Neural View Synthesis and Matching for Semi-Supervised Few-Shot Learning of 3D Pose

Angtian Wang, Shenxiao Mei, Alan Yuille, Adam Kortylewski

Contributions

In this work, we focus on learning to estimate the 3D object pose from a few labeled examples (minimum 7) and a collection of unlabeled data.

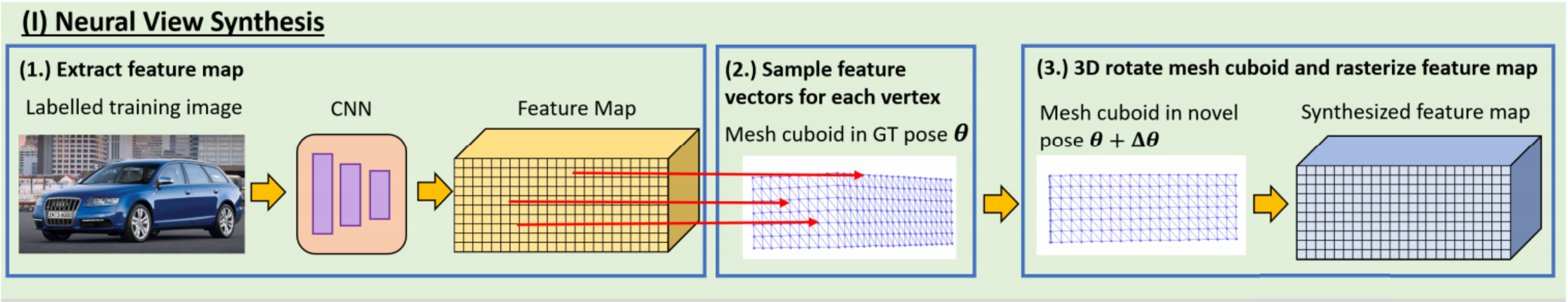
- We propose a pose matching method that can leverage pose information to unlabeled data given a 3D object shape Γ:
  - Given a pose annotated image A, and an unposed image B as input. It calculates the similarity score under a certain pose shift $\Delta\theta$:

$$S(A, B|\Delta\theta, \Gamma)$$

- We propose a semi-supervised few-shot learning method, which allows us to train a pose estimation model with very few labeled examples.

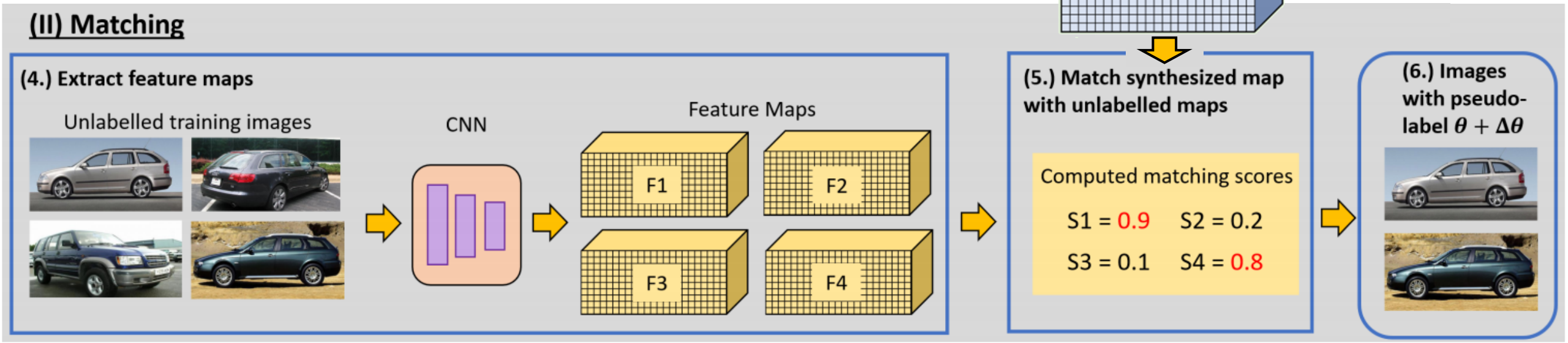
Neural View Synthesis

- To synthesis a feature under given camera pose:
  - Extract a feature map with a CNN backbone (can be ImageNet pretrained or contrastive trained).
  - Sample features using the location of the projected vertices given pose θ.
  - Computer vertices locations under pose θ+Δθ.
  - Rasterize and interpolate.



