SuperGlue:
Learning Feature Matching with Graph Neural Networks

Paper Authors:

Paul-Edouard Sarlin

Daniel DeTone

Tomasz Malisiewicz

Andrew Rabinovich

Report by Ilia Shipachev

A minimal matching pipeline

SuperGlue: context aggregation + matching + filtering

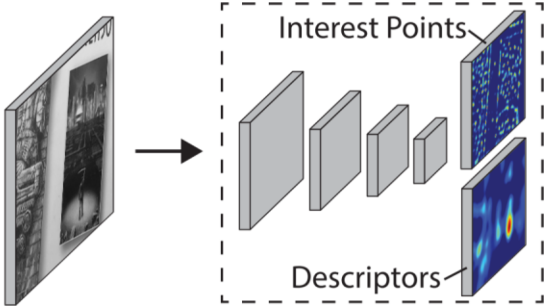
image pair

detection → description → feature matching → outlier filtering → pose estimation

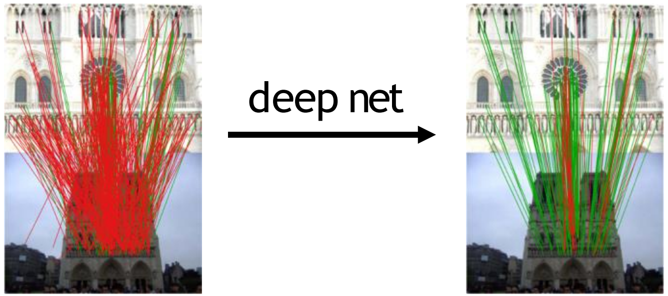
> Classical: SIFT, ORB
> Learned: SuperPoint, D2-Net

Nearest Neighbor Matching

> Heuristics: ratio test, mutual check
> Learned: classifier on set



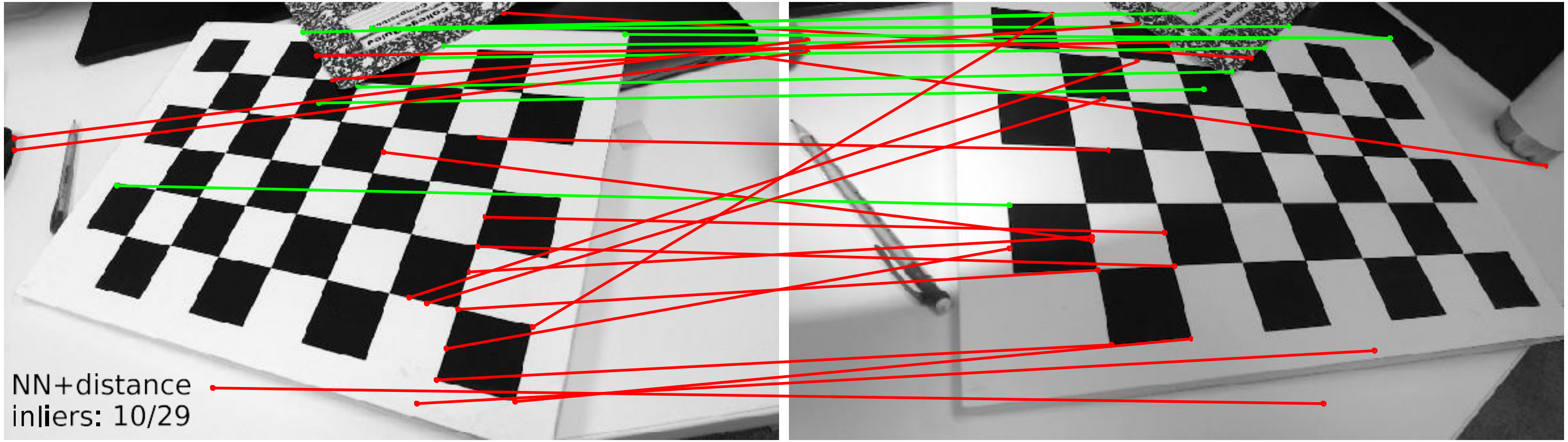
[DeTone et al, 2018]



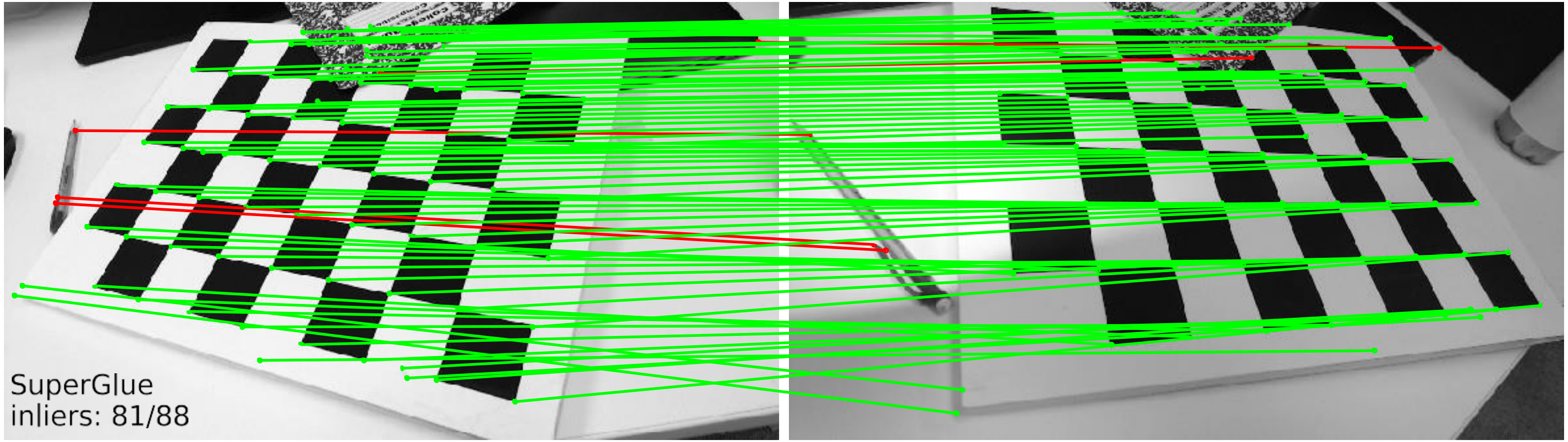
[Yi et al, 2018]

