Improving Semantic Correspondence with Viewpoint-Guided Spherical Maps

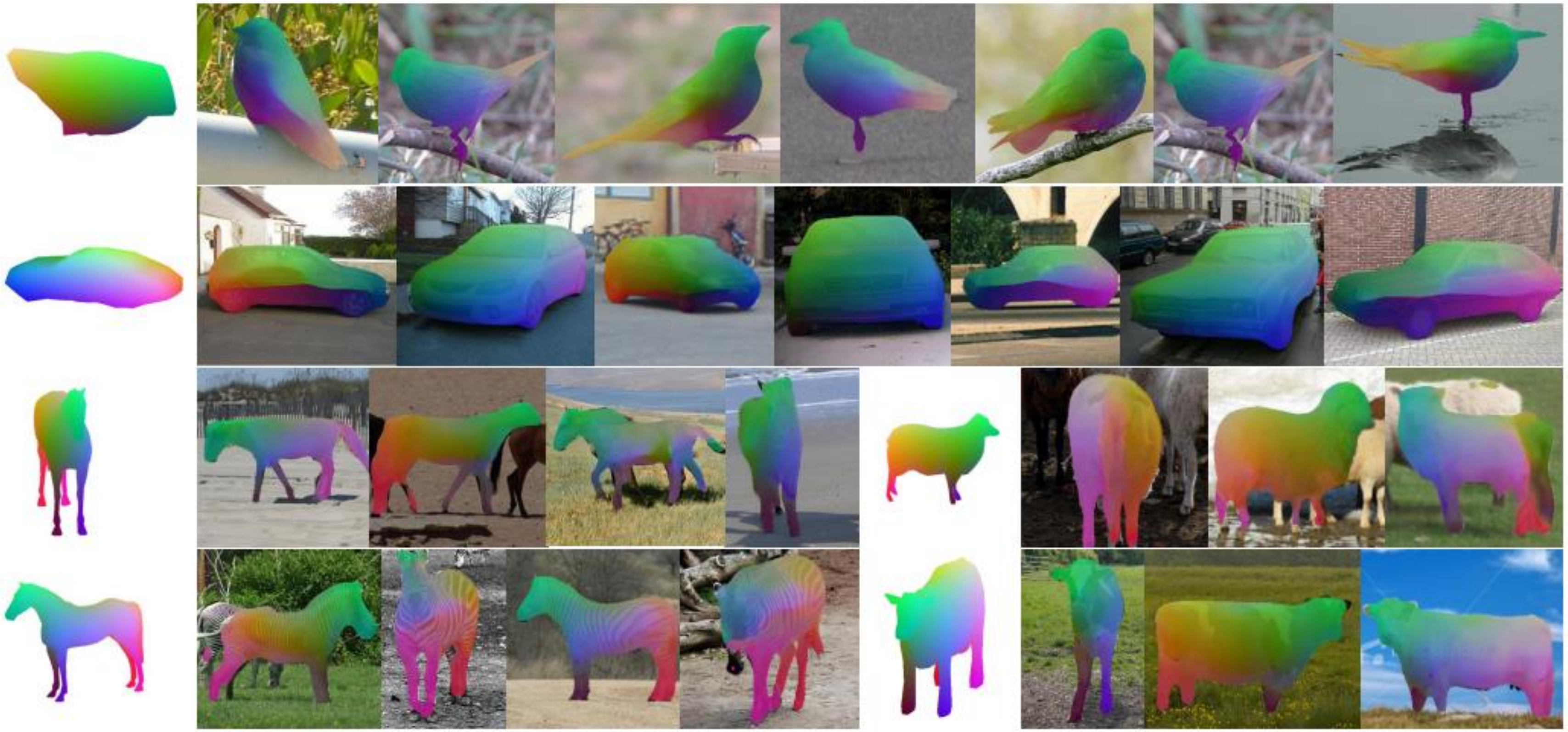
Octave Mariotti   Oisin Mac   Aodha Hakan Bilen

University of Edinburgh

CVPR 2024 (Oral)