Matching For Object Pose

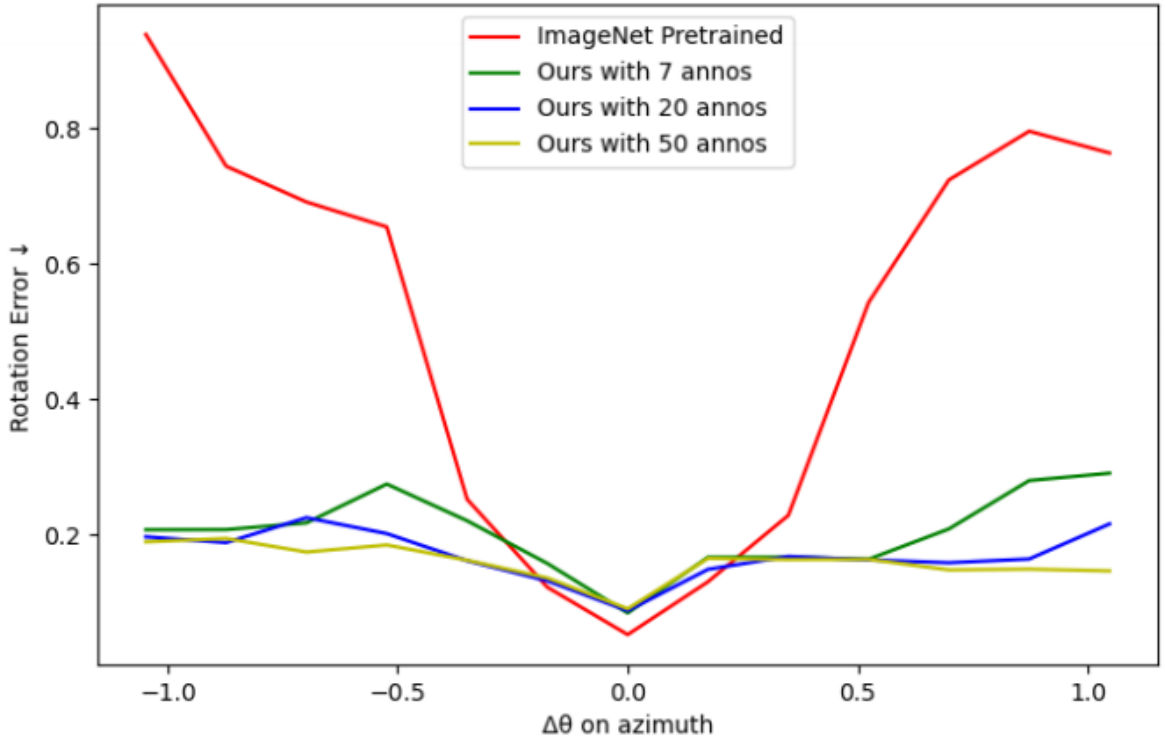
- To conduct match on unposed images using the synthesized feature map:
  - Extract feature using the same feature extract.
  - Compute the similarity between the synthesized feature and extracted feature.
  - Thresholding the similar to get matched images.

