Supplementary Material: PIEs: Pose Invariant Embeddings

Chih-Hui Ho Pedro Morgado Amir Persekian Nuno Vasconcelos
University of California, San Diego
{chh279,pmaravil,aperseki,nvasconcelos}@ucsd.edu

A. Ablation study of triplet center based method

A.1. Effect of the α

To examine the effect of hyperparameter α in the pose invariant distance $d^{inv}(\mathbf{x}, \mathbf{X}_n, \mathbf{p}_y) = \alpha d(g(\mathbf{x}), g_m(\mathbf{X}_n)) + \beta d(g_m(\mathbf{X}_n), \mathbf{p}_y)$ (Eq.14), we alter the hyperparameter α with fixed β and show that larger α improves the average classification and retreival result in Table 1. This is highly correlated with the property of pose invariant distant discussion in section 3.4. With $\alpha = 0.1$, it is observed that it acts more like an MV-TC, while for $\alpha = 0.2$, the classifier has better performance on image classification, object retrieval and image retrieval, which are all related to single view level inference.

Task		MV-TC	PI-TC	
			$\alpha = 0.1$	$\alpha = 0.2$
Cls.	Image	77.3	81.2	**82.9**
	Multi	**88.9**	**88.9**	88.7
	Avg	83.1	85.1	**85.8**
Rtr.	Object	36.6	41.4	**46.8**
	Image	63.5	71.5	**75.8**
	Multi	84.0	**84.2**	84.1
	Avg	61.4	65.7	**68.9**

Table 1. Triplet center(TC) based methods on ModelNet40 with margin 1 and $\beta = 1$ is used for PI-TC. It can be observed that increasing α from 0.1 to 0.2 improves the average classification and retreival resuls.

A.2. Effect of the margin

To evaluate the effect of the margin m in the triplet center (TC) based architectures, we fix the hyperparameter $\alpha = 0.2$ and $\beta = 1$. We then compare MV-TC with PI-TC under different margin settings in Table 2. It is observed that PIE wins multiview based architecture on **18** tasks (out of 21). This shows PIE works over different margins and constantly outperforms multiview based architectures.

Margin m	Task		MV-TC	PI-TC
m=1	Cls.	Image	77.3	**82.9**
		Multi	**88.9**	88.7
		Avg	83.1	**85.8**
	Rtr.	Object	36.6	**46.8**
		Image	63.5	**75.8**
		Multi	84.0	**84.1**
		Avg	61.4	**68.9**
m=3	Cls.	Image	77.7	**84.2**
		Multi	**89.5**	88.8
		Avg	83.6	**86.5**
	Rtr.	Object	32.2	**42.1**
		Image	59.9	**77.4**
		Multi	84.1	**84.5**
		Avg	58.7	**68.0**
m=5	Cls.	Image	78.0	**84.6**
		Multi	**89.3**	88.7
		Avg	83.7	**86.7**
	Rtr.	Object	35.6	**40.9**
		Image	62.8	**78.2**
		Multi	84.4	**84,9**
		Avg	60.9	**68.0**

Table 2. Triplet center(TC) based methods on ModelNet40 with different margins. $\alpha = 0.2$ and $\beta = 1$ is used for all PI-TC. It can be observed that PIE wins the muliview based classifier on 18 tasks (out of 21).

B. Comparison to RotationNet using AlexNet

We refer to RotationNet[12] (RN) author's Caffe[1] pretrained model and use the feature before softmax extracted from the pretrained model for retreival task. As shown in Table 3, classification results are as stated but the retrieval results are worse then the proposed method trained with same backbone (AlexNet). Clearly, RN does not generalized well to retreival task.

