Discovering Relationships between Object Categories via Universal Canonical Maps

Natalia Neverova*, Artsiom Sanakoyeu*, Patrick Labatut, David Novotny, Andrea Vedaldi
Facebook AI Research

A. Experiments

A.1. DensePose-LVIS v1.0 dataset details

We introduce DensePose-LVIS v1.0 dataset, an extended version of the DensePose-LVIS data of [7]. We improve the quality of the existing labels and expand the DensePose annotation pool for the same animal classes as in the previous version of this dataset [7]. In Tab. S1 we report the number of train and test instances annotated with DensePose in every category.

category	DensePose-LVIS		DensePose-LVIS v1.0	
	train, inst.	test, inst.	train, inst.	test, inst.
dog	483	200	1607	316
cat	586	200	1912	379
bear	98	200	735	132
sheep	257	200	1655	350
cow	426	200	2105	340
horse	605	200	2292	458
zebra	665	200	2864	556
giraffe	651	200	2709	534
elephant	670	200	2839	539
all	4441	1800	18718	3604

Table S1: **DensePose-LVIS v1.0 dataset:** 3.6x increase in a number of annotated instances, better quality of labels.

A.2. 3D mesh alignment

If not stated otherwise, we used cross-validation to find the **m2m** loss weight that maximizes **AP** metric after training. Additionally we also cross-validated the **m2m** loss weight to minimize the **GErr**, we denote this experiment as **m2m***. As mentioned in the caption of Fig. 4, the optimal weight of **m2m** term is tenfold larger for **GErr** than for **AP**. Visual mappings in Fig. 4 correspond to the **m2m** model.

A.3. Keypoint transfer

For keypoint transfer experiments we train our models on DensePose-LVIS v1.0 dataset and do not use any PASCAL VOC [1] images during training. We select animal categories from PASCAL VOC [1], overlapping with animal categories in DensePose-LVIS v1.0: horse, cow, sheep, cat, dog. Following Kulkarni et al. [3], we randomly sample 100 images for each category from PASCAL VOC mentioned above and use them for evaluation. We average PCK-Transfer score across all possible (source, target) image pairs.

method	animal category					mean
	HORSE	COW	SHEEP	CAT	DOG	
our baseline	58.1	49.9	43.9	41.6	41.9	47.1
w/ **m2m**	57.1	49.5	45.1	40.0	42.5	46.8
w/ **i2m**	59.0	51.1	46.2	45.9	45.7	49.7
w/ **i2m**-all	**59.2**	**51.5**	**46.3**	**46.5**	**45.9**	**49.9**
w/ **m2m+i2m**	57.7	49.9	44.8	40.6	42.4	47.1
w/ **m2m+i2m**-all	57.8	50.2	44.9	40.6	42.6	47.2

Table S2: **Keypoint transfer on PASCAL VOC, within each of training animal categories**. PCK-Transfer metric, higher is better. **m2m** term is not helpful for this task.

We conducted keypoint transfer experiments using three distinct settings:

(a) Within each category observed at training time, when source and target images are from the same category (Tab. 3 in the main paper);

(a) Across training categories (Tab. 4, I in the main paper). In this case source and target images are from different training categories. For example, keypoints from dog images are transferred to images from horse, cow, sheep, and cat categories;

(a) Zero shot scenario: Within new animal categories not observed at training time (Tab. 4 II in the main paper). In this case, we remove ground truth dense correspondences for one class from the training set and evaluate keypoint transfer on the images within the removed class. Note that we do not remove bounding boxes and object instance masks from training set and still use them to train our detection and segmentation heads.

*Both authors contributed equally to this work.

method	target class	HORSE	COW	SHEEP	CAT	DOG	mean
our baseline	single	28.8	27.3	33.7	31.1	29.4	30.0
w/ **i2m**	single	30.1	28.2	34.1	31.9	29.7	30.8

Table S3: **Effect of i2m loss term when trained and evaluated on individual categories of DensePose-LVIS v1.0.** We report DensePose **AP** score.

method	target class	HORSE	COW	SHEEP	CAT	DOG	mean
Rigid-CSM + keyp. [4]	single	42.1	28.5	31.5	–	–	–
A-CSM + keyp. [3]	single	44.6	29.2	39.0	–	–	–
our baseline	single	53.4	48.0	38.8	40.9	34.0	43.0
w/ **i2m**	single	54.0	49.1	39.2	34.7	44.2	44.2

Table S4: **Keypoint transfer on PASCAL VOC, when trained and evaluated on individual categories.** PCK-Transfer metric, higher is better.

In Tab. S2 we show results for all combinations of the loss terms on keypoint transfer within each category. The **i2m** term enforces alignment on image level which is more important for the task of within-category keypoint transfer, while the **m2m** loss term is not helpful in this case as it enforces the alignment between 3D templates.

