Supplementary Material: Universal Representation Learning from Multiple Domains for Few-shot Classification

Wei-Hong Li, Xialei Liu, and Hakan Bilen
VICO Group, University of Edinburgh, United Kingdom
groups.inf.ed.ac.uk/vico/research/URL

1. Implementation details

In all experiments we build our method on ResNet-18 [5] backbone for both single-domain and multi-domain networks.

1.1. Training details of single-domain models

We train one ResNet-18 model for each training dataset. For optimization, we follow the training protocol in [4]. Specifically, we use SGD optimizer and cosine annealing for all experiments with a momentum of 0.9 and a weight decay of 7×10^{-4}. The learning rate, batch size, annealing frequency, maximum number of iterations are shown in Table 1. To regularize training, we also use the exact same data augmentations as in [4], *e.g*. random crops and random color augmentations.

Dataset	learning rate	batch size	annealing freq.	max. iter.
ImageNet	3×10^{-2}	64	48,000	480,000
Omniglot	3×10^{-2}	16	3000	50,000
Aircraft	3×10^{-2}	8	3000	50,000
Birds	3×10^{-2}	16	3000	50,000
Textures	3×10^{-2}	32	1500	50,000
Quick Draw	1×10^{-2}	64	48,000	480,000
Fungi	3×10^{-2}	32	15,000	480,000
VGG Flower	3×10^{-2}	8	1500	50,000

Table 1. Training hyper-parameters of single domain learning.

1.2. Training details of our method

In the multi-domain network, we share all the layers but the last classifier across the domains. To train the multi-domain network, we use the same optimizer with a weight decay of 7×10^{-4} and a scheduler as single domain learning model for learning 240,000 iterations. The learning rate is 0.03 and the annealing frequency is 48,000. Similar to [13] that the training episodes have 50% probability coming from the ImageNet data source, each training batch for our multi-domain network consists of 50% data coming from ImageNet. In other words. The batch size for ImageNet is 64×7 and is 64 for the other 7 datasets.

We set λ^f and λ^p as 4 for ImageNet and 1 for other datasets, respectively. And we linearly anneal λ by $\lambda \leftarrow \lambda \times (1 - \frac{t}{T})$, where, t is the current iteration and T is the total number of iterations to anneal λ to zero. Here, $T = k \times (anneal.\ freq.)$, where $anneal.\ freq.$ is 48,000 in this work. We search the $k = \{1, 2, 3, 4, 5\}$ based on cross-validation over the validation sets of 8 training datasets and k is 5 (*i.e*. $T = 240,000$) for ImageNet, is 2 for Omniglot, Quick Draw, Fungi and is 1 for other datasets. For all experiments, early-stopping is performed based on cross-validation over the validations sets of 8 training datasets.

For the optimization of feature adaptation during meta-test stage, we initialize ϑ as an indentity matrix, which allows the NCC to use the original features produced by our universal network and optimize ϑ from a good start point. Similar to the optimization in [4], we optimize ϑ for 40 iterations using Adadelta [16] as optimizer with a learning rate of 0.1 for first eight datasets and 1 for the last five datasets.

2. More results

In this section, we first evaluate each single-domain model for few-shot classification on each test dataset. We then show complete results on varying-way five-shot and five-way one-shot settings. We also evaluate the effect of the adaptors for aligning features in knowledge distillation. As the code of Meta-dataset has been updated, we report results using the updated evaluation protocol from Meta-Dataset and compare our method with Cross-Transformer [3] and Transductive CNAPS [6] methods. Finally more qualitative results and global retrieval results are reported.

2.1. Complete results of single domain learning

To study the universal representation learning from multiple datasets, we train one network on each training dataset and use each single-domain network as the feature extractor and test it for few-shot classification in each dataset. This involves evaluating 8 single-domain networks on 13 datasets using Nearest Centroid Classifier (NCC). Table 2 shows the results of single domain learning models, where each

Test Dataset \ Train Dataset	ImageNet	Omniglot	Aircraft	Birds	Textures	Quick Draw	Fungi	Vgg Flower
ImageNet	$\mathbf{55.8 \pm 1.0}$	17.1 ± 0.6	21.7 ± 0.7	25.4 ± 0.8	24.2 ± 0.8	24.1 ± 0.8	32.9 ± 0.9	25.0 ± 0.8
Omniglot	67.4 ± 1.2	$\mathbf{93.2 \pm 0.5}$	58.2 ± 1.2	58.7 ± 1.4	57.3 ± 1.4	78.4 ± 1.0	57.6 ± 1.3	54.6 ± 1.3
Aircraft	49.5 ± 0.9	16.8 ± 0.5	$\mathbf{85.7 \pm 0.5}$	31.4 ± 0.8	26.0 ± 0.7	23.8 ± 0.6	31.0 ± 0.7	24.6 ± 0.6
Birds	$\mathbf{71.2 \pm 0.9}$	13.0 ± 0.6	19.9 ± 0.7	65.0 ± 0.9	19.6 ± 0.7	16.7 ± 0.7	42.8 ± 1.0	28.9 ± 0.8
Textures	$\mathbf{73.0 \pm 0.6}$	25.0 ± 0.5	38.6 ± 0.7	42.2 ± 0.7	54.9 ± 0.7	38.6 ± 0.6	54.1 ± 0.7	42.3 ± 0.7
Quick Draw	53.9 ± 1.0	51.0 ± 1.0	38.8 ± 1.0	38.2 ± 1.0	36.8 ± 0.9	$\mathbf{82.8 \pm 0.6}$	37.7 ± 0.9	39.7 ± 1.0
Fungi	41.6 ± 1.0	9.1 ± 0.5	14.9 ± 0.7	25.5 ± 0.8	15.6 ± 0.7	12.5 ± 0.6	$\mathbf{65.8 \pm 0.9}$	23.3 ± 0.8
VGG Flower	$\mathbf{87.0 \pm 0.6}$	23.8 ± 0.6	45.5 ± 0.8	62.9 ± 0.8	44.4 ± 0.8	33.4 ± 0.7	79.6 ± 0.7	78.3 ± 0.7
Traffic Sign	$\mathbf{47.4 \pm 1.1}$	15.1 ± 0.7	30.8 ± 0.9	31.0 ± 0.9	38.8 ± 1.1	31.1 ± 0.9	28.0 ± 0.9	30.4 ± 0.9
MSCOCO	$\mathbf{53.5 \pm 1.0}$	12.9 ± 0.6	22.5 ± 0.8	25.1 ± 0.9	23.7 ± 0.8	21.3 ± 0.8	32.5 ± 1.0	25.7 ± 0.8
MNIST	78.1 ± 0.7	$\mathbf{89.8 \pm 0.5}$	68.0 ± 0.8	73.0 ± 0.7	64.5 ± 0.8	88.2 ± 0.5	62.2 ± 0.8	72.1 ± 0.7
CIFAR-10	$\mathbf{67.3 \pm 0.8}$	28.5 ± 0.6	41.2 ± 0.7	41.8 ± 0.8	36.9 ± 0.7	40.0 ± 0.7	38.8 ± 0.7	41.3 ± 0.8
CIFAR-100	$\mathbf{56.6 \pm 0.9}$	12.3 ± 0.6	24.3 ± 0.9	28.8 ± 0.9	24.2 ± 0.9	23.4 ± 0.8	25.2 ± 0.9	29.1 ± 1.0

Table 2. Results of all single domain learning models. Mean accuracy and 95% confidence interval are reported. The first eight datasets are seen during training and the last five datasets are unseen for test only.