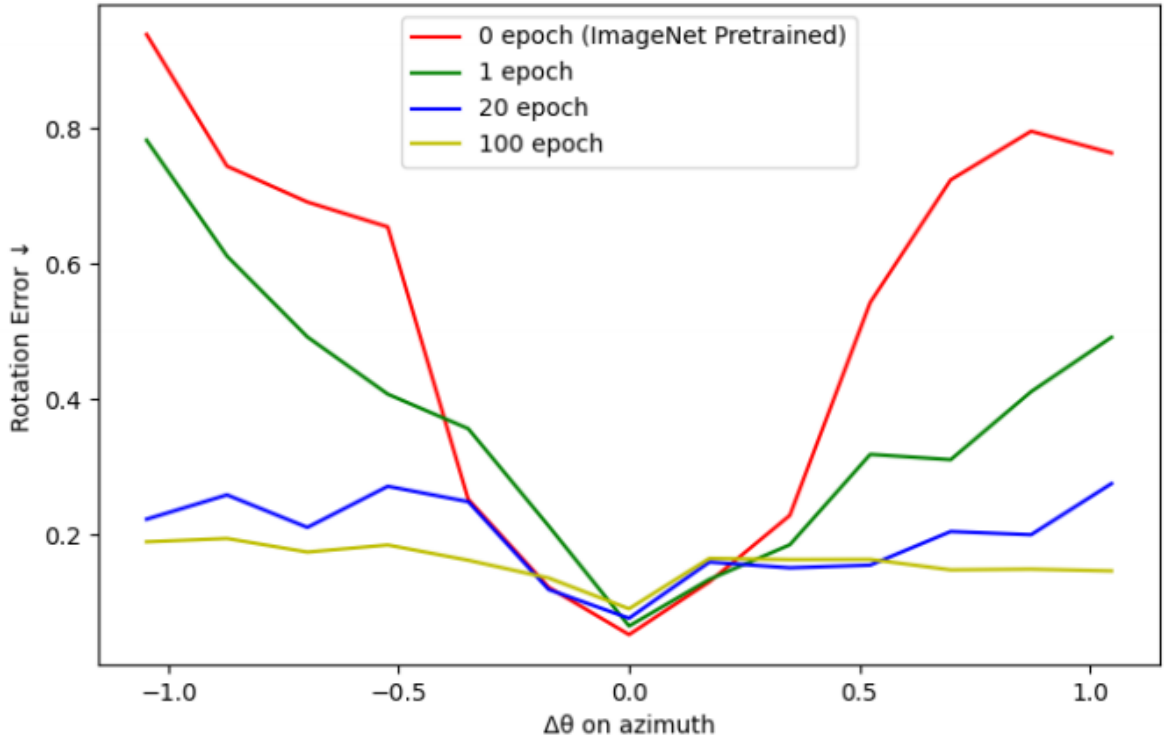


Qualitative Matching Result

Δθ	Synthesis Pose θ+Δθ	With ImageNet Pretrained (backbone) Mean(|Δθ error|): 41.5		After Semi-Supervised Training Mean(|Δθ error|): 8.7	
-40		 |Δθ error|: 140.8	 142.9	 |Δθ error|: 6.8	 18.3
-20		 |Δθ error|: 5.5	 11.1	 |Δθ error|: 2.2	 10.3
0		 |Δθ error|: 0.1	 1.4	 |Δθ error|: 4.3	 8.4
+20		 |Δθ error|: 29.7	 51.1	 |Δθ error|: 9.4	 6.9
+40		 |Δθ error|: 14.7	 14.2	 |Δθ error|: 4.6	 11.5

Quantitative Matching Result





Pose matching quality under different backbones (left) and different epochs (right). Evaluated with pose rotation error (lower is better).

Semi-Supervised Learning

- A set of anchor images (7 in experiment), manually picked.
- A set of annotated images includes anchor images (7, 20, 50), randomly selected.

Anchor Images for Car Category

Semi-Supervised Learning

Goal: Eventually leveraging the pose information from the annotated images to the whole pose space.

- Pseudo labeling unlabeled training images. ⟶ E step
- Training the feature extractor and a mean feature representation (to make the computational cost reasonable, we only maintain a mean representation instead of match a new image with all annotated images). ⟶ M step

We conduct the semi-supervised learning process in an EM manner that we iteratively do the two steps.