The importance of context

no SuperGlue



with SuperGlue

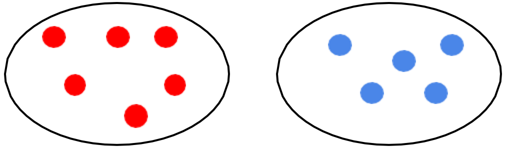


Problem formulation

Inputs → 

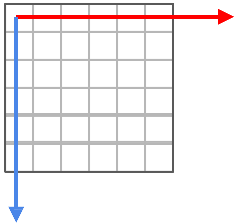
→ **Outputs**

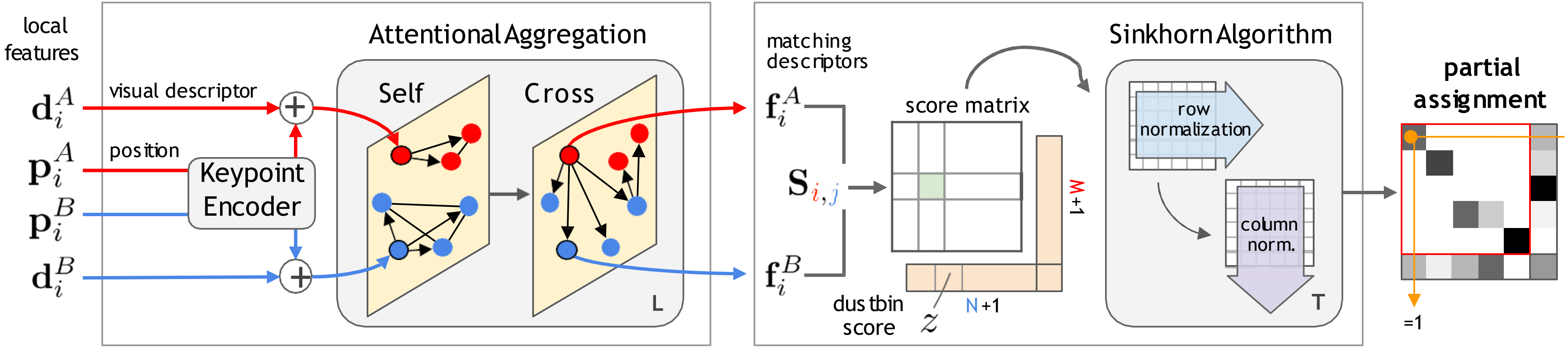
- Images A and B
- **2 sets of M, N local features**
  - Keypoints: $\mathbf{p}_i := (x, y, c)_i$
    - Coordinates (x, y)
    - Confidence c
  - Visual descriptors: $\mathbf{d}_i$



Single a match per keypoint
+ occlusion and noise
→ a **soft partial assignment:**