	Five-Shot				Five-Way One-Shot			
Test Dataset	Simple CNAPS [2]	SUR [4]	URT [8]	Ours	Simple CNAPS [2]	SUR [4]	URT [8]	Ours
ImageNet	47.2 ± 1.0	46.7 ± 1.0	48.6 ± 1.0	$\mathbf{49.4 \pm 1.0}$	42.6 ± 0.9	40.7 ± 1.0	47.4 ± 1.0	$\mathbf{49.6 \pm 1.1}$
Omniglot	95.1 ± 0.3	95.8 ± 0.3	$\mathbf{96.0 \pm 0.3}$	$\mathbf{96.0 \pm 0.3}$	93.1 ± 0.5	93.0 ± 0.7	95.6 ± 0.5	$\mathbf{95.8 \pm 0.5}$
Aircraft	74.6 ± 0.6	82.0 ± 0.6	81.2 ± 0.6	$\mathbf{84.8 \pm 0.5}$	65.8 ± 0.9	67.1 ± 1.4	77.9 ± 0.9	$\mathbf{79.6 \pm 0.9}$
Birds	69.6 ± 0.7	62.8 ± 0.9	71.2 ± 0.7	$\mathbf{76.0 \pm 0.6}$	67.9 ± 0.9	59.2 ± 1.0	70.9 ± 0.9	$\mathbf{74.9 \pm 0.9}$
Textures	57.5 ± 0.7	60.2 ± 0.7	65.2 ± 0.7	$\mathbf{69.1 \pm 0.6}$	42.2 ± 0.8	42.5 ± 0.8	49.4 ± 0.9	$\mathbf{53.6 \pm 0.9}$
Quick Draw	70.9 ± 0.6	79.0 ± 0.5	$\mathbf{79.2 \pm 0.5}$	78.2 ± 0.5	70.5 ± 0.9	$\mathbf{79.8 \pm 0.9}$	79.6 ± 0.9	79.0 ± 0.8
Fungi	50.3 ± 1.0	66.5 ± 0.8	66.8 ± 0.9	$\mathbf{70.0 \pm 0.8}$	58.3 ± 1.1	64.8 ± 1.1	71.0 ± 1.0	$\mathbf{75.2 \pm 1.0}$
VGG Flower	86.5 ± 0.4	76.9 ± 0.6	82.4 ± 0.5	$\mathbf{89.3 \pm 0.4}$	$\mathbf{79.9 \pm 0.7}$	65.0 ± 1.0	72.7 ± 0.9	$\mathbf{79.9 \pm 0.8}$
Traffic Sign	55.2 ± 0.8	44.9 ± 0.9	45.1 ± 0.9	$\mathbf{57.5 \pm 0.8}$	55.3 ± 0.9	44.6 ± 0.9	52.6 ± 0.9	$\mathbf{57.9 \pm 0.9}$
MSCOCO	49.2 ± 0.8	48.1 ± 0.8	52.3 ± 0.8	$\mathbf{56.1 \pm 0.8}$	48.8 ± 0.9	47.8 ± 1.1	56.9 ± 1.1	$\mathbf{59.1 \pm 1.0}$
MNIST	88.9 ± 0.4	$\mathbf{90.1 \pm 0.4}$	86.5 ± 0.5	89.7 ± 0.4	$\mathbf{80.1 \pm 0.9}$	77.0 ± 0.9	75.6 ± 0.9	78.7 ± 0.9
CIFAR-10	$\mathbf{66.1 \pm 0.7}$	50.3 ± 1.0	61.4 ± 0.7	66.0 ± 0.7	50.3 ± 0.9	35.8 ± 0.8	47.3 ± 0.9	$\mathbf{54.7 \pm 0.9}$
CIFAR-100	53.8 ± 0.9	46.4 ± 0.9	52.5 ± 0.9	$\mathbf{57.0 \pm 0.9}$	53.8 ± 0.9	42.9 ± 1.0	54.9 ± 1.1	$\mathbf{61.8 \pm 0.9}$
Average Rank	3.1	3.0	2.5	1.3	2.8	3.5	2.4	1.2

Table 3. Results of Five-Way One-Shot and Varying-Way Five-Shot settings. Mean accuracies are reported and the results with confidence interval are reported.

column present the mean accuracy and 95% confidence interval of a single-domain network trained on one dataset (*e.g.* ImageNet) and evaluated on 13 test datasets. The average accuracy and 95% confidence intervals computed over 600 few-shot tasks. The numbers in bold indicate that a method has the best accuracy per dataset.

As shown in Table 2, the feature of the ImageNet model generalizes well and achieves the best results on four out of eight seen datasets, *e.g.* ImageNet, Birds, Texture, VGG Flower and four out of five previously unseen datasets, *e.g.* Traffic Sign, MSCOCO, CIFAR-10, CIFAR-100. The models trained on Omniglot, Aircraft, Quick Draw, and Fungi perform the best on the corresponding datasets while the Omniglot model also generalizes well to MNIST which has the similar style images to Omniglot. We then pick the best performing model, forming the best single-domain model (Best SDL) which serves a very competitive baseline for universal representation learning.

2.2. Effect of adaptors in knowledge distillation

In this section, we evaluate our method with adaptors or without adaptors for aligning features when we use CKA for knowledge distillation. From Table 4, We can see that using adaptors can improve the performance, such as Birds (+1.7) and VGG Flower (+3.6), MSCOCO (+1.3). This indicates that the adaptors A_θ help align features between multi-domain and single-domain learning networks which are learned from very different domains.

Test Dataset	Ours (CKA w/o A_θ)	Ours (CKA)
ImageNet	58.3 ± 1.0	$\mathbf{59.0 \pm 1.0}$
Omniglot	94.4 ± 0.4	$\mathbf{94.7 \pm 0.4}$
Aircraft	$\mathbf{88.9 \pm 0.5}$	$\mathbf{88.9 \pm 0.4}$
Birds	78.7 ± 0.8	$\mathbf{80.4 \pm 0.7}$
Textures	$\mathbf{74.8 \pm 0.7}$	74.5 ± 0.7
Quick Draw	$\mathbf{82.1 \pm 0.6}$	81.9 ± 0.6
Fungi	65.4 ± 0.9	$\mathbf{66.4 \pm 0.9}$
VGG Flower	87.5 ± 0.6	$\mathbf{91.3 \pm 0.5}$
Traffic Sign	$\mathbf{63.3 \pm 1.1}$	63.2 ± 1.1
MSCOCO	55.3 ± 1.0	$\mathbf{56.6 \pm 1.0}$
MNIST	$\mathbf{94.9 \pm 0.4}$	94.7 ± 0.4
CIFAR-10	73.4 ± 0.7	$\mathbf{73.8 \pm 0.7}$
CIFAR-100	61.8 ± 1.0	$\mathbf{62.1 \pm 1.0}$

Table 4. Results of our method using CKA, CKA without adaptors (*i.e.* A_θ). Mean accuracy and 95% confidence interval are reported. Here, Ours (CKA w/o A_θ) indicates that adaptors are not applied for aligning features. All results are obtained with feature adaptation during meta-test stage.