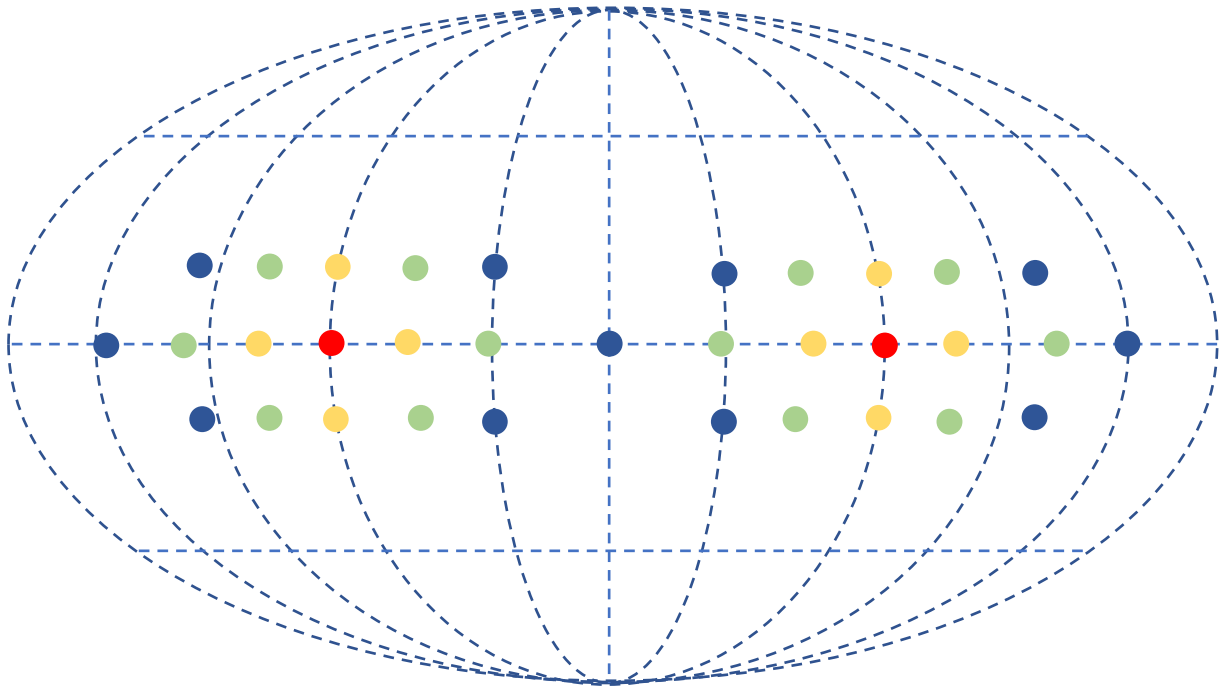
Semi-Supervised Learning – E step

- Starting from an anchor pose, we sample some poses in a small distance.
- Use the sampled pose as Δθ to synthesis a feature map.

$$F_{\theta'} = \Re(\Gamma, \Sigma, \theta') \in \mathbb{R}^{H \times W \times C}$$

- Match the synthesized feature with all unlabeled images -> images with the top similarities:

$$S(F_{\theta'}, F_m) = \frac{1}{HW} \sum_h \sum_w [1 - d(F_{\theta'}(h,w), F_m(h,w))].$$



Anchor Pose

Pose Samples in Step 1

Pose Samples in Step 2

Pose Samples in Step 2

Semi-Supervised Learning – M step

- We use the contrastive loss induced in CoKe[1] to train the backbone (F is the feature map, Γ is the mesh, P_θ is the projection matrix given pose θ):

$$L_+(F_i, F_j, \Gamma) = \sum_{r=1}^{R} [1 - d(F_i(P_{\theta_i} \cdot x_r), F_j(P_{\theta_j} \cdot x_r))].$$

$$L_-(F_i, F_j, \Gamma) = \sum_{r=1}^{R} \sum_{r' \neq r} d(F_i(P_{\theta_i} \cdot x_r), F_j(P_{\theta_j} \cdot x_{r'}))].$$