$$\mathbf{P} \in [0, 1]^{M \times N}$$





A Graph Neural Network with attention

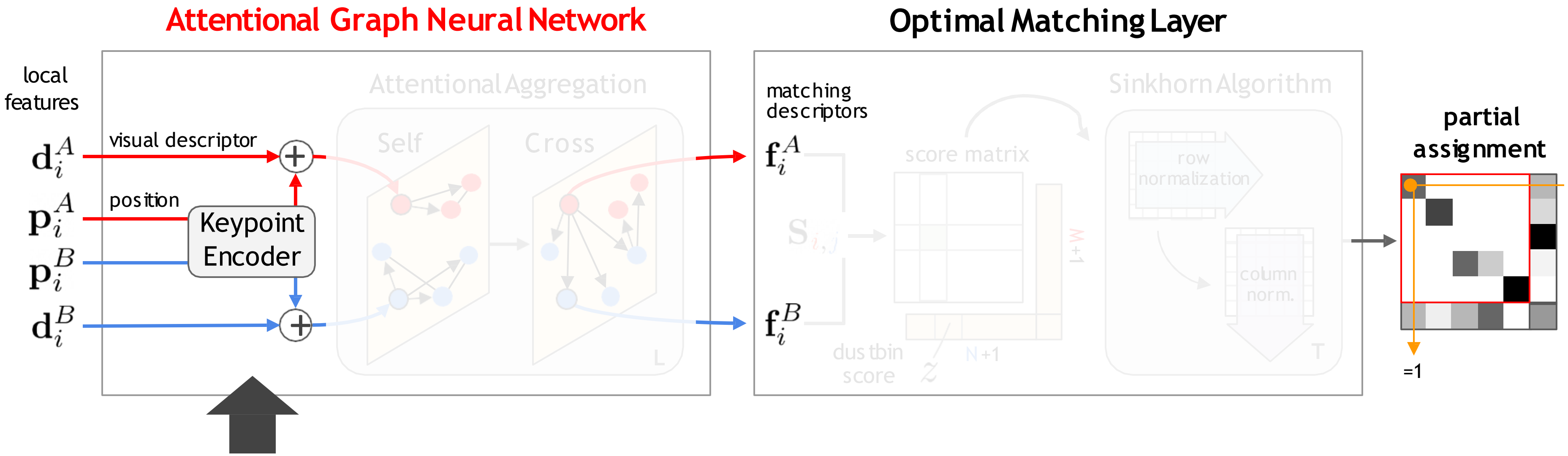
Solving a partial assignment problem

Encodes **contextual cues** & priors

Reasons about the 3D scene

Differentiable **solver**

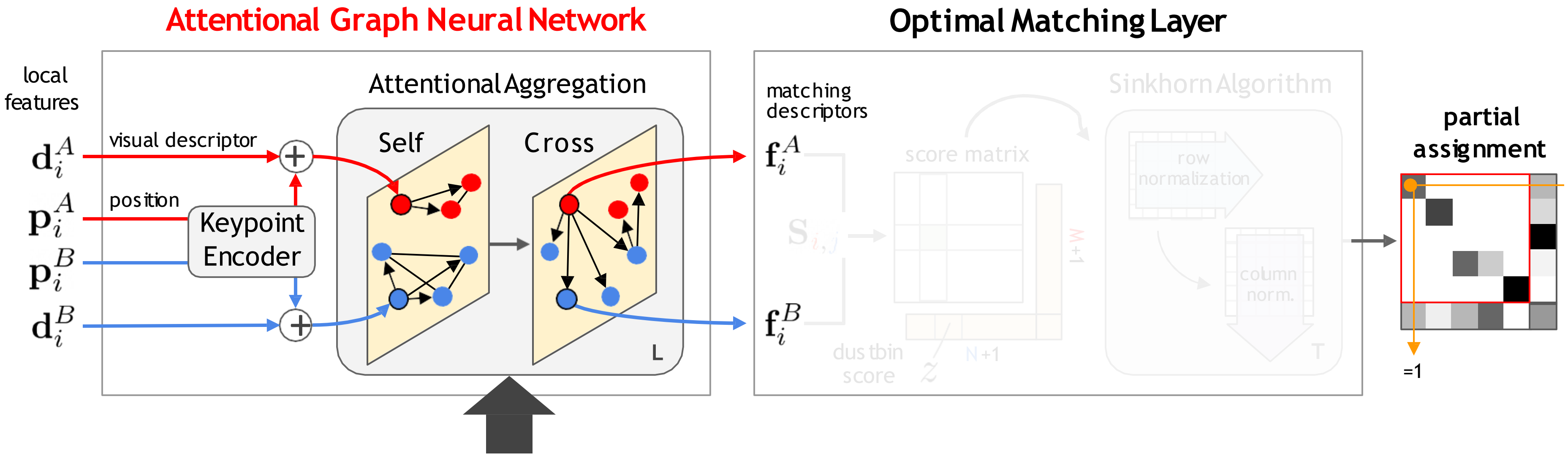
Enforces the assignment constraints = **domain knowledge**



- Initial representation for each keypoints i : ${}^{(0)}\mathbf{x}_i$
- Combines visual appearance and position with an MLP:

$$ {}^{(0)}\mathbf{x}_i = \mathbf{d}_i + \mathrm{MLP}\left(\mathbf{p}_i\right) $$

Multi-Layer Perceptron

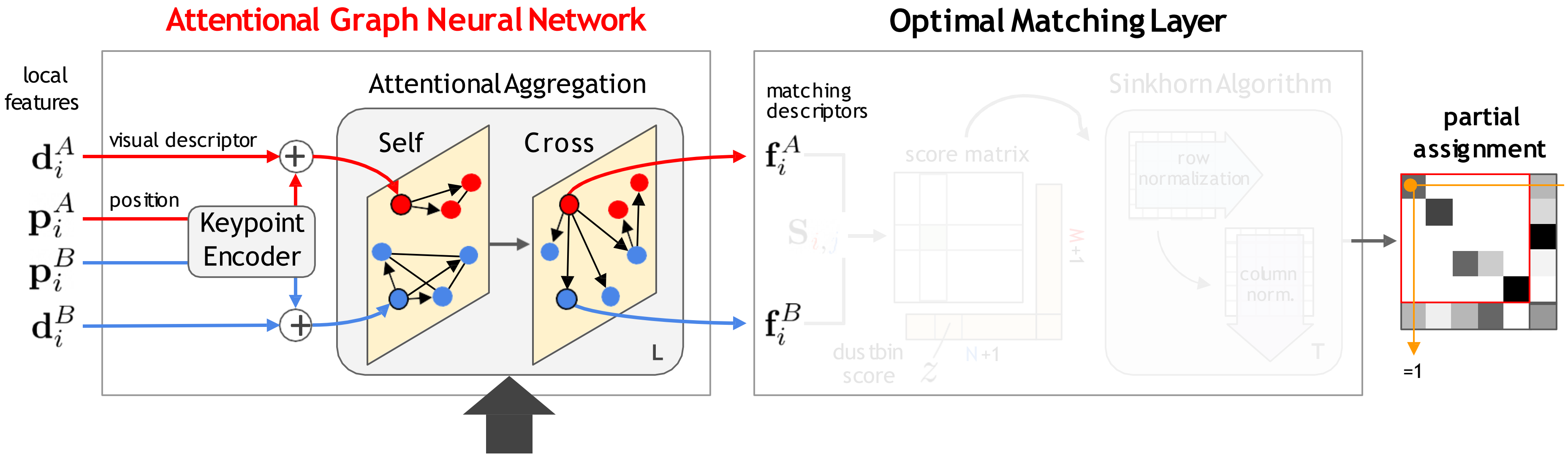


Update the representation based on other keypoints:

- in the same image: "**self**" edges
- in the other image: "**cross**" edges

$$ {}^{(\ell)}\mathbf{x}_i^A \longrightarrow {}^{(\ell+1)}\mathbf{x}_i^A $$

→ A complete **graph** with two types of edges



Update the representation using a **Message Passing Neural Network**

$$^{(\ell+1)}\mathbf{x}_i^A = {}^{(\ell)}\mathbf{x}_i^A + \mathrm{MLP}\left(\left[{}^{(\ell)}\mathbf{x}_i^A \,||\, \mathbf{m}_{\mathcal{E}\to i}\right]\right)$$

the message

Attentional Aggregation

- Compute the **message** $\mathbf{m}_{\mathcal{E}\rightarrow i}$ using **self** and **cross attention**
- Soft database retrieval: query $\mathbf{q}_i$, key $\mathbf{k}_j$, and value $\mathbf{v}_j$

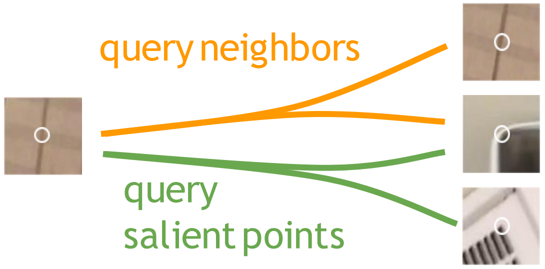
$$\mathbf{m}_{\mathcal{E}\rightarrow i} = \sum_{j:(i,j)\in\mathcal{E}} \alpha_{ij}\mathbf{v}_j$$

$$\alpha_{ij} = \mathrm{Softmax}_j\left(\mathbf{q}_i^\top \mathbf{k}_j\right)$$

$$\mathbf{q}_i = \mathbf{W}_1{}^{(\ell)}\mathbf{x}_i + \mathbf{b}_1$$

$$\begin{bmatrix}\mathbf{k}_j\\ \mathbf{v}_j\end{bmatrix} = \begin{bmatrix}\mathbf{W}_2\\ \mathbf{W}_3\end{bmatrix}{}^{(\ell)}\mathbf{x}_j + \begin{bmatrix}\mathbf{b}_2\\ \mathbf{b}_3\end{bmatrix}$$