Test Dataset	ImageNet				Omniglot				Aircraft				Birds				Textures				Quick Draw				Fungi				VGG Flower			
Recall@k	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8
Sum	22.1	30.3	39.6	50.0	84.7	91.8	95.8	97.8	69.7	80.7	88.6	94.5	45.9	59.7	72.0	84.1	66.3	78.2	87.3	94.0	77.4	84.3	89.1	92.1	31.9	42.9	54.0	65.4	85.1	92.1	96.7	98.6
Concate	20.2	28.0	36.9	47.8	84.4	91.5	95.8	97.8	44.3	58.1	71.1	82.9	35.5	48.8	62.8	76.0	68.8	78.2	87.3	93.9	73.0	80.8	86.2	90.6	30.7	40.4	51.8	63.0	83.4	91.3	95.2	98.2
MDL	29.8	39.6	49.9	60.9	**89.8**	**94.3**	96.8	98.2	80.3	87.1	92.5	95.9	63.2	75.9	84.7	91.6	67.0	77.1	85.4	92.9	79.5	85.4	89.7	92.8	40.2	51.7	63.0	72.4	86.9	93.3	96.6	98.4
Simple CNAPS [2]	34.0	43.8	54.4	65.1	84.9	91.6	95.5	97.5	70.5	82.5	91.3	96.1	55.9	70.5	82.0	90.2	64.8	76.9	**87.6**	**94.4**	75.3	83.0	88.0	91.7	29.1	39.0	49.6	61.5	88.1	94.1	**97.6**	**99.2**
Ours	**36.1**	**46.2**	**56.3**	**66.6**	89.7	**94.3**	**97.2**	**98.3**	**83.3**	**90.4**	**93.7**	**96.3**	**66.7**	**78.9**	**87.9**	**94.1**	**70.2**	**80.8**	87.5	93.8	**79.9**	**86.5**	**90.5**	**93.2**	**44.5**	**56.2**	**67.3**	**76.4**	**90.0**	**94.6**	97.5	98.9

Table 5. Global retrieval performance on Meta-Dataset (seen datasets). In addition to few-shot learning experiments, we evaluate our method in a non-episodic retrieval task to further compare the generalization ability of our universal representations.

Test Dataset	Traffic Sign				MSCOCO				MNIST				CIFAR-10				CIFAR-100			
Recall@k	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8
Sum	94.6	97.2	98.5	**99.3**	62.6	71.2	78.9	85.0	98.3	99.2	**99.6**	**99.8**	54.0	68.9	81.9	90.6	27.8	37.4	48.4	60.4
Concate	**95.1**	**97.3**	**98.6**	99.2	60.7	69.8	77.4	83.6	**98.7**	**99.3**	**99.6**	**99.8**	49.7	65.3	79.4	88.9	25.4	34.6	45.3	57.2
MDL	89.5	94.1	96.6	98.3	63.6	72.6	79.9	86.0	97.6	98.8	99.2	99.6	58.9	72.9	84.1	92.2	31.6	42.0	53.4	64.8
Simple CNAPS [2]	79.9	86.9	92.6	96.2	65.2	73.8	81.1	86.6	97.5	98.8	99.3	99.7	**66.2**	**79.3**	**88.5**	**94.7**	33.2	44.2	57.3	68.7
Ours	87.9	93.0	96.1	98.2	**67.4**	**76.3**	**83.0**	**88.5**	97.0	98.4	99.1	99.5	62.1	76.5	86.0	93.3	**35.1**	**46.1**	**57.8**	**69.0**

Table 6. Global retrieval performance on Meta-Dataset (unseen datasets). In addition to few-shot learning experiments, we evaluate our method in a non-episodic retrieval task to further compare the generalization ability of our universal representations.

2.3. Complete results of varying-way five-shot and five-way one-shot

We further analyze our method for 5-shot setting with varying number of categories. To this end, we follow the setting in [3], compare our method to the best three state-of-the-art methods including Simple CNAPS, SUR and URT. In this setting, we sample a varying number of ways in Meta-Dataset the same as the standard setting but a fixed number of shots to form balanced support and query sets. The mean accuracy and 95% confidence interval of our method and compared approaches are depicted in Table 3. As shown in Table 3, overall performance for all methods decreases in most datasets compared to results in the conventional setting shown in Table 1 in the paper, indicating that this is a more challenging setting. It is due to that five-shot setting samples much less support images than the standard setting. While both Simple CNAPS and SUR obtain 3.1 and 3.0 average rank, respectively. SUR performs the best on MNIST, Simple CNAPS outperforms others on CIFAR-10 and URT is top-1 on Quick Draw. Ours still achieves significant better performance than other methods on the rest ten datasets.

Results in five-way one-shot setting. Next we test an extremely challenging five-way one-shot setting on Meta-Dataset. For each task, only one image per class is seen as support set. This setting is often used in evaluating different methods in a single domain [7, 10, 14], while we adopt it for multiple domains. As shown in Table 3, our method achieves consistent gain as observed in previous two settings, which validates the importance of good universal representations in case of limited labeled samples in meta-test. Interestingly, Simple CNAPS achieves better rank than SUR in this setting, which is opposite in previous settings.

Results evaluated with updated evaluation protocol. As the code from Meta-dataset has been updated, we evaluate all methods with the updated evaluation protocol from the Meta-dataset [1] and report the results [2] in Table 7. As shown in Table 7, the update does not affect much on the results and our method rank 1.2 in average and the state-of-the-art methods SUR and URT rank 5.4 and 4.2, respectively. More specifically, we obtain significantly better results than the second best approach on Aircraft (+4.1), Birds (+1.3), Texture (+4.1), and Fungi (+2.9) for seen domains and Traffic Sign (+4.1) and MSCOCO (+4.3). The results show that jointly learning a single set of representations provides better generalization ability than fusing the ones from multiple single-domain feature extractors as done in SUR and URT. Notably, our method requires less parameters and less computations to run during inference than SUR and URT, as it runs only one universal network to extract features, while both SUR and URT need to pass the query set to multiple single-domain network.

Comparison to Cross-Transformer [3] and Transductive CNAPS [6]. Here we compare our method to CTX [3] and TCNAPS [1][3] in Table 8. Note that TCNAPS and CTX are *not directly comparable* to our method. TCNAPS extends the Simple CNAPS [2] to a more favorable *transductive inference* setting and exploits the query set at test time which is in contrast to the inductive learning in our submission. CTX [3] focuses on learning from *a single domain* (ImageNet), while our method is proposed to learn a single set of universal representation from multiple domains. In addition, CTX is built on a heavier network (ResNet-34) and larger resolution images (224×224) than the one (ResNet-18, 84×84 images)

[1] As mentioned in `https://github.com/google-research/meta-dataset/issues/54`, we also set the shuffle_buffer_size as 1000 to evaluate all methods and report the results in Table 7. This change does not affect much on the results as the datasets we used were shuffled using the latest data convert code from Meta-Dataset.