Task: Dense Visual Semantic Correspondence

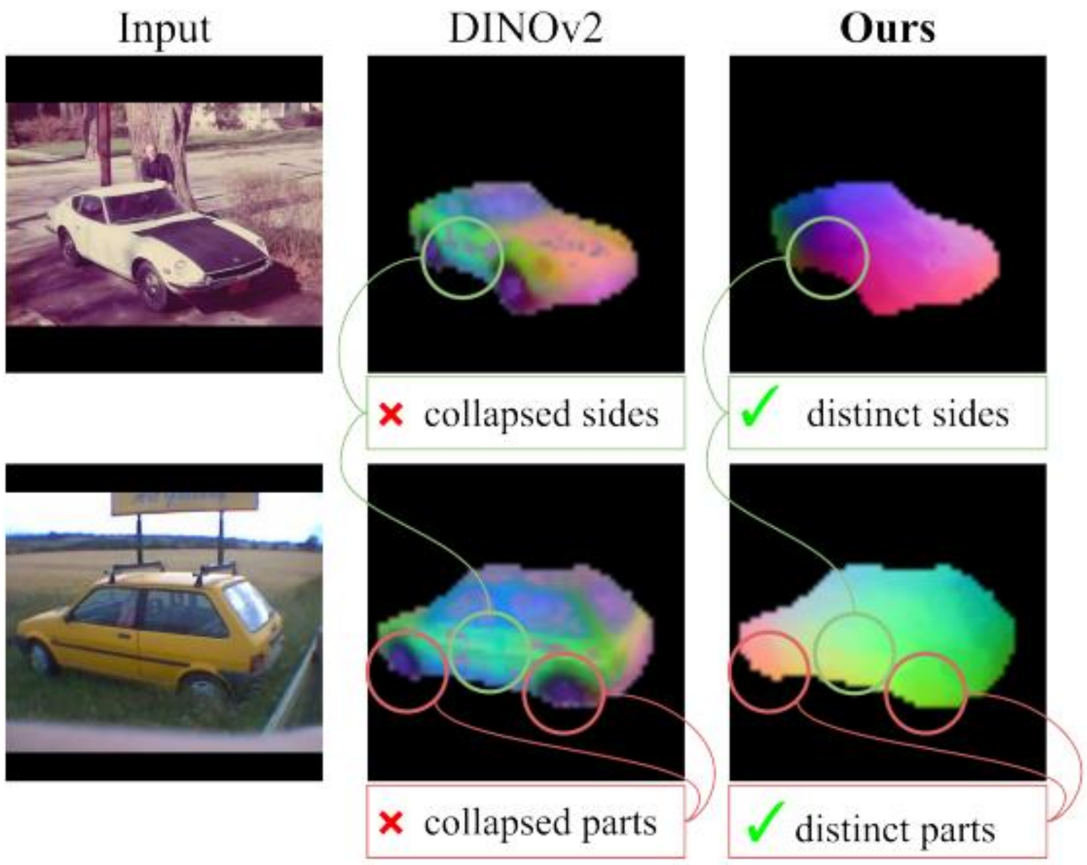
Semantic correspondence (SC) estimation aims to find local regions that correspond to the same semantic entities across a collection of images , where each image contains a different instance of the same object category.



A related task: canonical surface mapping [1]

[1] Kulkarni, Nilesh, Abhinav Gupta, and Shubham Tulsiani. "Canonical surface mapping via geometric cycle consistency.“ ICCV. 2019.

Limitations of SSL-based Correspondence Matching



Limitations:

Current models are typically trained only on 2D images, **they are not able to learn 3D-aware representations**, and often converge to similar features for object parts that share appearance but not fine-grained semantics. (2 main limitations, **symmetries** and **repeated parts**)

(i) they fail to correctly distinguish object symmetries, e.g. the left and right side of the car have the same features

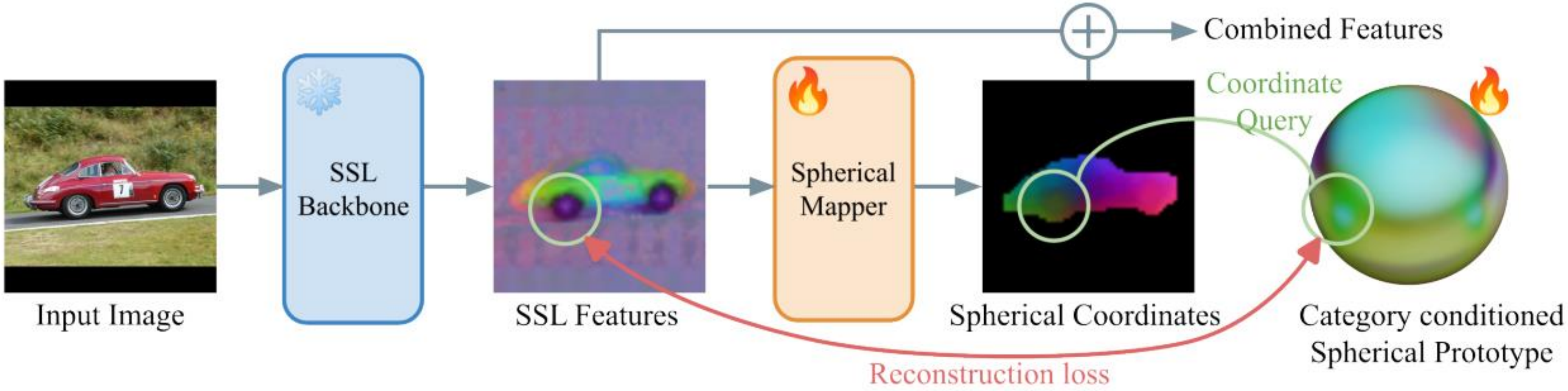
(ii) they struggle to distinguish individual parts, e.g. the wheels are represented by the same features irrespective of their location on the car.