$\mathbf{x}_i$ = [tile, position (70, 100)]



= [tile, pos. (80, 110)]

= [corner, pos. (60, 90)]

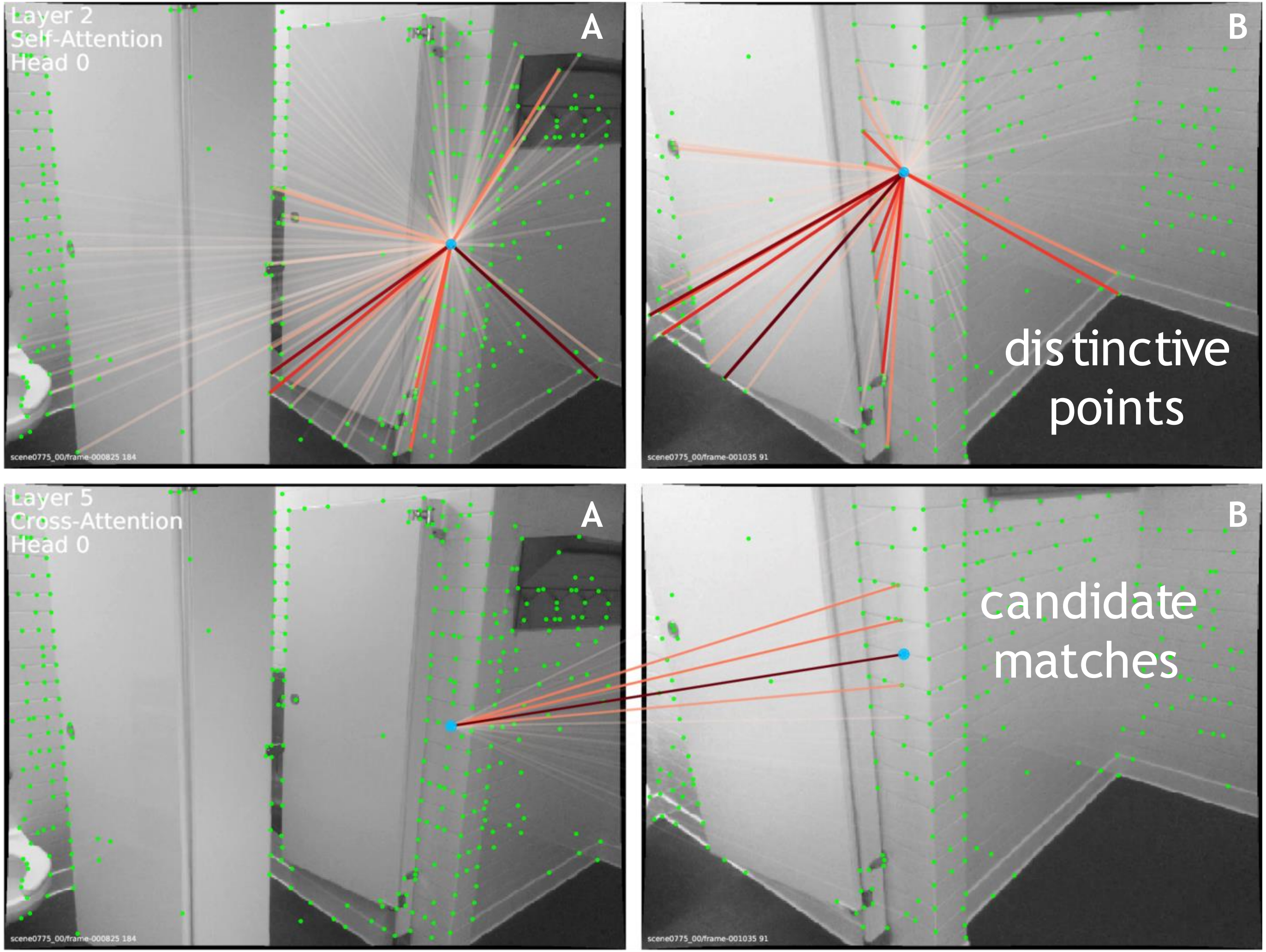
= [grid, pos. (400, 600)]

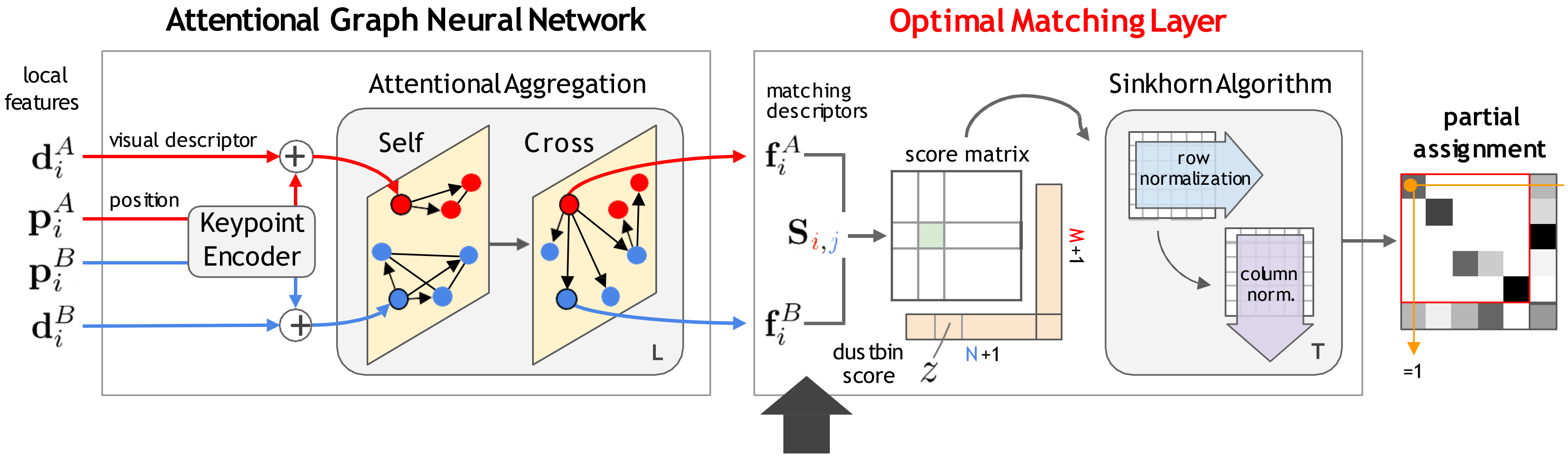
[Vaswani et al, 2017]