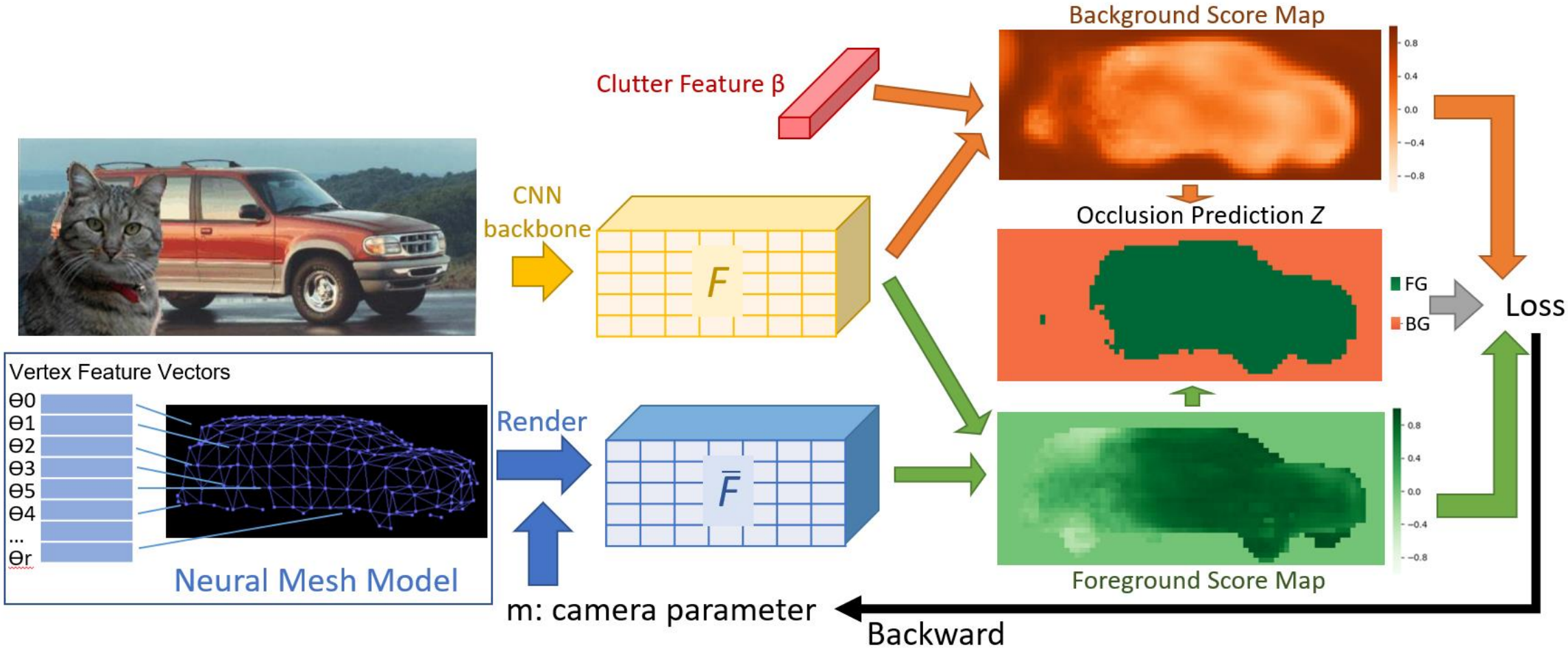
- We also use the moving average method introduced in CoKe to computer the approximated mean feature representation (σ is the average representation, α is the momentum):

$$\sigma_r^{t+1} = (1 - \alpha) * \phi_r^t + \alpha * \sum_n F_n(P_{\theta_n} \cdot x_r),$$

Bai Y, Wang A, Kortylewski A, et al. Coke: Localized contrastive learning for robust keypoint detection[J]. arXiv preprint arXiv:2009.14115, 2020.

Inference

We use the same inference pipeline from our previous work NeMo.



Wang A, Kortylewski A, Yuille A. NeMo: Neural Mesh Models of Contrastive Features for Robust 3D Pose Estimation, ICLR 2021.

Qualitative Pose Estimation Result

We illustrate the predicted 3D pose using a CAD model. Note that the CAD model is not used in our approach.

(a) PASCAL3D+, motorbike

(b) PASCAL3D+, bicycle

(c) PASCAL3D+, boat

(d) PASCAL3D+, bus

(e) KITTI, car

(f) KITTI, car

Quantitative Pose Estimation Result

Few-shot pose estimation results on PASCAL3D+. We indicate the number of annotations during training for each category.