How to address above limitations without:

- re-training SSL model
- Using groundtruth correspondences

Solution:

Introducing explicit 3D knowledge via a weak geometric spherical prior



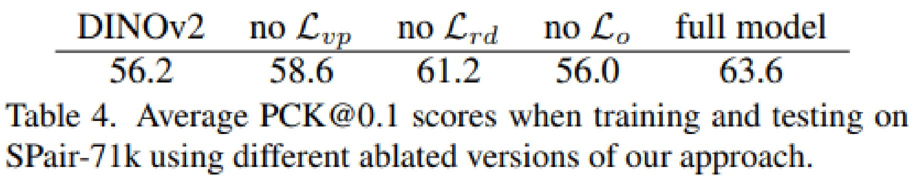
$\phi: (\boldsymbol{I}, \mathrm{x}) \rightarrow z \in Z \subset \mathbb{R}^d$ ϕ is a frozen SSL backbone model (DINOv2 here)

$f_S: (\boldsymbol{I}, \mathrm{x}) \rightarrow s \in S^2 \subset \mathbb{R}^3$ f_S is the spherical Mapper $f_S = f'_S \circ \phi.$

$S_z: (\mathrm{s}, \mathrm{c}) \rightarrow z \in Z \subset \mathbb{R}^d$ S_z is a joint learned spherical prototype, c is a category token

$$\begin{tikzcd} I \arrow[r, "\Phi"] & z \arrow[r, "f_s"'] & S \arrow[l, bend right, "S_z"'] \end{tikzcd}$$

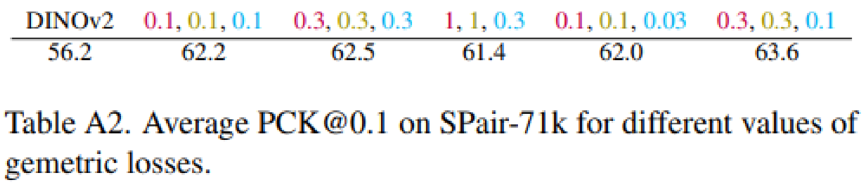
Training



DINOv2	no $\mathcal{L}_{vp}$	no $\mathcal{L}_{rd}$	no $\mathcal{L}_{o}$	full model
56.2	58.6	61.2	56.0	63.6

Table 4. Average PCK@0.1 scores when training and testing on SPair-71k using different ablated versions of our approach.

$$L = L_{rec} + 0.3L_{rd} + 0.3L_{o} + 0.1L_{vp}$$



DINOv2	0.1, 0.1, 0.1	0.3, 0.3, 0.3	1, 1, 0.3	0.1, 0.1, 0.03	0.3, 0.3, 0.1
56.2	62.2	62.5	61.4	62.0	63.6

Table A2. Average PCK@0.1 on SPair-71k for different values of gemetric losses.

Reconstruction Loss

Instance masks

Cosine distance

$$\mathcal{L}_{rec} = \frac{1}{|\Lambda|}\sum_{x\in\Lambda} M(\boldsymbol{I},x) \times \Gamma\left(\phi(\boldsymbol{I},x), S_{\mathcal{Z}}(f_S(\boldsymbol{I},x))\right),$$

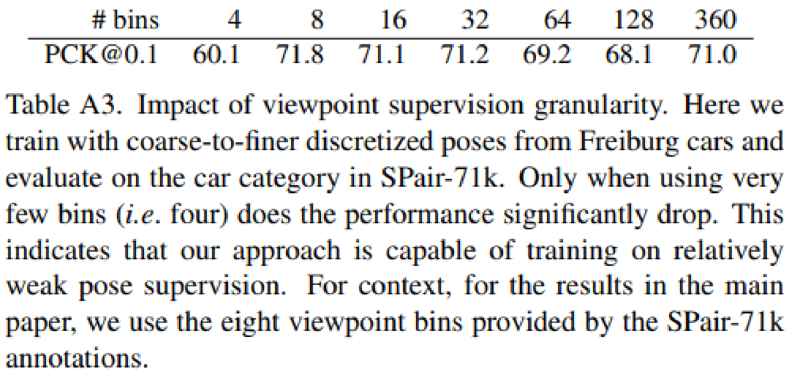
Viewpoint regularization

Assumption:
The average coordinate of a spherical map of an image I can be viewed as a coarse approximation of the camera viewpoint under which the object is seen.

$$\mathcal{L}_{vp} = \sum_{I,I'} ||v_I \cdot v_{I'} - \mu(f_S(\boldsymbol{I})) \cdot \mu(f_S(\boldsymbol{I}'))||^2.$$

Gt camera viewpoint

Average direction



# bins	4	8	16	32	64	128	360
PCK@0.1	60.1	71.8	71.1	71.2	69.2	68.1	71.0

Table A3. Impact of viewpoint supervision granularity. Here we train with coarse-to-finer discretized poses from Freiburg cars and evaluate on the car category in SPair-71k. Only when using very few bins (*i.e.* four) does the performance significantly drop. This indicates that our approach is capable of training on relatively weak pose supervision. For context, for the results in the main paper, we use the eight viewpoint bins provided by the SPair-71k annotations.