Self-attention
= intra-image information flow

Cross-attention
= inter-image

Attention builds a soft, dynamic, sparse graph

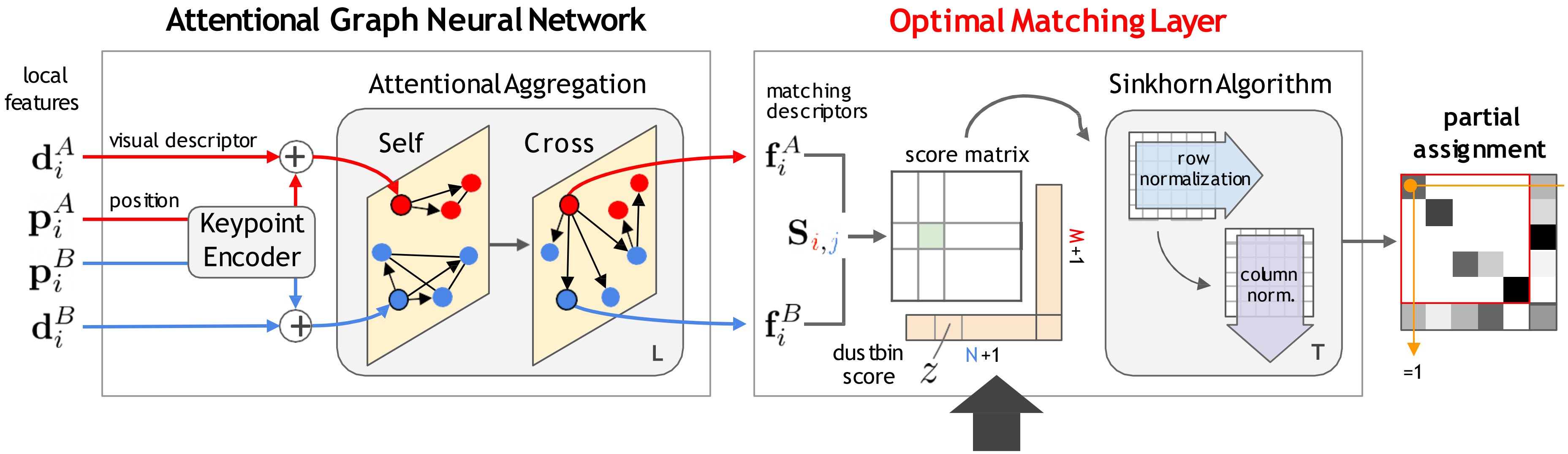




Compute a **score matrix** $\mathbf{S} \in \mathbb{R}^{M \times N}$ for all matches:

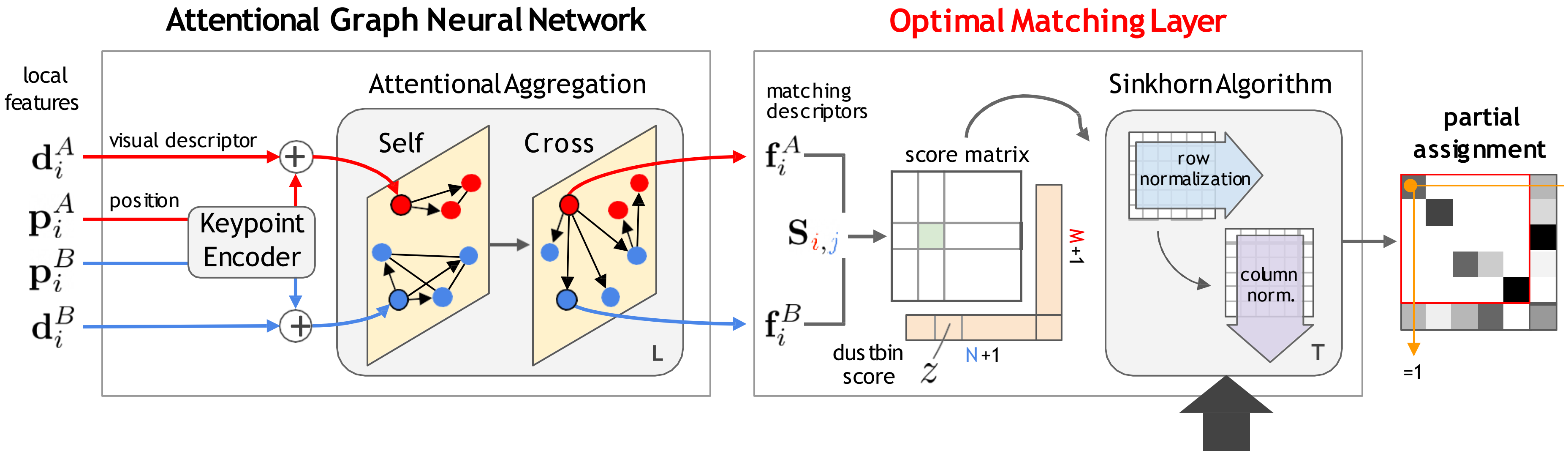
$$\mathbf{f}_i^A = \mathbf{W} \cdot {}^{(L)}\mathbf{x}_i^A + \mathbf{b}$$