[1] https://github.com/kanezaki/rotationnet

Method	Backbone	Cls. (Acc. %)		Rtr. (mAP %)		
		Img.	Shape	Obj.	Img.	Shape
RN[12]	AlexNet	84.0	90.6	21.4	21.9	66.9
PI-Proxy	AlexNet	82.6	87.9	34.8	75.2	83.0

Table 3. Comparison to RotationNet[12] using AlexNet

Dataset		ModelNet				ObjectPI			
Task		Single	Multiview	Multiview-Rand.	PIE	Single	Multiview	Multiview-Rand.	PIE
Cls.	Img.	**85.3**	79.7	80.5	85.1	68.5	63.2	64.7	**68.7**
	Multi	88.0	89.6	**90.1**	88.7	78.8	78.3	79.3	**80.0**
	Avg.	86.6	84.7	85.3	**86.9**	73.7	70.7	72.0	**74.4**
Rtr.	Obj.	**44.1**	35.0	36.0	40.6	47.7	49.3	45.7	**49.4**
	Img.	79.8	66.1	68.7	**79.9**	59.7	57.9	57.4	**62.6**
	Multi	83.9	85.1	**85.9**	85.1	76.8	74.7	74.0	**78.2**
	Avg.	**69.2**	62.1	63.6	68.6	61.4	60.6	59.1	**63.4**

Table 4. Proxy based methods on ModelNet40 and ObjectPI. Mutliview-Rand is trained with randomly selected views.

C. Comparison with random number of views

We provide results for multiview classifier trained with random number of view inputs as another baseline and denote it as multiview-rand. Table 4 shows the results for multiview-rand as well as single view for proxy based method. For ObjectPI, PIE has better performance over single view, multiview and multiview-random methods for all the tasks. For ModelNet, PIE has comparable results for the task that the classifier excels at. For example, PIE has approximate performance on single image classification task compared to that of single view classifier. This again demonstrates the robustness of the proposed approach.

D. Examples of ObjectPI

We have presented more examples in the ObjectPI dataset in Figure 1.

E. TSNE visualization

TSNE visualizations of view and multiview descriptors extracted from 3 different datasets, ModelNet, ObjectPI and MIRO, with different training architectures are provided. The visualization extracted from CNN, proxy and triplet center based architecture, are illustrated in Figure 2, 3 and 4 respectively.