Metric	$ACC_{\frac{\pi}{6}} \uparrow$				$ACC_{\frac{\pi}{18}} \uparrow$				*MedErr* $\downarrow$			
Num Annos	7	20	50	Mean	7	20	50	Mean	7	20	50	Mean
Res50-Gene	36.1	45.2	54.6	45.3	14.7	25.5	34.2	24.8	39.1	26.3	20.2	28.5
Res50-Spec	29.6	42.8	50.4	40.9	13.3	23.0	29.3	21.9	46.5	29.4	23.0	32.9
StarMap	30.7	35.6	53.8	40.0	4.3	7.2	19.0	10.1	49.6	46.4	27.9	41.3
NeMo	38.4	51.7	69.3	53.1	17.8	31.9	45.7	31.8	60.0	33.3	22.1	38.5
Ours	53.8	61.7	65.6	60.4	27.0	34.0	39.8	33.6	37.5	28.7	24.2	30.1

Few-shot pose estimation results on KITTI dataset at different levels of partial occlusion.

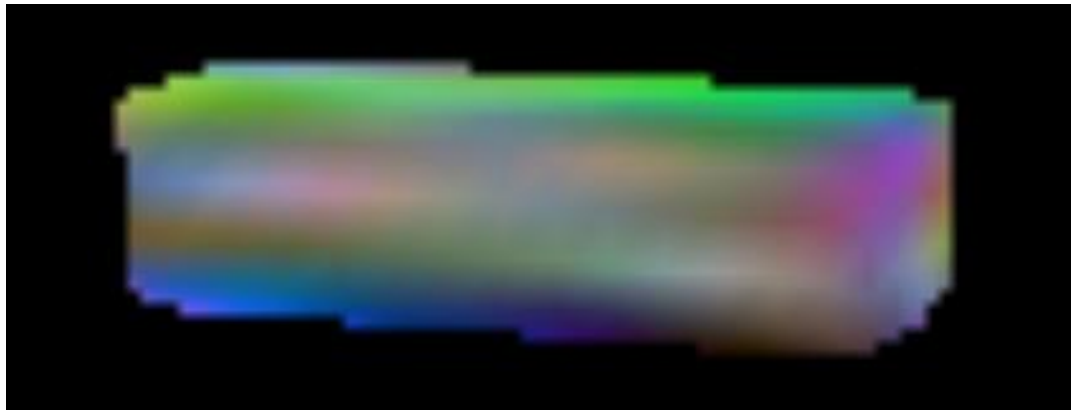
Eval Metric	Occ level	Fully visible			Partially occluded			Largely Occluded			Mean
	Num Annos	7	20	50	7	20	50	7	20	50	
$ACC_{\frac{\pi}{6}} \uparrow$	NeMo	34.3	83.9	89.8	14.9	58.2	74.6	4.3	27.5	30.4	58.5
	Ours	84.2	94.6	97.6	68.2	88.6	92.5	52.2	60.9	63.8	86.1
$ACC_{\frac{\pi}{18}} \uparrow$	NeMo	17.3	74.9	81.5	4.5	37.8	61.7	0.0	7.2	11.6	45.8
	Ours	23.2	81.1	88.3	18.7	81.6	82.1	16.4	39.1	42.0	60.0
MedErr $\downarrow$	NeMo	64.1	5.9	5.6	84.1	13.2	7.8	99.9	59.8	50.0	32.6
	Ours	20.3	8.1	4.1	24	12.2	5.3	27	13.3	12.2	12.4

Visualization of Feature

We train a PCA on extracted features on the whole dataset. Then we use the trained PCA model to reduces features into RGB space.