$$\mathbf{S}_{i,j} = < \mathbf{f}_i^A, \mathbf{f}_j^B >$$



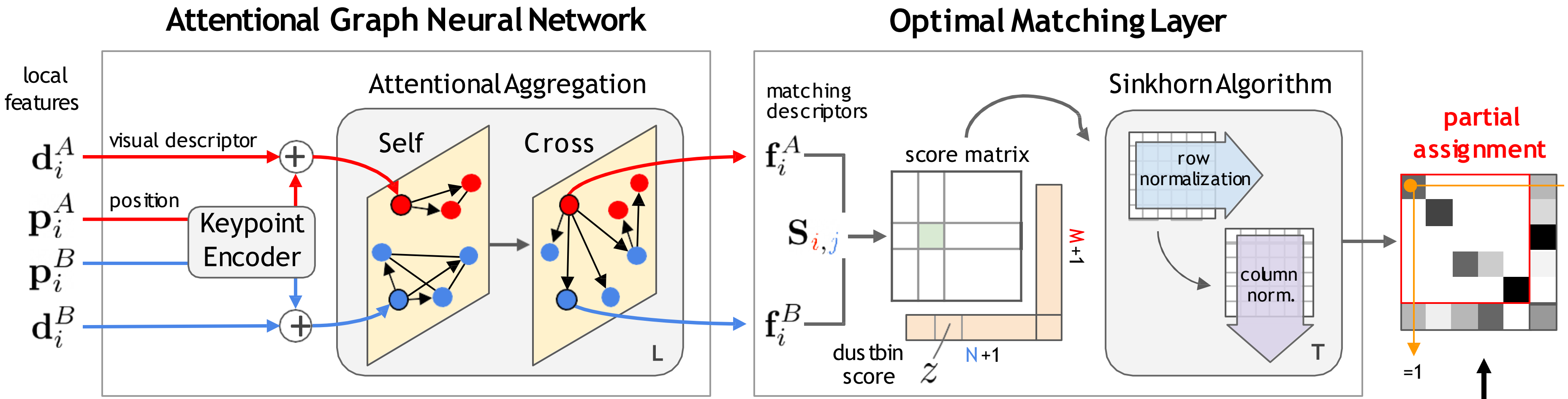
- Occlusion and noise: unmatched keypoints are assigned to a **dustbin**
- **Augment** the scores with a learnable dustbin score z

$$\bar{\mathbf{S}}_{i,N+1} = \bar{\mathbf{S}}_{M+1,j} = \bar{\mathbf{S}}_{M+1,N+1} = z \in \mathbb{R}$$



- Compute the assignment $\bar{\mathbf{P}}$ that maximizes $\sum_{i,j} \bar{\mathbf{S}}_{i,j}\bar{\mathbf{P}}_{i,j}$
- Solve an **optimal transport** problem
- With the **Sinkhorn algorithm:** differentiable & soft Hungarian algorithm

[Sinkhorn & Knopp, 1967]



- Compute **ground truth correspondences** from pose and depth
- Find which keypoints should be **unmatched**
- Loss: maximize the log-likelihood $\bar{\mathbf{P}}_{i,j}$ of the GT cells

Loss function

$\mathcal{M} = \{(i, j)\} \subset \mathcal{A} \times \mathcal{B}$ - set of GT matches

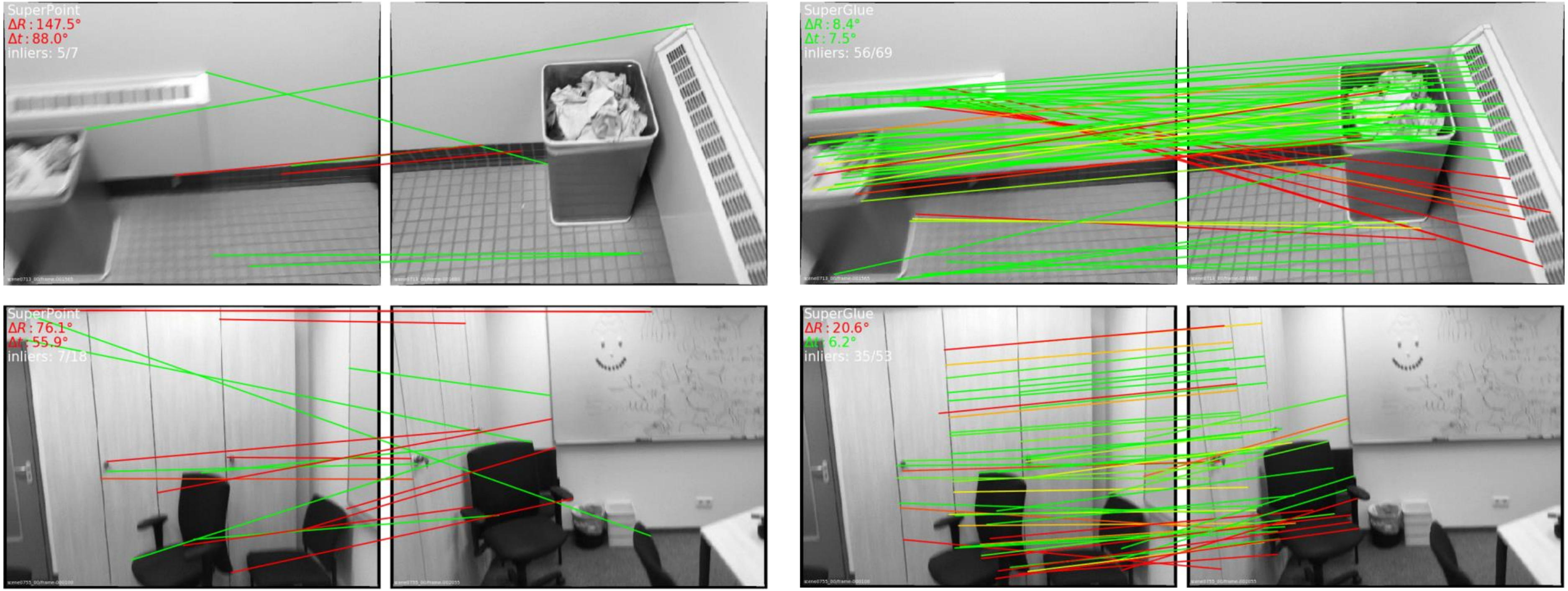
$\mathcal{I} \subseteq \mathcal{A}$ and $\mathcal{J} \subseteq \mathcal{B}$ - set of unmached points in GT

$$\text{Loss} = -\sum_{(i,j)\in\mathcal{M}} \log \bar{\mathbf{P}}_{i,j} - \sum_{i\in\mathcal{I}} \log \bar{\mathbf{P}}_{i,N+1} - \sum_{j\in\mathcal{J}} \log \bar{\mathbf{P}}_{M+1,j}$$