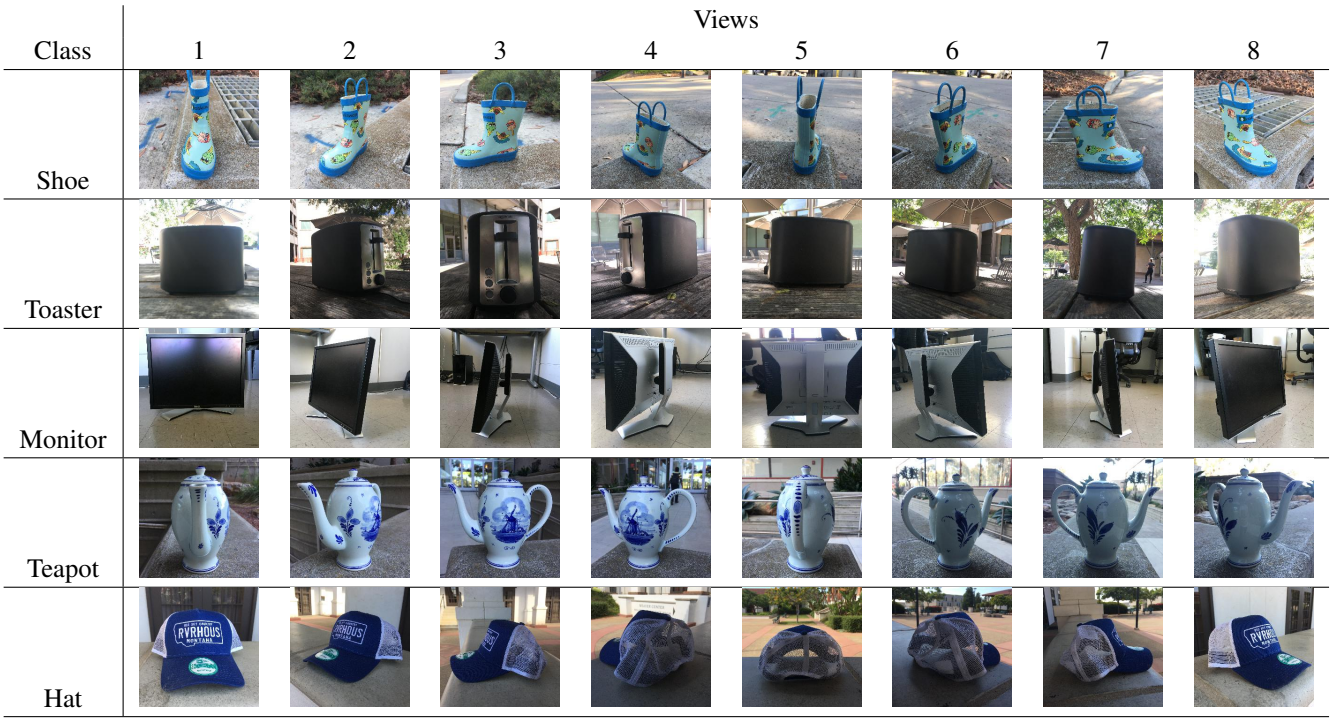

Figure 1. Examples of the 8 viewpoints of ObjectPI.