A.4. Effect of i2m loss in a single-class training scenario

In contrast to our approach, Rigid-CSM [4] and A-CSM [3] cannot learn multiple animal categories in a single model and have to train separate models for every animal class. To make our setup closer to those in [4, 3], we trained our models on individual categories from DensePose-LVIS v1.0 as well (i.e., trained a new model for each class). Then we evaluated each model on the corresponding individual categories: (a) on the test set of DensePose-LVIS v1.0 by computing DensePose AP score (see Tab. S3); and (b) on PASCAL VOC by computing PCK-Transfer metric for Keypoint Transfer task (see Tab. S4). From the tables S3, S4 we can see that **i2m** improves the performance even when trained and evaluated on individual categories and outperforms methods [4, 3] (we used the results reported in the corresponding papers). However, a part of the strength of our method comes from training on several classes jointly, which results in even stronger performance of our models (see Tab. 3 in the main paper).

A.4.1 Can the i2m loss be used in combination with Rigid-CSM [4] or A-CSM [3] models?

Our **i2m** loss requires every pixel and every vertex of the mesh to be embedded in a common embedding space. However, models [4, 3] directly predict (u, v) coordinates for every pixel. Therefore our loss cannot be applied during training of Rigid-CSM [4] and A-CSM [3].

A.5. Evaluation metrics

For completeness, we provide brief descriptions of **AP/AR** metrics used for evaluation of learned dense pose predictions on DensePose-LVIS v1.0 dataset and cross-category mesh alignment metrics **GErr**, **GPS**.

- **GPS** (Geodesic Point Similarity) [2] is a correspondence matching score indicator of the quality of aligning of two sets of vertices $A = (a_1, \ldots, a_N)$ and $B = (b_1, \ldots, a_N)$ on a mesh.

$$\mathbf{GPS}(A, B) = \frac{1}{N} \sum_{i=1}^{N} \exp\left(\frac{-g(a_i, b_i)^2}{2\kappa^2}\right),$$

  where N is the number of vertices in each set, $g(\cdot, \cdot)$ is the geodesic distance between two surface points, and κ is a normalization constant. To make our **GPS** score comparable to the **GPS** score used for Human DensePose evaluation in Guler et al. [2], we normalize all vertex coordinates in every animal mesh to have the maximum geodesic distance $d_{max} = 2.27$, which is equal to the maximal geodesic distance in the SMPL [6] mesh of a human utilized in [2]. We set $\kappa = 0.255$ so that a single point has a **GPS** value of 0.5 if its geodesic distance from the ground truth equals the average half-size of a body segment. When we evaluate cross-category mesh alignment quality, we compute **GPS** between a set of the ground truth semantic keypoints on a target mesh and the estimated locations of these keypoints obtained by transferring keypoints from a source mesh of other category. The mean **GPS** score is then computed as an average across all possible (source, target) pairs of categories.

- Similarly to Geodesic Point Similarity, we define **GErr** (Geodesic Error), the error between two sets of

vertices along the surface of a mesh. Before computing this error, all vertex coordinates are normalised to have the maximum of geodesic distance $d_{max} = 2.27$ (similar to [2, 7]).

$$\mathbf{GErr}(A, B) = \frac{1}{|N|} \sum_{i=1}^{N} g(a_i, b_i).$$

Analogously to **GPS**, we use **GErr** to estimate the quality of inter-category mesh alignment by comparing the ground truth semantic keypoints on a target mesh and the estimated locations of these keypoints obtained by transferring keypoints from a source mesh of other category.

- To evaluate the quality of mapping from image pixels to 3D vertices on the category-specific mesh, we use **AP** (Average Precision) and **AR** (Average Recall) [2]. The location of the vertices on the mesh corresponding to image pixels are estimated by finding for every pixel the most similar mesh vertex in the learned embedding space. After that we compare estimated vertex locations with the ground truth using **GPS** metric. Then we calculate **AP** and **AR** at different **GPS** thresholds ranging from 0.5 to 0.95, following the COCO challenge protocol [5]. We separately report Average Precision and Average Recall at **GPS** thresholds equal to 0.5 and 0.75, denoted as $\mathbf{AP}_{50}, \mathbf{AP}_{75}, \mathbf{AR}_{50}, \mathbf{AR}_{75}$. In addition to this we separately compute Average Precision and Average Recall for instances with *medium* and *large* sizes ($\mathbf{AP}_M, \mathbf{AR}_M$ for medium size and $\mathbf{AP}_L, \mathbf{AR}_L$ for large).

Note, that we report **GPS** $\times 100$ and **GErr** $\times 100$ in all tables in the main paper.

References

[1] Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 111(1), 2015. 1

[2] Riza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. DensePose: Dense human pose estimation in the wild. In *Proc. CVPR*, 2018. 2, 3

[3] Nilesh Kulkarni, Abhinav Gupta, David F Fouhey, and Shubham Tulsiani. Articulation-aware canonical surface mapping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 452–461, 2020. 1, 2

[4] Nilesh Kulkarni, Abhinav Gupta, and Shubham Tulsiani. Canonical surface mapping via geometric cycle consistency. In *Proc. ICCV*, 2019. 2

[5] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. In *Proc. ECCV*, 2014. 3

[6] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: a skinned multi-person linear model. *ACM Trans. on Graphics (TOG)*, 2015. 2

[7] Natalia Neverova, David Novotný, and Andrea Vedaldi. Continuous surface embeddings. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 1, 3