Results: indoor -ScanNet

SuperPoint + NN + heuristics

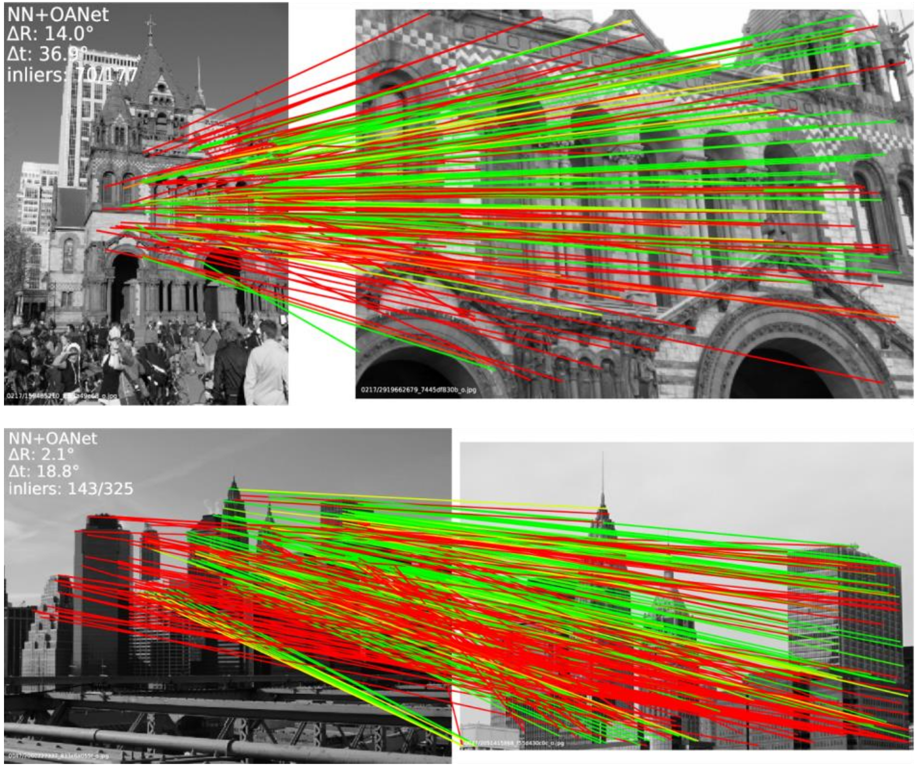
SuperPoint + **SuperGlue**



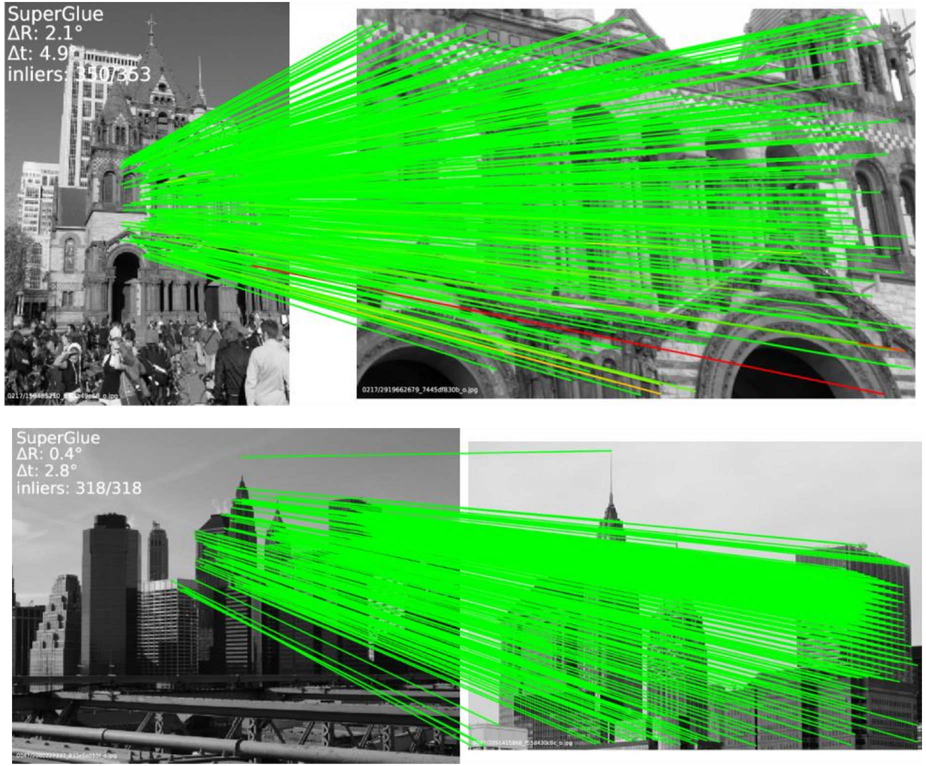
SuperGlue: more correct matches and fewer mismatches

Results: outdoor -SfM

SuperPoint + NN + OA-Net (inlier classifier)



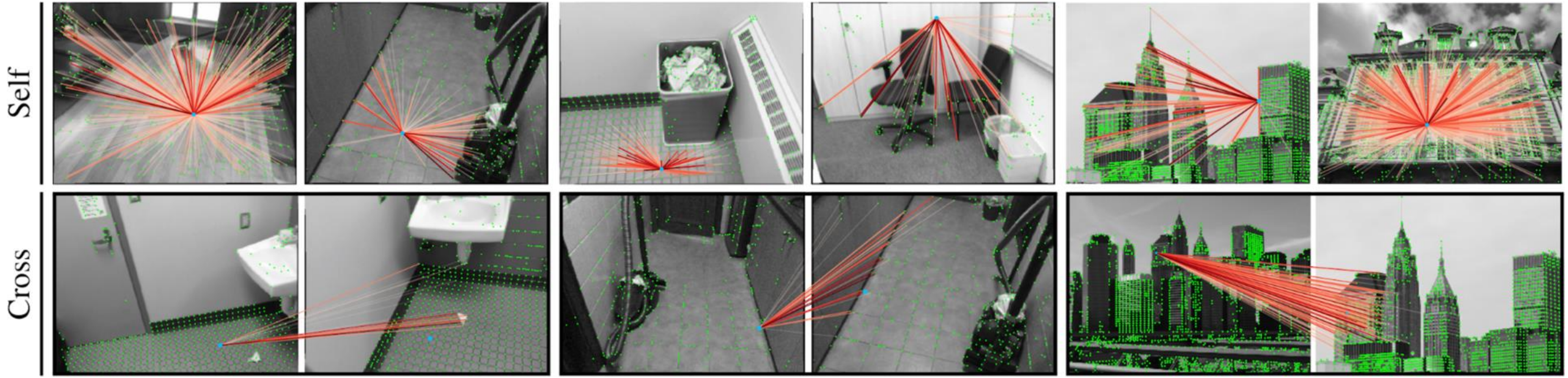
SuperPoint + **SuperGlue**



SuperGlue: more correct matches and fewer mismatches

Results: attention patterns

global context neighborhood distinctive keypoints self-similarities



match candidates

Flexibility of attention → **diversity of patterns**

Homography estimation

Local features	Matcher	Homography estimation AUC		P	R
		RANSAC	DLT		
SuperPoint	NN	39.47	0.00	21.7	65.4
	NN + mutual	42.45	0.24	43.8	56.5
	NN + PointCN	43.02	45.40	76.2	64.2
	NN + OANet	44.55	52.29	82.8	64.7
	SuperGlue	**53.67**	**65.85**	**90.7**	**98.3**

