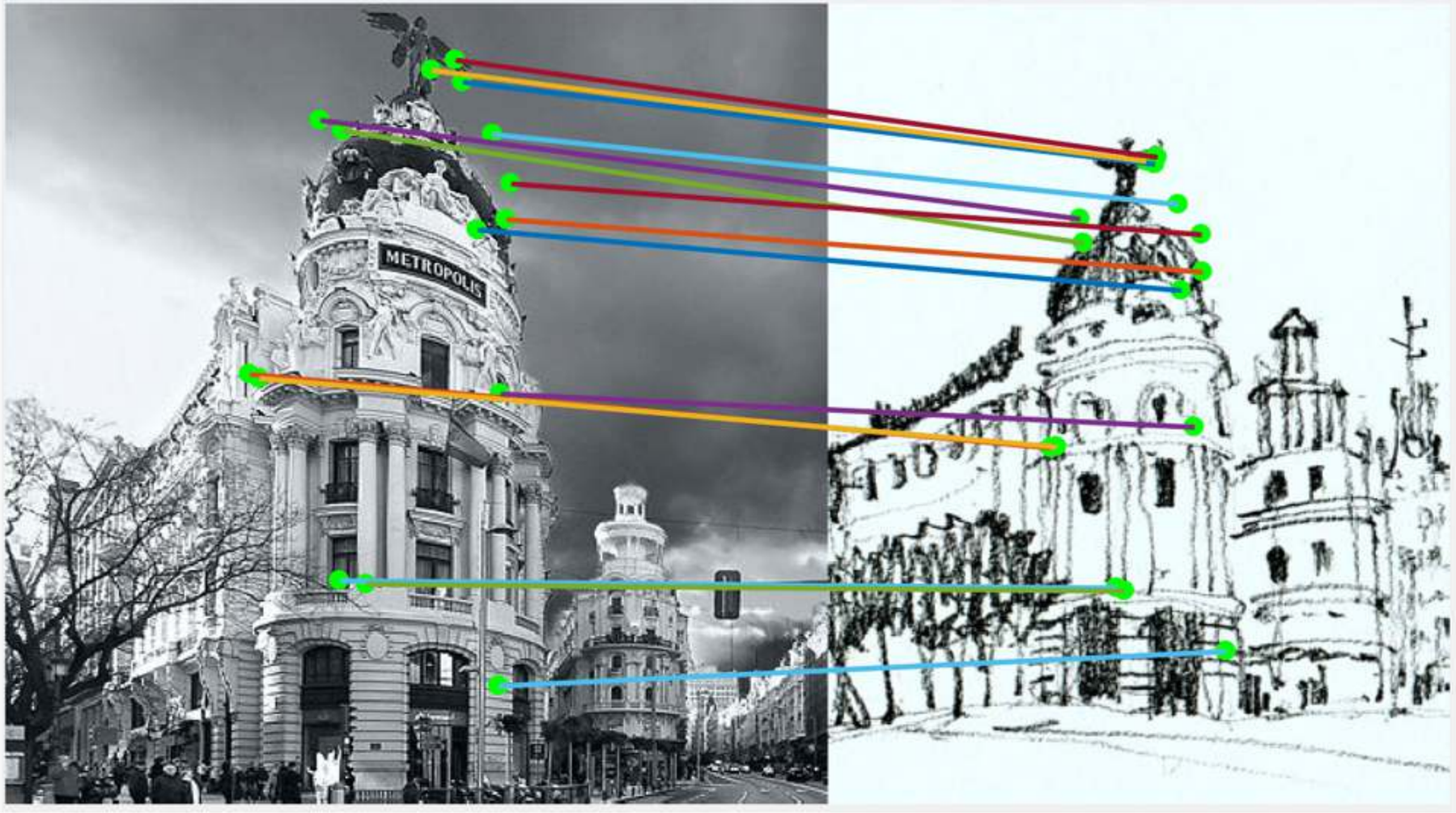
[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