In practice, v_I is discretized among a small number of bins, representing azimuth angle

Training

Relative distance loss

Assumption $\|a-b\| \leq \|a-c\| \iff \Gamma(s_a, s_b) \leq \Gamma(s_a, s_c)$

$$\begin{aligned}\mathcal{L}_{rd} = \max(&\Gamma(f_S(\boldsymbol{I}, anc), f_S(\boldsymbol{I}, pos)) \\ &-\Gamma(f_S(\boldsymbol{I}, anc), f_S(\boldsymbol{I}, neg)) + \delta, 0),\end{aligned} \qquad \delta = 0.5$$

$anc = a \qquad pos = argmin_{x\in\{b,c\}}\|a-x\| \qquad neg = argmax_{x\in\{b,c\}}\|a-x\|$

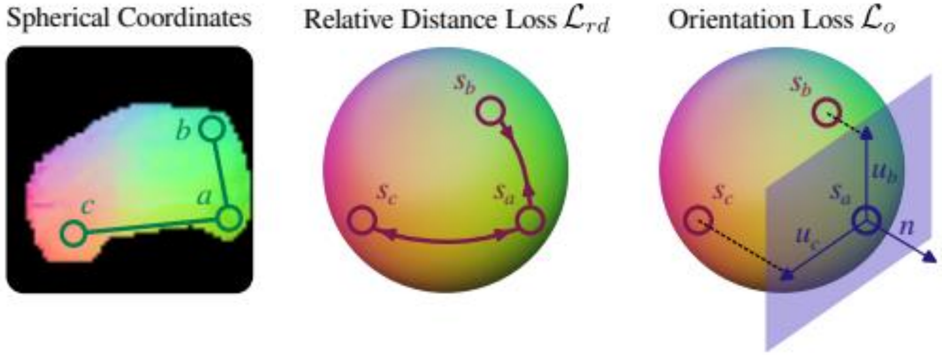

Spherical Coordinates
Relative Distance Loss $\mathcal{L}_{rd}$
Orientation Loss $\mathcal{L}_o$
b
a
c
s_b
s_c
s_a
s_b
s_c
s_a
u_b
u_c
n

Figure 3. Illustration of our geometry losses $\mathcal{L}_{rd}$ and $\mathcal{L}_o$. The left image shows a spherical map from which a triplet of points is sampled. $\mathcal{L}_{rd}$: as the anchor patch a is closer to the positive b on the image compared with the negative c, its corresponding position s_a on the sphere must also be closer to s_b than s_c. $\mathcal{L}_o$: after projecting s_b and s_c to the plane tangent to the sphere at s_a, we ensure orientation is preserved by enforcing positive colinearity between $u_b \times u_c$ and the normal vector n.

Orientation loss

Assumption image triplets of large determinants should also have large determinant on the sphere

$$\mathcal{L}_o = \begin{cases} 0 & \text{if } d_I < d_\tau \\ max(d_\tau - d_S, 0) & \text{if } d_I \geq d_\tau. \end{cases}$$