[2] Results of Proto-MAML [13], BOHB-E [12], and CNAPS [11] are obtained from Meta-Dataset. The results of Simple CNAPS [2] are reproduced by the authors and reported at `https://github.com/peymanbateni/simple-cnaps`. We reproduce the results of SUR [4] and URT [8] with the updated evaluation protocol for fair comparison.

[3] Results of CTX and TCNAPS are from `https://github.com/google-research/meta-dataset`

Test Dataset	Proto-MAML [13]	BOHB-E [12]	CNAPS [11]	Simple CNAPS [2]	SUR [4]	URT [8]	Best SDL	MDL	Ours
ImageNet	46.5 ± 1.1	51.9 ± 1.1	50.8 ± 1.1	56.5 ± 1.1	54.5 ± 1.1	55.0 ± 1.1	54.3 ± 1.1	52.9 ± 1.2	$\mathbf{57.5 \pm 1.1}$
Omniglot	82.7 ± 1.0	67.6 ± 1.2	91.7 ± 0.5	91.9 ± 0.6	93.0 ± 0.5	93.3 ± 0.5	93.8 ± 0.5	93.7 ± 0.5	$\mathbf{94.5 \pm 0.4}$
Aircraft	75.2 ± 0.8	54.1 ± 0.9	83.7 ± 0.6	83.8 ± 0.6	84.3 ± 0.5	84.5 ± 0.6	84.5 ± 0.5	84.9 ± 0.5	$\mathbf{88.6 \pm 0.5}$
Birds	69.9 ± 1.0	70.7 ± 0.9	73.6 ± 0.9	76.1 ± 0.9	70.4 ± 1.1	75.8 ± 0.8	70.6 ± 0.9	79.2 ± 0.8	$\mathbf{80.5 \pm 0.7}$
Textures	68.2 ± 0.8	68.3 ± 0.8	59.5 ± 0.7	70.0 ± 0.8	70.5 ± 0.7	70.6 ± 0.7	72.1 ± 0.7	70.9 ± 0.8	$\mathbf{76.2 \pm 0.7}$
Quick Draw	66.8 ± 0.9	50.3 ± 1.0	74.7 ± 0.8	78.3 ± 0.7	81.6 ± 0.6	82.1 ± 0.6	$\mathbf{82.6 \pm 0.6}$	81.7 ± 0.6	81.9 ± 0.6
Fungi	42.0 ± 1.2	41.4 ± 1.1	50.2 ± 1.1	49.1 ± 1.2	65.0 ± 1.0	63.7 ± 1.0	65.9 ± 1.0	63.2 ± 1.1	$\mathbf{68.8 \pm 0.9}$
VGG Flower	88.7 ± 0.7	87.3 ± 0.6	88.9 ± 0.5	91.3 ± 0.6	82.2 ± 0.8	88.3 ± 0.6	86.7 ± 0.6	88.7 ± 0.6	$\mathbf{92.1 \pm 0.5}$
Traffic Sign	52.4 ± 1.1	51.8 ± 1.0	56.5 ± 1.1	59.2 ± 1.0	49.8 ± 1.1	50.1 ± 1.1	47.1 ± 1.1	49.2 ± 1.0	$\mathbf{63.3 \pm 1.2}$
MSCOCO	41.7 ± 1.1	48.0 ± 1.0	39.4 ± 1.0	42.4 ± 1.1	49.4 ± 1.1	48.9 ± 1.1	49.7 ± 1.0	47.3 ± 1.1	$\mathbf{54.0 \pm 1.0}$
MNIST	-	-	-	94.3 ± 0.4	$\mathbf{94.9 \pm 0.4}$	90.5 ± 0.4	91.0 ± 0.5	94.2 ± 0.4	94.5 ± 0.5
CIFAR-10	-	-	-	$\mathbf{72.0 \pm 0.8}$	64.2 ± 0.9	65.1 ± 0.8	65.4 ± 0.8	63.2 ± 0.8	71.9 ± 0.7
CIFAR-100	-	-	-	60.9 ± 1.1	57.1 ± 1.1	57.2 ± 1.0	56.2 ± 1.0	54.7 ± 1.1	$\mathbf{62.6 \pm 1.0}$
Average Rank	7.7	8.0	6.8	4.8	5.4	4.2	4.8	4.8	1.2

Table 7. Comparison to baselines and state-of-the-art methods on Meta-Dataset. Mean accuracy, 95% confidence interval are reported. The first eight datasets are seen during training and the last five datasets are unseen and used for test only. Average rank is computed according to first 10 datasets as some methods do not report results on last three datasets.

Test Dataset	CTX [3]	TCNAPS [1]	Ours
ImageNet	$\mathbf{62.8 \pm 1.0}$	57.9 ± 1.1	57.5 ± 1.1
Omniglot	82.2 ± 1.0	94.3 ± 0.4	$\mathbf{94.5 \pm 0.4}$
Aircraft	79.5 ± 0.9	84.7 ± 0.5	$\mathbf{88.6 \pm 0.5}$
Birds	$\mathbf{80.6 \pm 0.9}$	78.8 ± 0.7	80.5 ± 0.7
Textures	75.6 ± 0.6	66.2 ± 0.8	$\mathbf{76.2 \pm 0.7}$
Quick Draw	72.7 ± 0.8	77.9 ± 0.6	$\mathbf{81.9 \pm 0.6}$
Fungi	51.6 ± 1.1	48.9 ± 1.2	$\mathbf{68.8 \pm 0.9}$
VGG Flower	$\mathbf{95.3 \pm 0.4}$	92.3 ± 0.4	92.1 ± 0.5
Traffic Sign	$\mathbf{82.7 \pm 0.8}$	59.7 ± 1.1	63.3 ± 1.2
MSCOCO	$\mathbf{59.9 \pm 1.0}$	42.5 ± 1.1	54.0 ± 1.0
MNIST	-	-	$\mathbf{94.5 \pm 0.5}$
CIFAR-10	-	-	$\mathbf{71.9 \pm 0.7}$
CIFAR-100	-	-	$\mathbf{62.6 \pm 1.0}$

Table 8. Comparison to CrossTransformer (CTX) and TransductiveCNAPS (TCNAPS) on Meta-Dataset. Mean accuracy, 95% confidence interval are reported. The first eight datasets are seen during training and the last five datasets are unseen and used for test only. Note that TCNAPS and CTX are *not directly comparable* to our method.

in ours. Nevertheless, as shown in Table 8, our method still outperforms TCNAPS and CTX on most of the domains (8 out of 10 and 5 out of 10 respectively). Both the transductive learning in TCNAPS and the cross-attention mechanism in CTX are potentially orthogonal to our universal representation learning and thus can be incorporated to ours, while we leave this as future work. We will include the results and detailed discussion in the final version.

2.4. Qualitatively results

We qualitatively analyze our method and compare it to the vanilla multi-domain leanring (MDL) baseline, Simple CNAPS [2], SUR [4] and URT [8] in Figs. 1 to 13 by illustrating the nearest neighbors in all test datasets given a query image. It is clear that our method produces more correct neighbors than other methods. While other methods retrieves images with more similar colors, shapes and backgrounds, *e.g.* in Figs. 9 and 10, our method is able to retrieve semantically similar images. It again suggests that our method is able to learn more useful and general representations.