- Left: Trained Backbone -> Extracted feature -> PCA -> RGB
- Right: Averaged per verts feature -> PCA -> RGB

Extracted feature

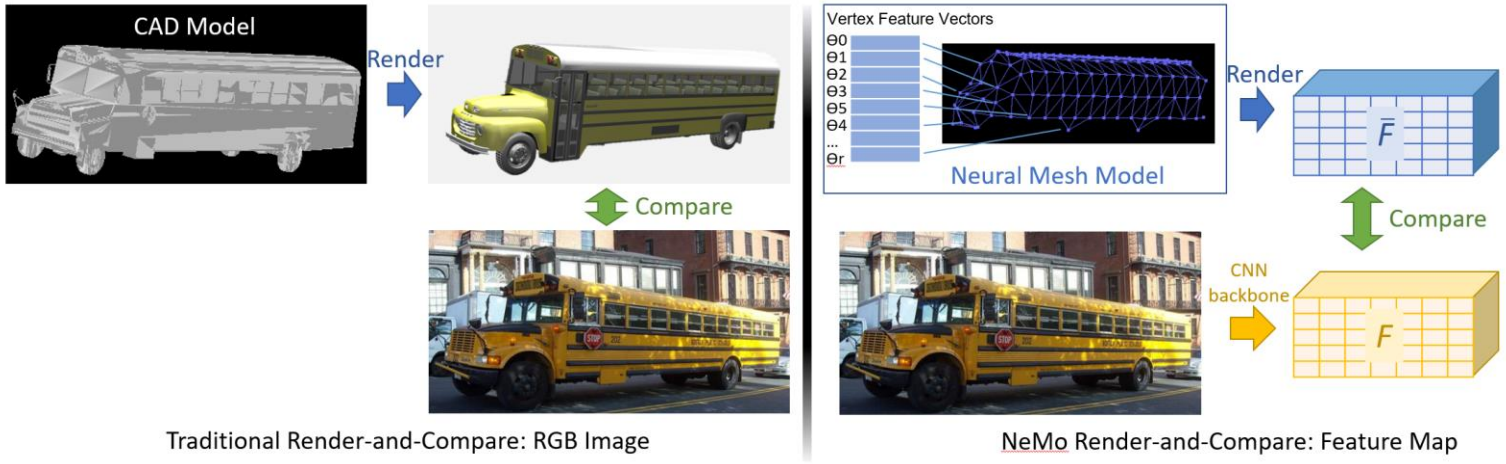


Average feature

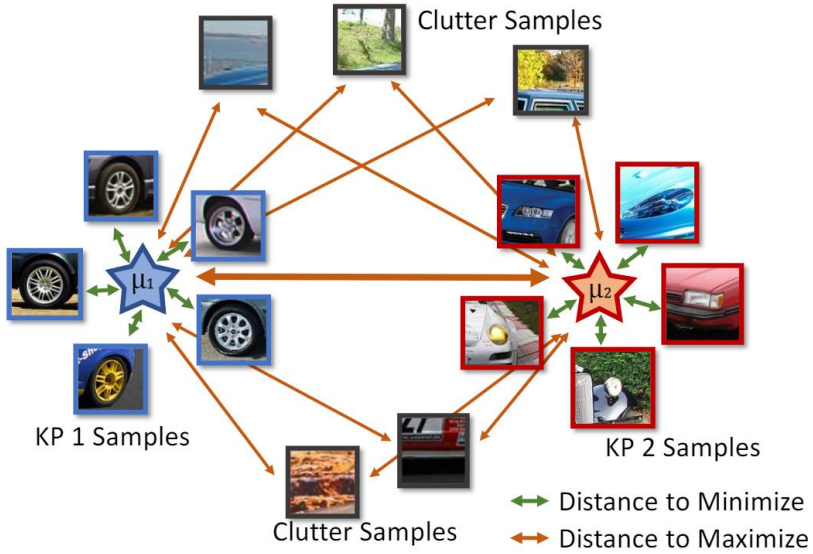
Other Works

If you are interested in our paper, please also have a look at our previous works:

- NeMo: Neural Mesh Models of Contrastive Features for Robust 3D Pose Estimation
- CoKe: Localized Contrastive Learning for Robust Keypoint Detection



NeMo: Render & Compare on Feature Level



CoKe: Contrastive Feature for Keypoints Detection

Thanks For Listen!