a threshold dτ = 0.7

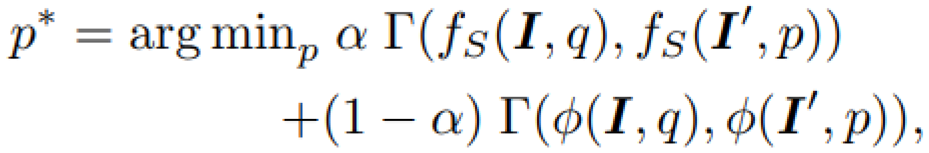
Swapping b and c if d_I is negative

$$d_I = \det(b-a, c-a)$$

$$d_S = \det(P_a(s_b) - s_a), P_a(s_c) - s_a)$$

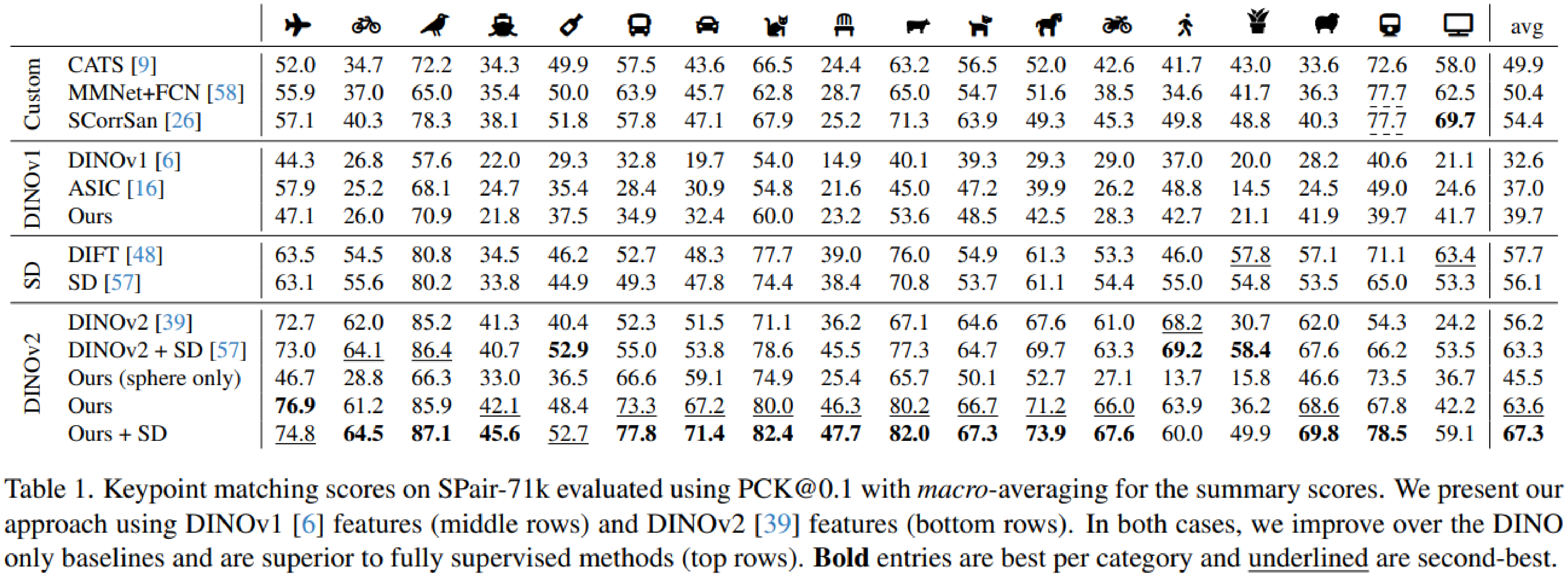
Linear projection to the plane tangent

Correspondence via combined representations



$$p^* = \arg\min_p \alpha\, \Gamma(f_S(\boldsymbol{I}, q), f_S(\boldsymbol{I}', p)) + (1-\alpha)\, \Gamma(\phi(\boldsymbol{I}, q), \phi(\boldsymbol{I}', p)),$$