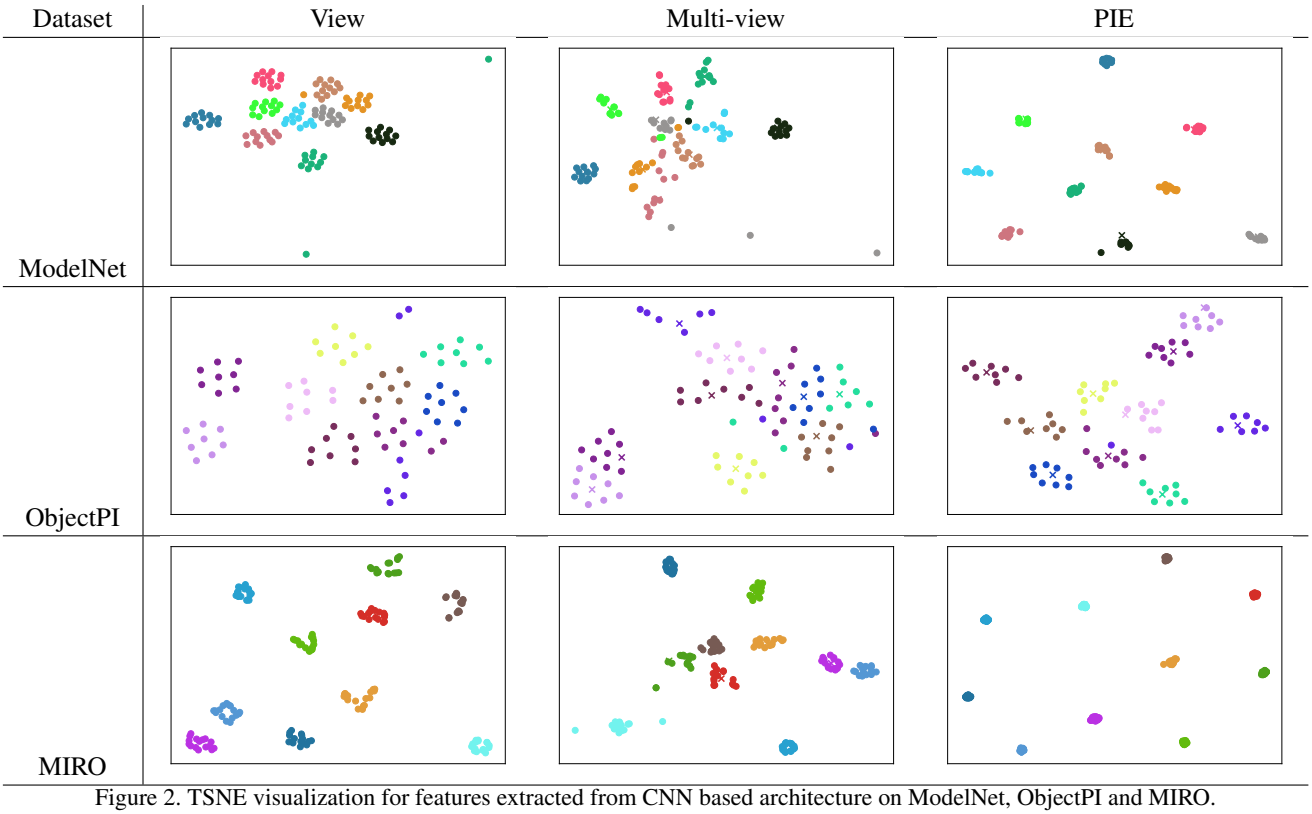

Figure 2. TSNE visualization for features extracted from CNN based architecture on ModelNet, ObjectPI and MIRO.

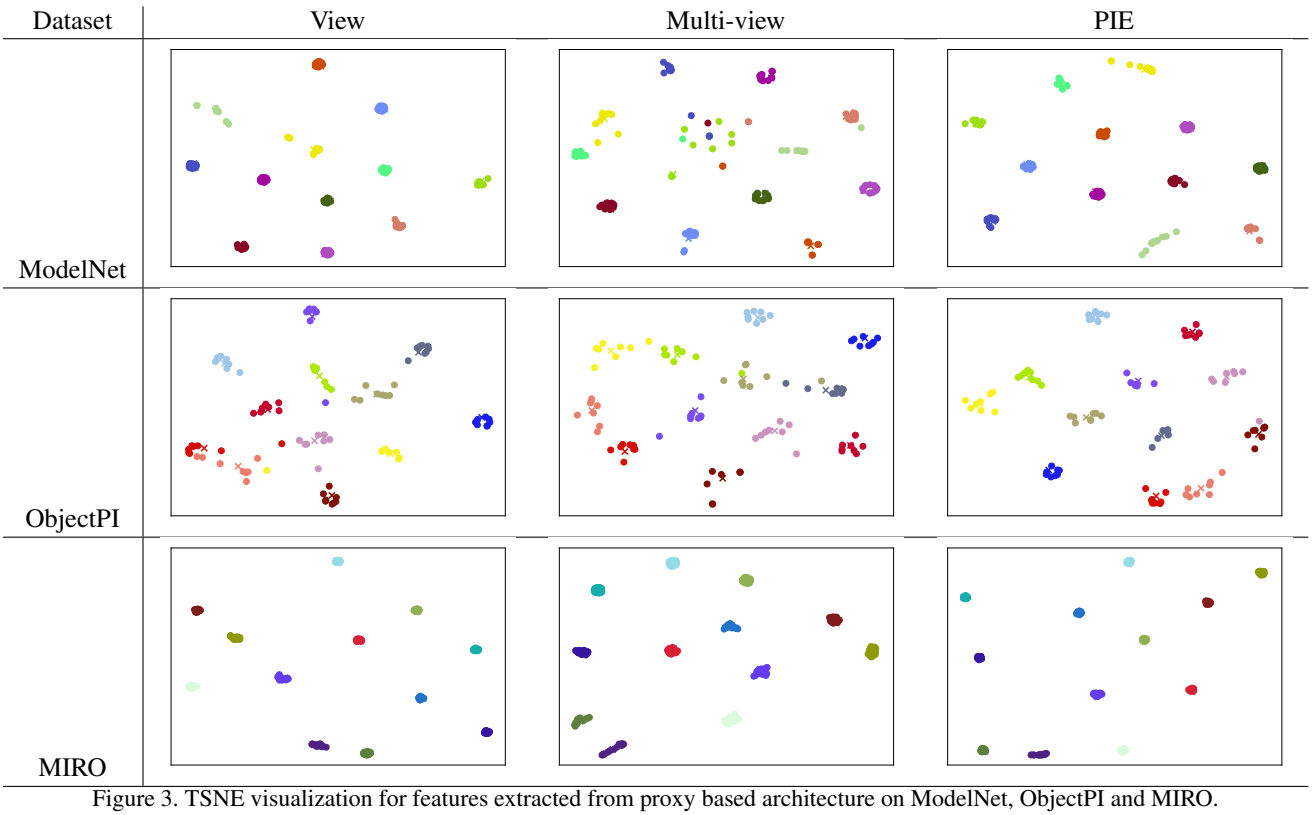

Figure 3. TSNE visualization for features extracted from proxy based architecture on ModelNet, ObjectPI and MIRO.