2.5. Complete global retrieval results

Here we go beyond the few-shot classification experiments and evaluate the generalization ability of our representations that are learned in the multi-domain network in a retrieval task, inspired from metric learning literature [9, 15]. To this end, for each test image, we find the nearest images in entire test set in the feature space and test whether they correspond to the same category. For evaluation metric, we use Recall@k which considers the predictions with one of the k closest neighbors with the same label as positive. In Tables 5 and 6, we compare our method with Simple CNAPS in Recall@1, Recall@2, Recall@4 and Recall@8. URT and SUR require adaption using support set and no such adaptation in retrieval task is possible, we replace them with two baselines that concatenate or sum features from multiple domain-specific networks. Our method achieves the best performance in ten out of thirteen domains with significant gains in Aircraft, Birds, Textures and Fungi. This strongly suggests that our multi-domain representations are the key to the success of our method in the previous few-shot classification tasks.

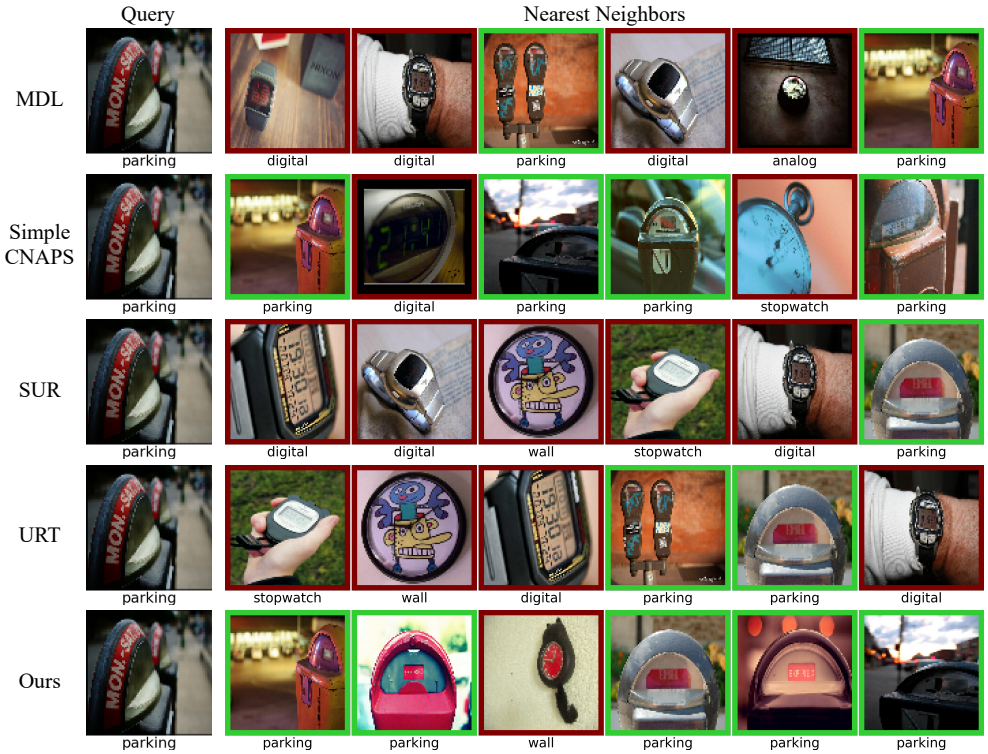

Figure 1. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in ImageNet. Green and red colors indicate correct and false predictions respectively.

References

[1] Peyman Bateni, Jarred Barber, Jan-Willem van de Meent, and Frank Wood. Enhancing few-shot image classification with unlabelled examples. *arXiv preprint arXiv:2006.12245*, 2020.

[2] Peyman Bateni, Raghav Goyal, Vaden Masrani, Frank Wood, and Leonid Sigal. Improved few-shot visual classification. In *CVPR*, pages 14493–14502, 2020.

[3] Carl Doersch, Ankush Gupta, and Andrew Zisserman. Crosstransformers: spatially-aware few-shot transfer. In *NeurIPS*, 2020.

[4] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. Selecting relevant features from a multi-domain representation for few-shot classification. In *ECCV*, pages 769–786, 2020.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[6] Jianan Jiang, Zhenpeng Li, Yuhong Guo, and Jieping Ye. A transductive multi-head model for cross-domain few-shot learning. *arXiv preprint arXiv:2006.11384*, 2020.

[7] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[8] Lu Liu, William Hamilton, Guodong Long, Jing Jiang, and Hugo Larochelle. A universal representation transformer layer for few-shot image classification. In *ICLR*, 2021.

[9] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, pages 4004–4012, 2016.

[10] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018.

[11] James Requeima, Jonathan Gordon, John Bronskill, Sebastian Nowozin, and Richard E Turner. Fast and flexible multi-task classification using conditional neural adaptive processes. In *CVPR*, 2019.

[12] Tonmoy Saikia, Thomas Brox, and Cordelia Schmid. Optimized generic feature learning for few-shot classification across domains. *arXiv preprint arXiv:2001.07926*, 2020.

[13] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. In *ICLR*, 2020.

[14] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In *NeurIPS*, 2016.

[15] Lu Yu, Vacit Oguz Yazici, Xialei Liu, Joost van de Weijer, Yongmei Cheng, and Arnau Ramisa. Learning metrics from teachers: Compact networks for image embedding. In *CVPR*, pages 2907–2916, 2019.

[16] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

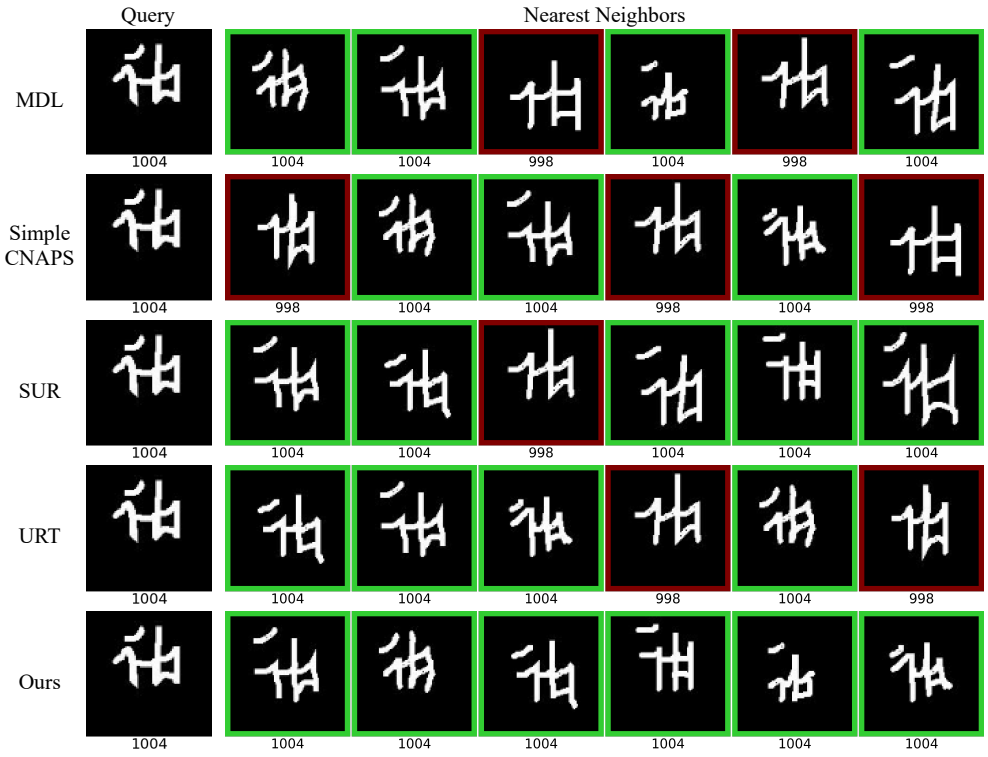

Figure 2. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in Omniglot. Green and red colors indicate correct and false predictions respectively.