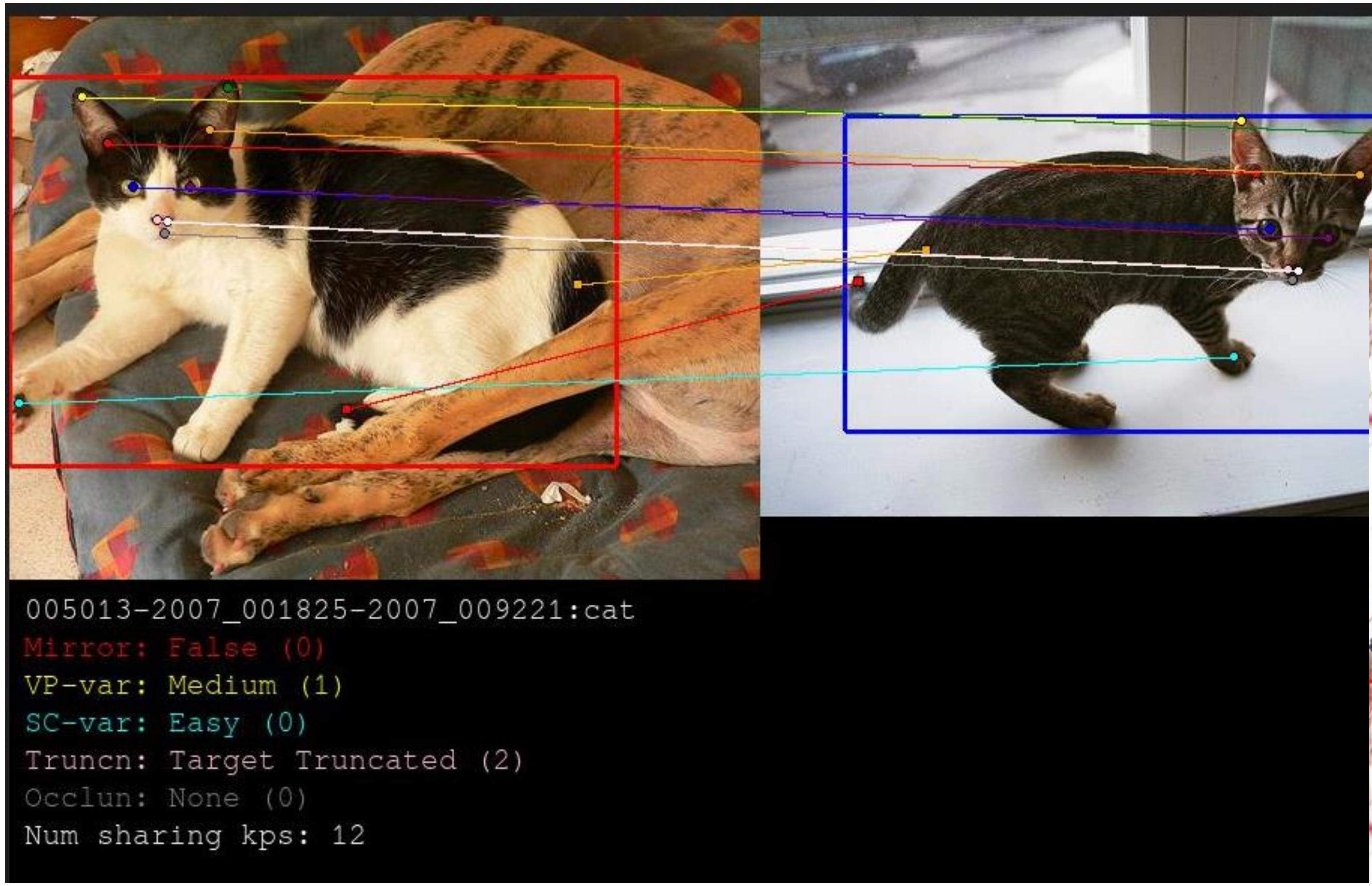
$\alpha = 0.2$



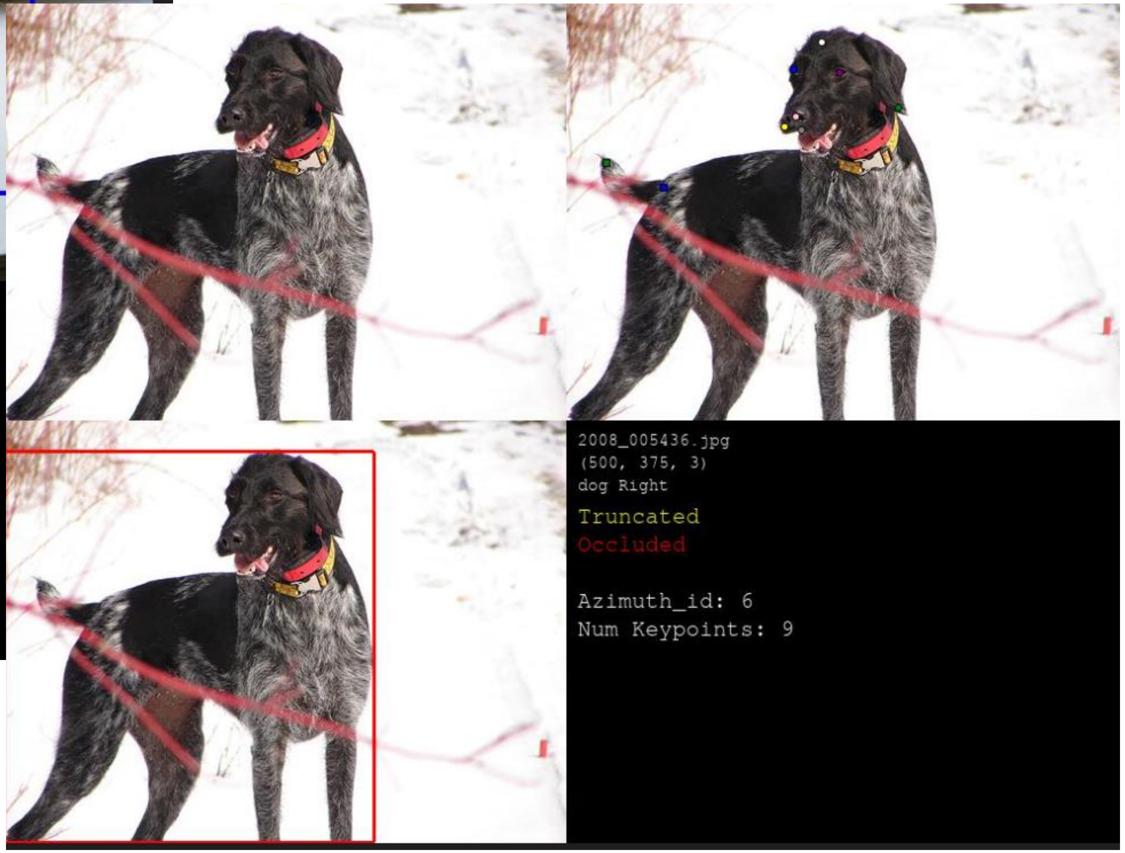
		✈	🚲	🐦	🚢	🍾	🚌	🚗	🐈	🪑	🐄	🐕	🐎	🏍	🚶	🪴	🐑	🚆	🖥	avg
Custom	CATS [9]	52.0	34.7	72.2	34.3	49.9	57.5	43.6	66.5	24.4	63.2	56.5	52.0	42.6	41.7	43.0	33.6	72.6	58.0	49.9
	MMNet+FCN [58]	55.9	37.0	65.0	35.4	50.0	63.9	45.7	62.8	28.7	65.0	54.7	51.6	38.5	34.6	41.7	36.3	77.7	62.5	50.4
	SCorrSan [26]	57.1	40.3	78.3	38.1	51.8	57.8	47.1	67.9	25.2	71.3	63.9	49.3	45.3	49.8	48.8	40.3	77.7	**69.7**	54.4
DINOv1	DINOv1 [6]	44.3	26.8	57.6	22.0	29.3	32.8	19.7	54.0	14.9	40.1	39.3	29.3	29.0	37.0	20.0	28.2	40.6	21.1	32.6
	ASIC [16]	57.9	25.2	68.1	24.7	35.4	28.4	30.9	54.8	21.6	45.0	47.2	39.9	26.2	48.8	14.5	24.5	49.0	24.6	37.0
	Ours	47.1	26.0	70.9	21.8	37.5	34.9	32.4	60.0	23.2	53.6	48.5	42.5	28.3	42.7	21.1	41.9	39.7	41.7	39.7