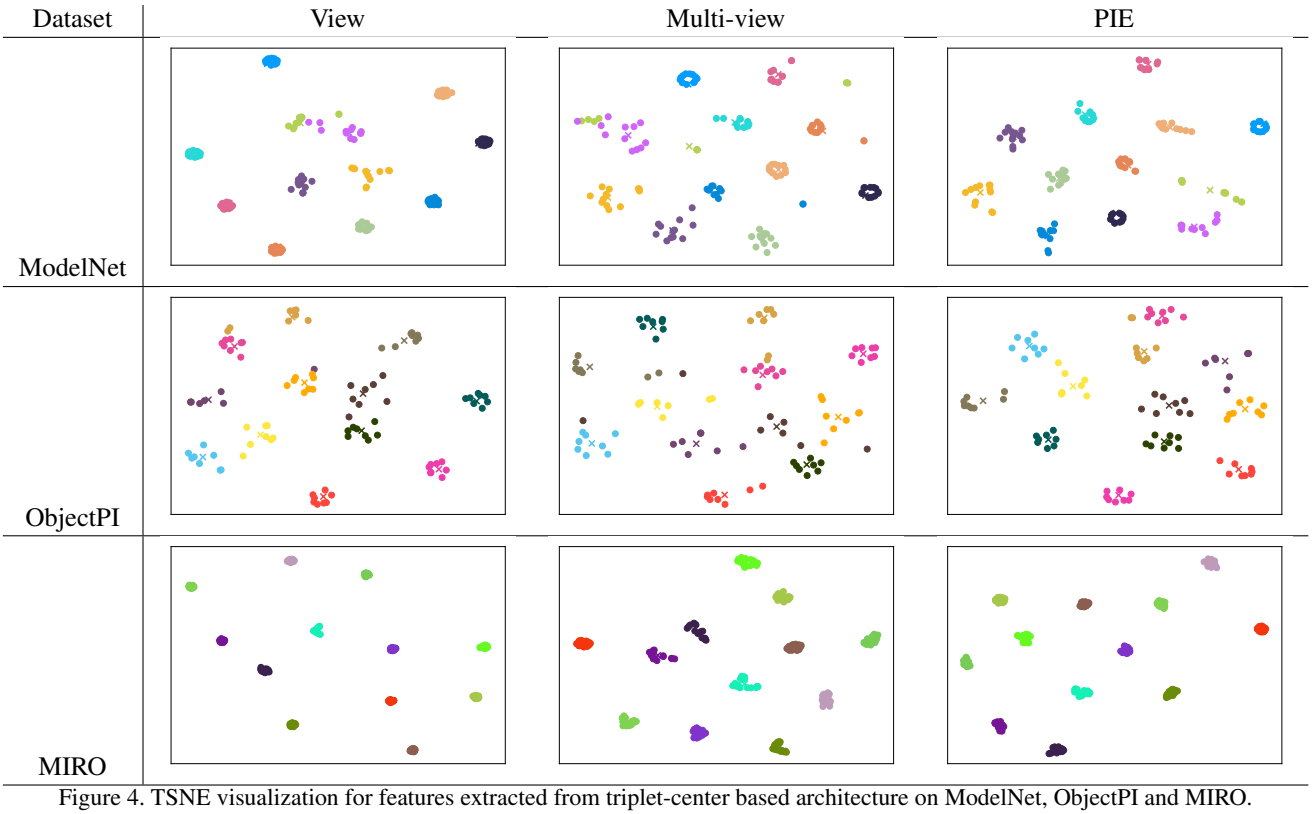

Figure 4. TSNE visualization for features extracted from triplet-center based architecture on ModelNet, ObjectPI and MIRO.