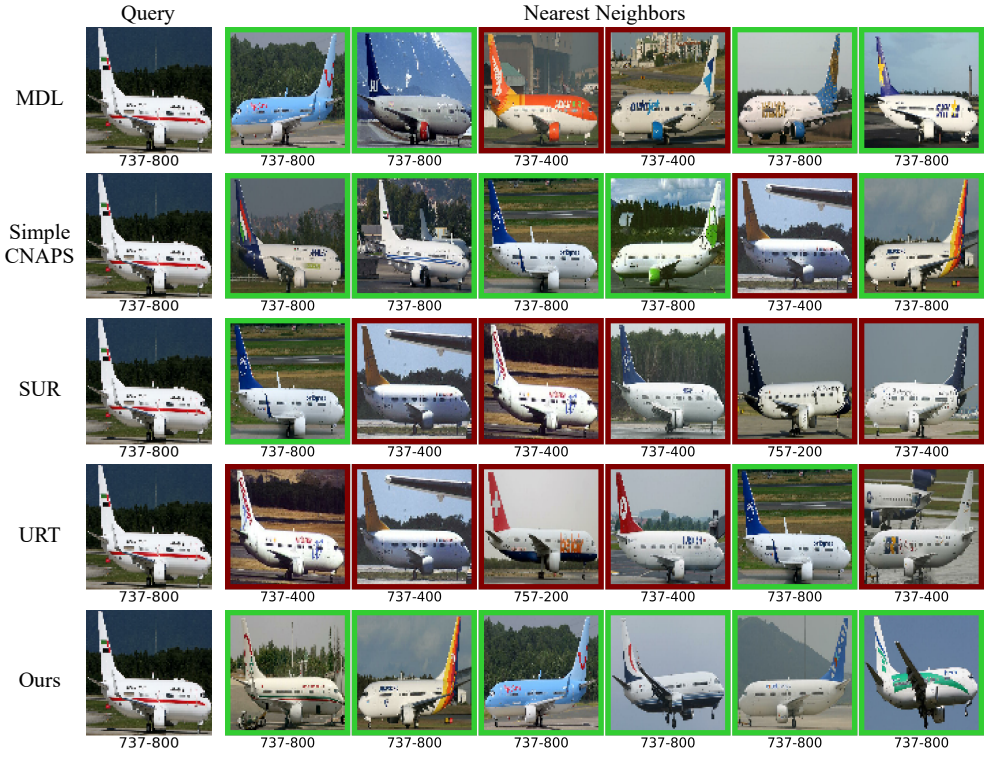

Figure 3. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in Aircraft. Green and red colors indicate correct and false predictions respectively.

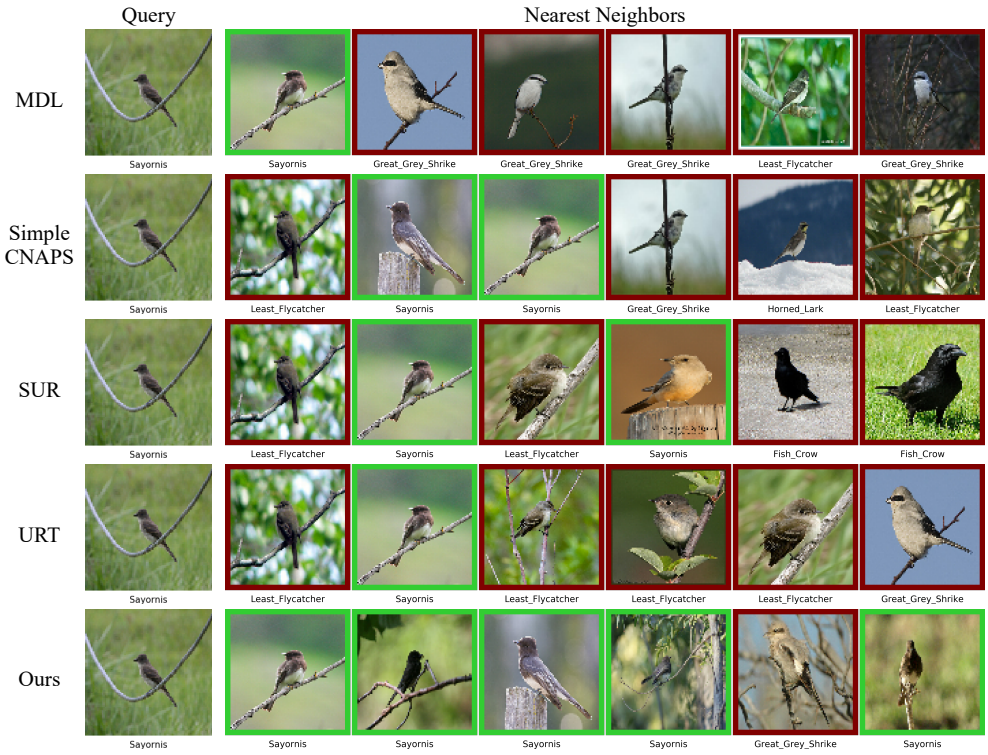

Figure 4. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in Birds. Green and red colors indicate correct and false predictions respectively.