The Good



[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

The Good

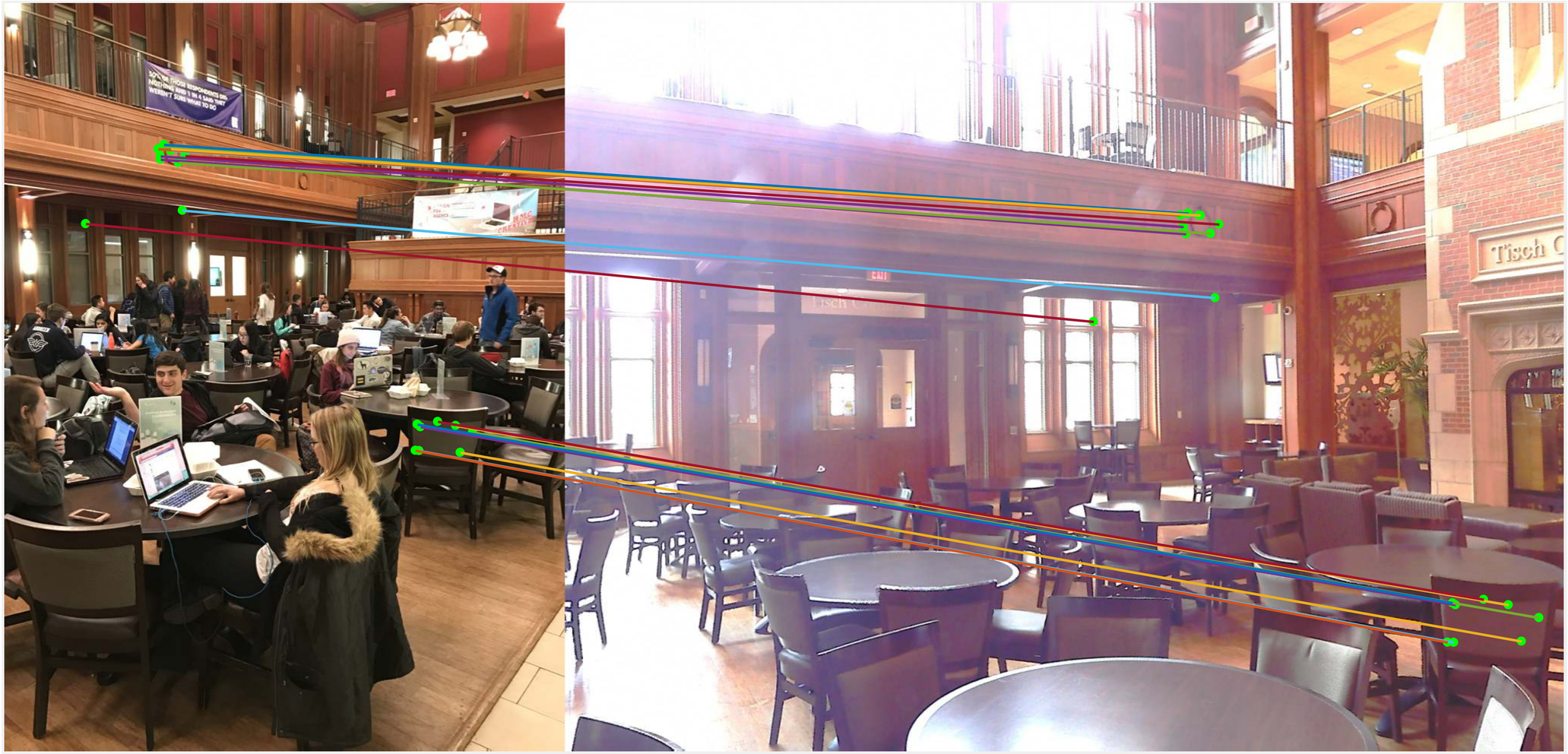


[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

The Ambiguous

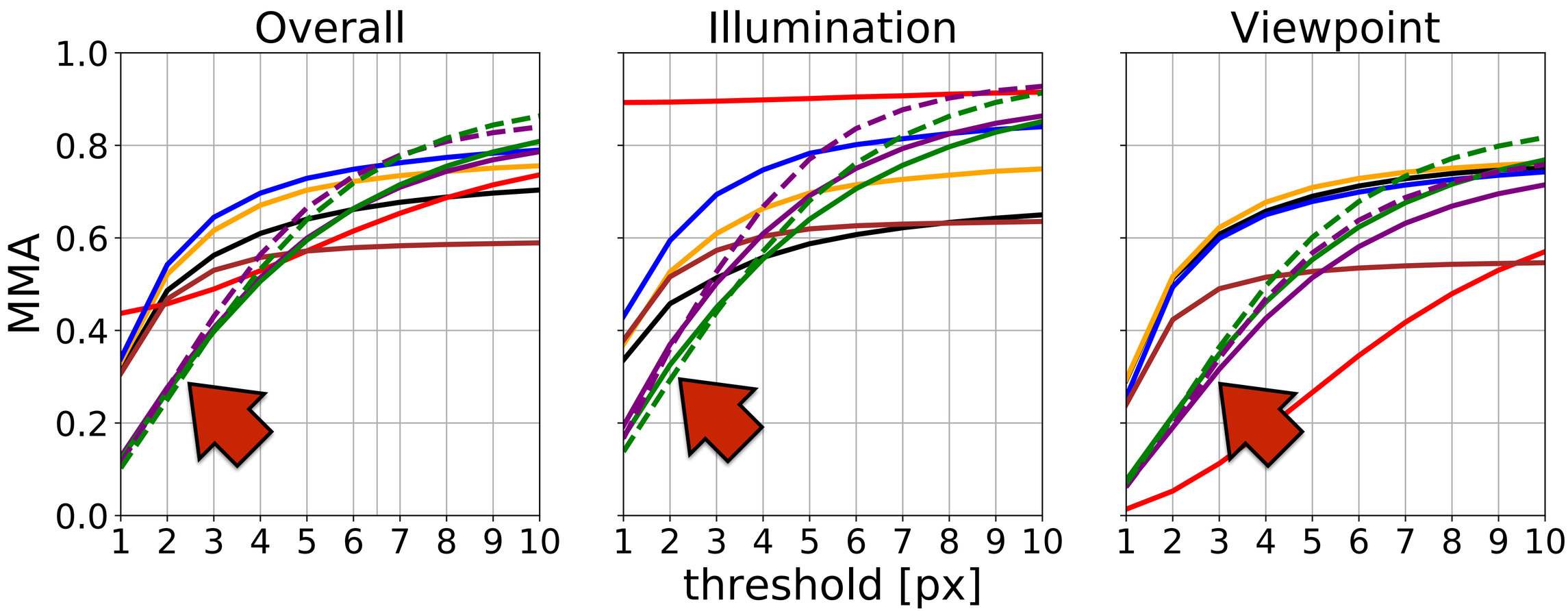
[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

The Ambiguous



[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

The Bad



Our features are less well-localized (max-pooling hurts further)

[Dusmanu, Rocco, Pajdla, Pollefeys, Sivic, Torii, Sattler, D2-Net: A Trainable CNN for Joint Detection and Description of Local Features, CVPR 2019]

My Take / Observations

- Deep Neural Networks trained for other tasks (here: classification) work (surprisingly) very well as feature descriptors
- Features from deeper layers in a network help to handle strong changes in image appearance, e.g., day to night
- One problem: Descriptors can be too general, produce too many false positive matches
- A lot of work left to increase the accuracy of the detected keypoints

Next Lecture

Jan. 20	Introduction, Linear classifiers and filtering	
Jan. 23	Filtering, gradients, scale	Lab 1
Jan. 27	Local features	
Jan. 30	Learning a classifier	Lab 2
Feb. 3	Convolutional neural networks	
Feb. 6	More convolutional neural networks	
Feb. 10	Robust model fitting and RANSAC	Lab 3
Feb. 13	Image registration	
Feb. 17	Camera Geometry	Lab 4
Feb. 20	More camera geometry	
Feb. 24	Generative neural networks	
Feb. 27	Generative neural networks	
Mar. 2	Visual Localization & Feature Learning	
Mar. 9	**No lecture**	