SD	DIFT [48]	63.5	54.5	80.8	34.5	46.2	52.7	48.3	77.7	39.0	76.0	54.9	61.3	53.3	46.0	57.8	57.1	71.1	63.4	57.7
	SD [57]	63.1	55.6	80.2	33.8	44.9	49.3	47.8	74.4	38.4	70.8	53.7	61.1	54.4	55.0	54.8	53.5	65.0	53.3	56.1
DINOv2	DINOv2 [39]	72.7	62.0	85.2	41.3	40.4	52.3	51.5	71.1	36.2	67.1	64.6	67.6	61.0	68.2	30.7	62.0	54.3	24.2	56.2
	DINOv2 + SD [57]	73.0	64.1	86.4	40.7	**52.9**	55.0	53.8	78.6	45.5	77.3	64.7	69.7	63.3	**69.2**	**58.4**	67.6	66.2	53.5	63.3
	Ours (sphere only)	46.7	28.8	66.3	33.0	36.5	66.6	59.1	74.9	25.4	65.7	50.1	52.7	27.1	13.7	15.8	46.6	73.5	36.7	45.5
	Ours	**76.9**	61.2	85.9	42.1	48.4	73.3	67.2	80.0	46.3	80.2	66.7	71.2	66.0	63.9	36.2	68.6	67.8	42.2	63.6
	Ours + SD	74.8	**64.5**	**87.1**	**45.6**	52.7	**77.8**	**71.4**	**82.4**	**47.7**	**82.0**	**67.3**	**73.9**	**67.6**	60.0	49.9	**69.8**	**78.5**	59.1	**67.3**