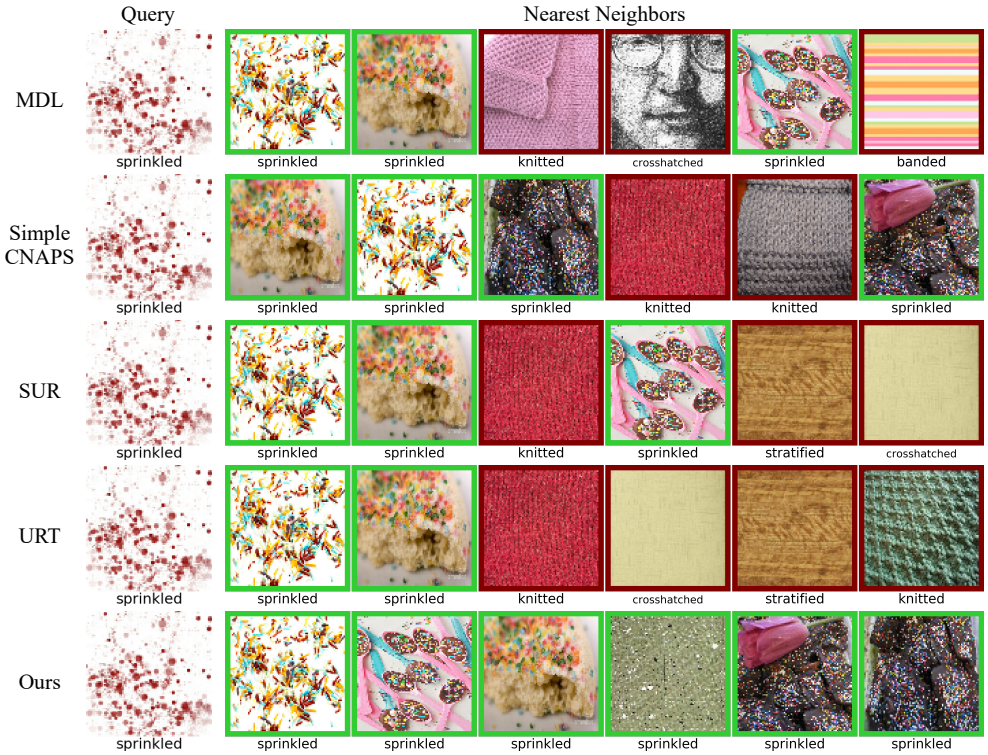

Figure 5. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in Textures. Green and red colors indicate correct and false predictions respectively.

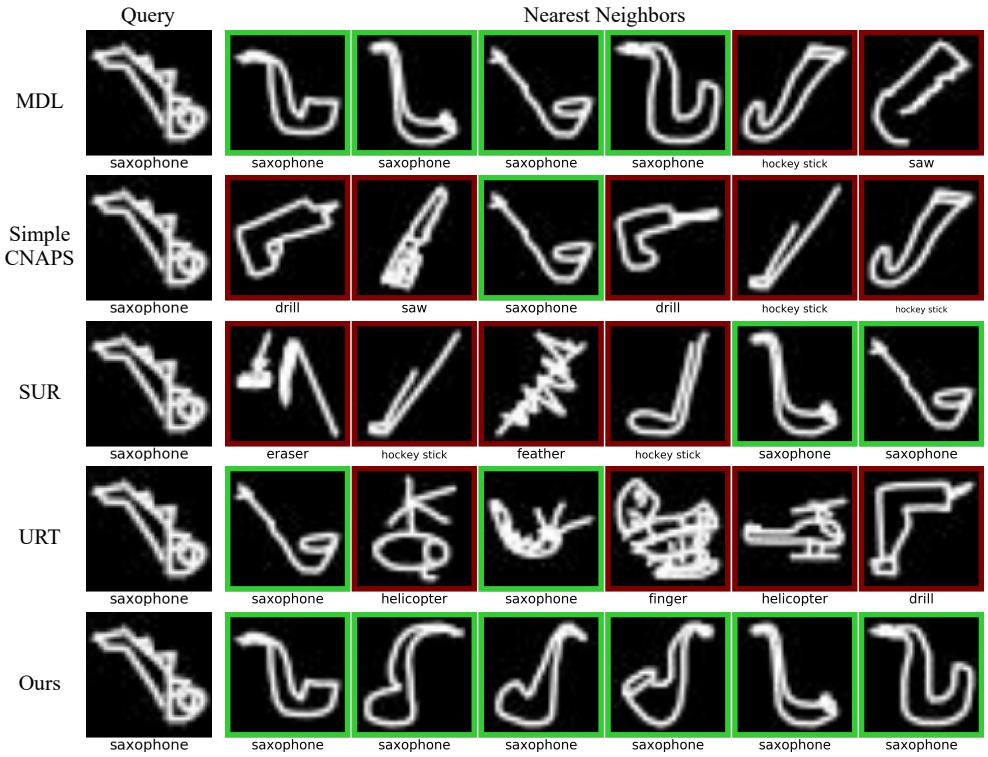

Figure 6. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in Quick Draw. Green and red colors indicate correct and false predictions respectively.

Figure 7. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in Fungi. Green and red colors indicate correct and false predictions respectively.

Figure 8. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in VGG Flower. Green and red colors indicate correct and false predictions respectively.

Figure 9. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in Traffic Sign. Green and red colors indicate correct and false predictions respectively.

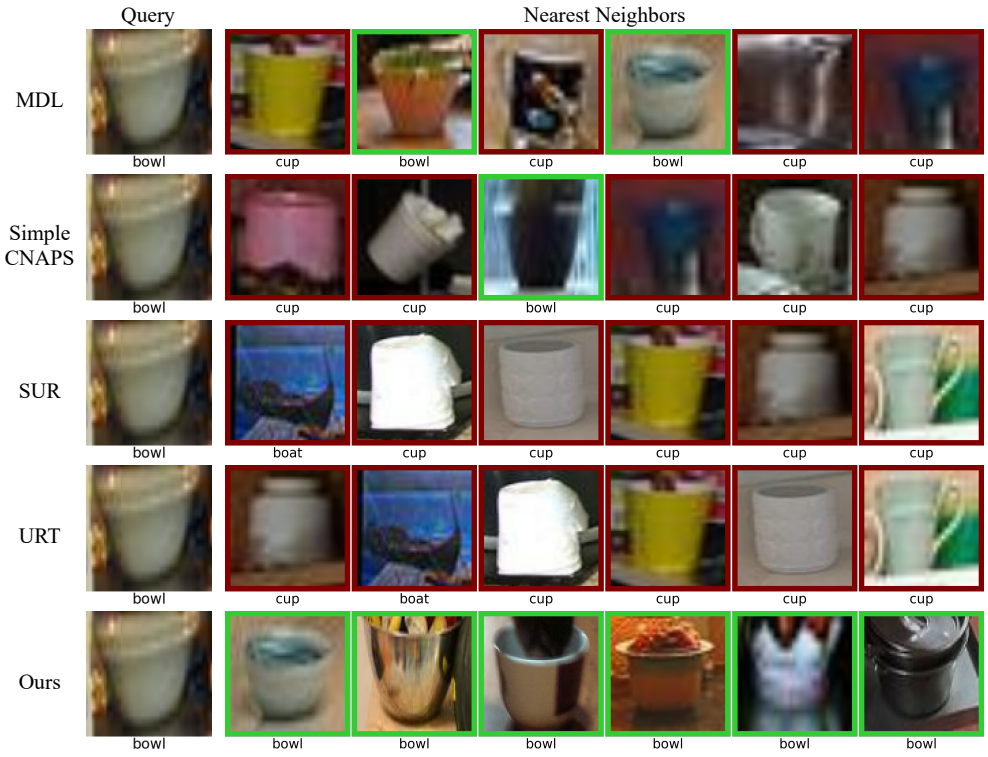

Figure 10. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in MSCOCO. Green and red colors indicate correct and false predictions respectively.