Indoor pose estimation

Local features	Matcher	Pose estimation AUC @5°	@10°	@20°	P	MS
ORB	NN + GMS	5.21	13.65	25.36	72.0	5.7
D2-Net	NN + mutual	5.25	14.53	27.96	46.7	12.0
ContextDesc	NN + ratio test	6.64	15.01	25.75	51.2	9.2
SIFT	NN + ratio test	5.83	13.06	22.47	40.3	1.0
	NN + NG-RANSAC	6.19	13.80	23.73	61.9	0.7
	NN + OANet	6.00	14.33	25.90	38.6	4.2
	SuperGlue	**6.71**	**15.70**	**28.67**	**74.2**	**9.8**
SuperPoint	NN + mutual	9.43	21.53	36.40	50.4	18.8
	NN + distance + mutual	9.82	22.42	36.83	63.9	14.6
	NN + GMS	8.39	18.96	31.56	50.3	19.0
	NN + PointCN	11.40	25.47	41.41	71.8	25.5
	NN + OANet	11.76	26.90	43.85	74.0	25.7
	SuperGlue	**16.16**	**33.81**	**51.84**	**84.4**	**31.5**

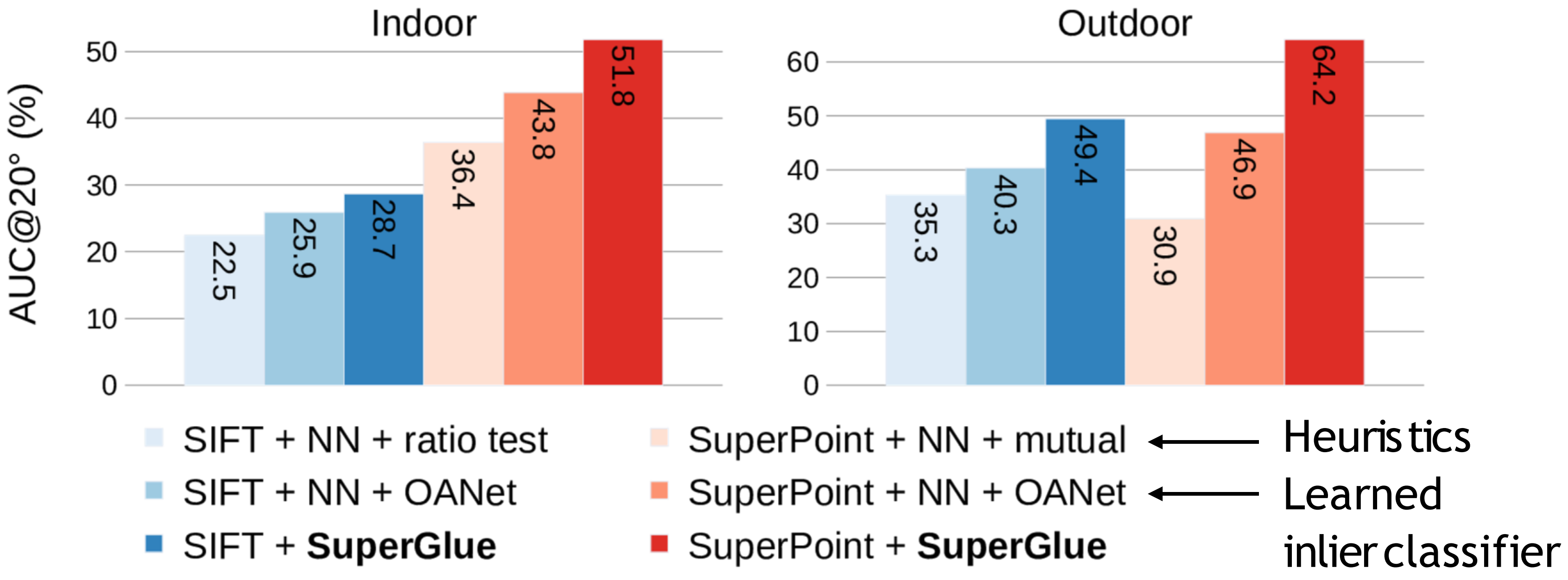
Outdoor pose estimation

Local features	Matcher	Pose estimation AUC			P	MS
		@5°	@10°	@20°		
ContextDesc	NN + ratio test	20.16	31.65	44.05	56.2	3.3
SIFT	NN + ratio test	15.19	24.72	35.30	43.4	1.7
	NN + NG-RANSAC	15.61	25.28	35.87	64.4	1.9
	NN + OANet	18.02	28.76	40.31	55.0	3.7
	SuperGlue	**23.68**	**36.44**	**49.44**	**74.1**	**7.2**
SuperPoint	NN + mutual	9.80	18.99	30.88	22.5	4.9
	NN + GMS	13.96	24.58	36.53	47.1	4.7
	NN + OANet	21.03	34.08	46.88	52.4	8.4
	SuperGlue	**34.18**	**50.32**	**64.16**	**84.9**	**11.1**

Ablation of SuperGlue

Matcher		Pose AUC@20°	Match precision	Matching score
NN + mutual		36.40	50.4	18.8
SuperGlue	No Graph Neural Net	38.56	66.0	17.2
	No cross-attention	42.57	74.0	25.3
	No positional encoding	47.12	75.8	26.6
	Smaller (3 layers)	46.93	79.9	30.0
	Full (9 layers)	**51.84**	**84.4**	**31.5**

Evaluation



SuperGlue yields **large improvements** in all cases