Table 1. Keypoint matching scores on SPair-71k evaluated using PCK@0.1 with *macro*-averaging for the summary scores. We present our approach using DINOv1 [6] features (middle rows) and DINOv2 [39] features (bottom rows). In both cases, we improve over the DINO only baselines and are superior to fully supervised methods (top rows). **Bold** entries are best per category and underlined are second-best.

Spair71k data example



```
005013-2007_001825-2007_009221:cat
Mirror: False (0)
VP-var: Medium (1)
SC-var: Easy (0)
Truncn: Target Truncated (2)
Occlun: None (0)
Num sharing kps: 12
```



```
2008_005436.jpg
(500, 375, 3)
dog Right
Truncated
Occluded

Azimuth_id: 6
Num Keypoints: 9
```

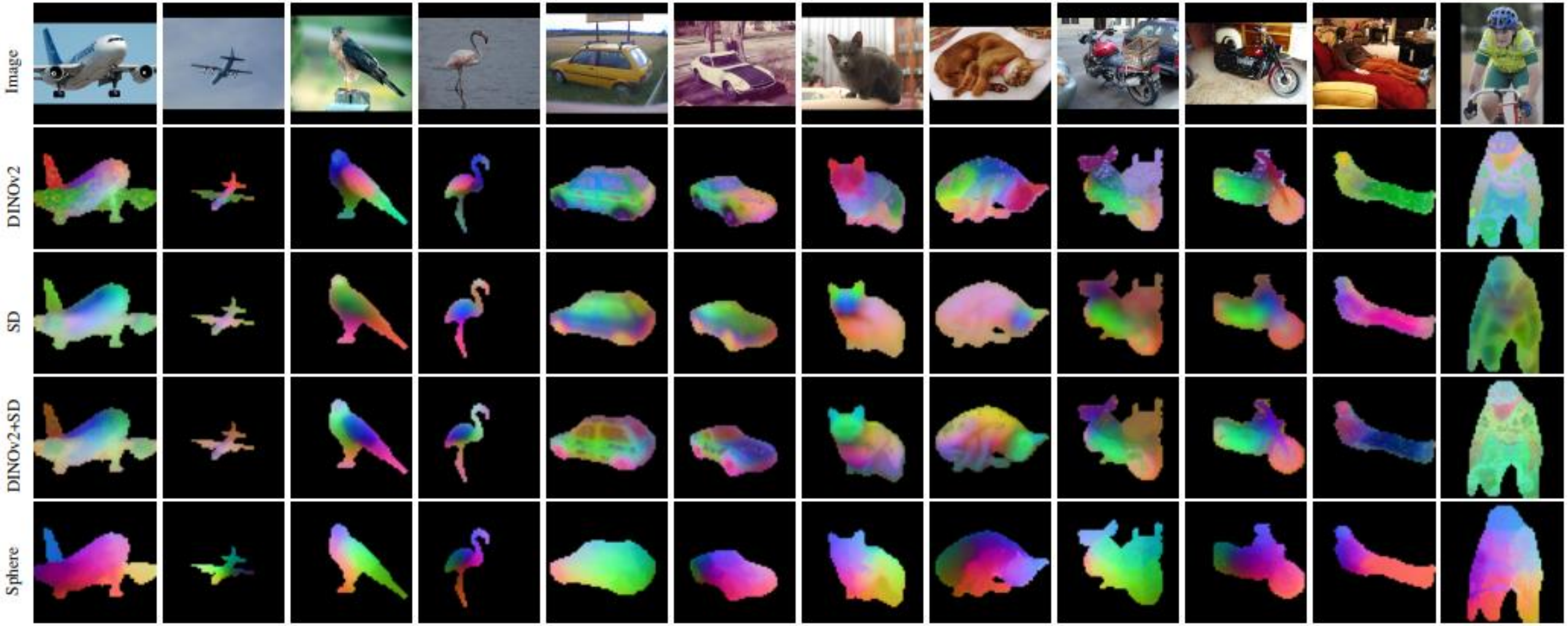


Figure 4. Qualitative comparison of dense correspondence maps. For DINOv2, SD, and DINO+SD features we perform PCA on the segmented object features independently for each category, then visualize the three main components. Note that the SD and DINO+SD features are not completely equivalent to the ones used to compute matches, but are provided here for illustration. Spherical maps from f_S (Sphere) for our approach are visualized directly. Our spherical maps correctly identify the different sides of objects, whereas other features fail to capture these differences.

	✈	🚲	🐦	🚢	🍾	🚌	🚗	🐈	🪑	🐄	🐕	🐎	🏍	🚶	🪴	🐑	🚆	🖥	avg
DINOv2 [40]	53.5	54.0	60.2	35.5	44.4	36.3	31.7	61.3	37.4	54.7	52.5	51.5	48.8	48.2	37.8	44.1	47.4	38.2	46.5
SD [58]	44.4	48.5	54.5	31.5	45.2	32.7	30.0	68.4	35.8	55.2	47.9	48.1	44.8	42.3	44.5	39.2	52.7	51.2	45.4
DINOv2 + SD [58]	52.0	**55.9**	59.2	34.7	**49.0**	36.0	32.5	70.3	39.8	59.8	53.1	52.4	50.6	**50.4**	**47.8**	46.2	53.3	49.8	49.6
Ours (sphere only)	38.4	34.2	53.9	33.0	37.9	49.7	43.4	71.7	29.8	57.1	45.8	42.5	32.4	27.0	29.5	37.1	57.4	36.0	42.1
Ours	**60.7**	51.2	**63.1**	38.4	45.0	**55.9**	45.7	69.7	40.4	63.2	54.8	54.3	51.2	48.7	38.8	47.9	55.5	42.2	51.5
Ours + SD	58.9	54.2	62.2	**39.6**	46.6	54.5	**47.1**	**76.2**	**40.9**	**65.3**	**57.3**	**56.1**	**54.2**	47.4	43.7	**49.4**	**62.4**	**52.0**	**53.8**

Table 2. Keypoint matching scores on SPair-71k evaluated using KAP@0.1 with macro-averaging for the summary scores.

	Dv2	SD	Dv2+SD	Ours	Ours+SD
PCK@0.1	65.9	56.0	68.1	68.7	69.8
KAP@0.1	55.0	50.7	56.8	58.9	60.6

Table 3. Average scores when evaluating on AwA-pose using a random subset of 200 pairs per category. 'Ours' denotes our model trained on SPair-71k with a DINOv2 backbone.

DINOv2	SD	DINOv2+SD	Ours	Ours+SD
3.4	0.38	0.35	3.3	0.34

Table 5. Descriptor computation throughput in pairs/second at inference time on a single A5000 GPU, where higher is better.