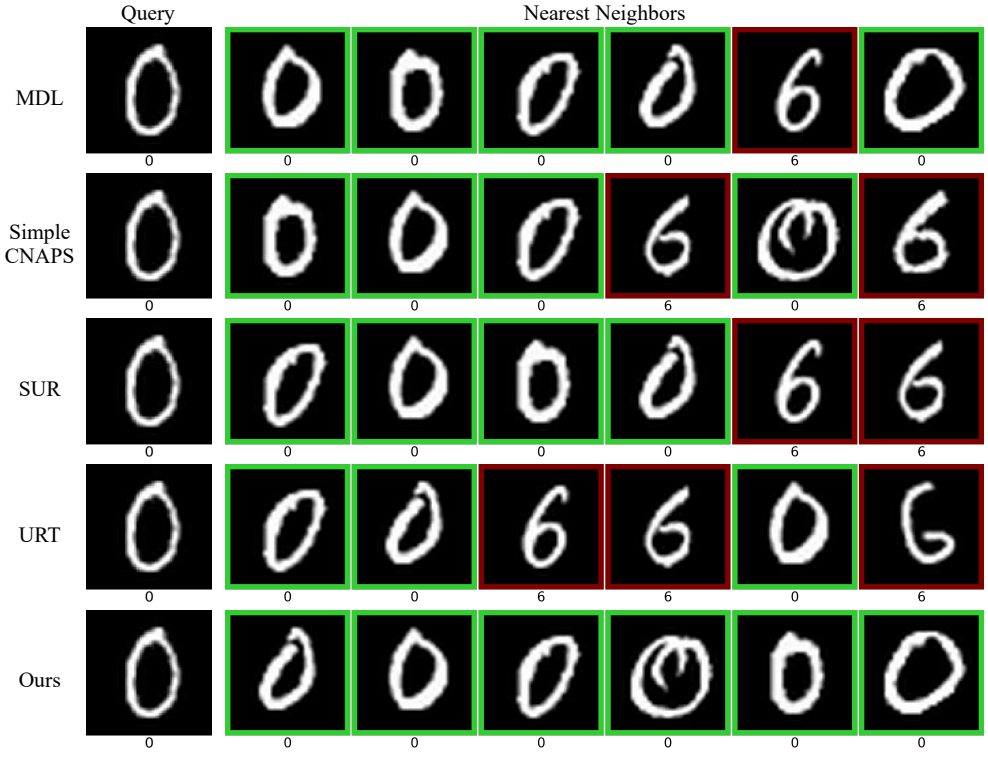

Figure 11. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in MNIST. Green and red colors indicate correct and false predictions respectively.

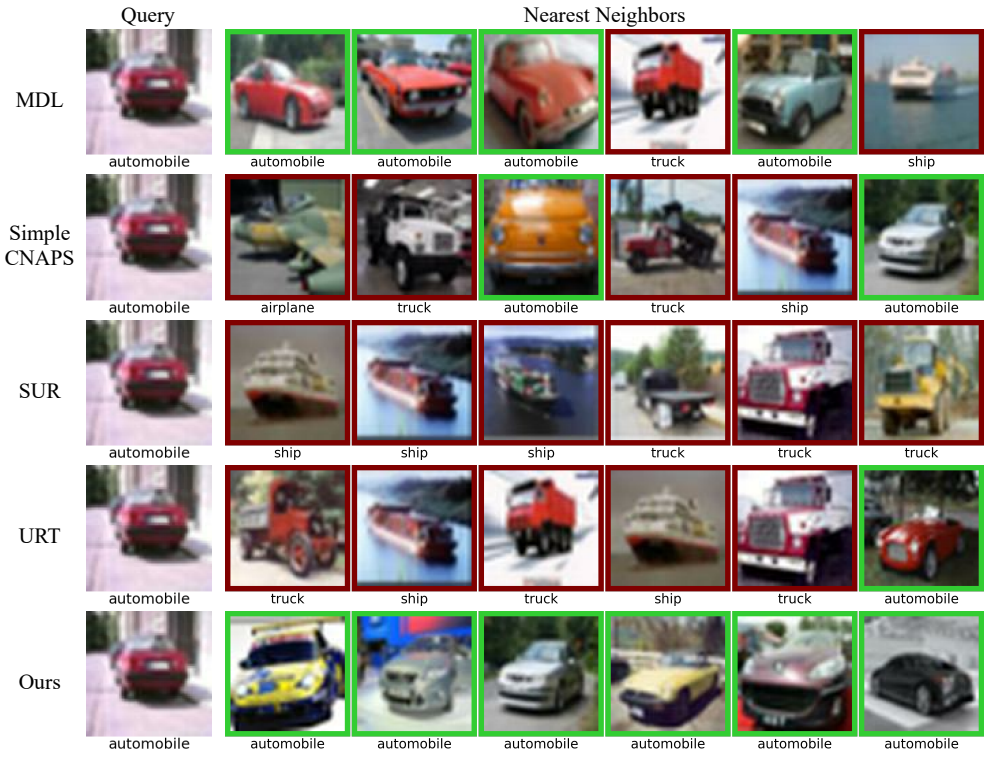

Figure 12. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in CIFAR-10. Green and red colors indicate correct and false predictions respectively.

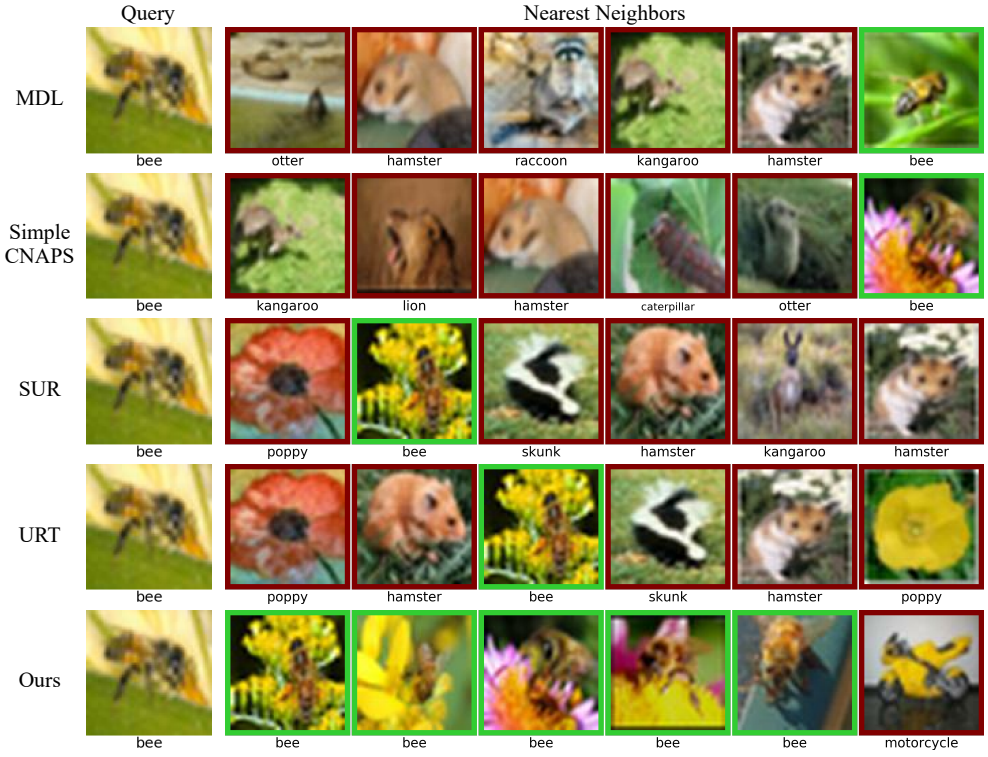

Figure 13. Qualitative comparison to MDL, Simple CNAPS [2], SUR [4], and URT [8] in CIFAR-100. Green and red colors indicate correct and false predictions respectively.