LVP-CLIP: Revisiting CLIP for Continual Learning with Label Vector Pool

Supplementary Material

1. Ablation study on the effect of the similarity function

The performance of various similarity functions is compared in Tab. 1. The LVP-CLIP-T, based only on text embeddings (which corresponds to the traditional CLIP approach), is very sensitive to the choice of the similarity function. It performs extremely poorly with L1 similarity, and the cosine similarity gives much better performance compared to L1 and L2 distances. Because of the poor match between the text-embedding and the L1 similarity, LVP-CLIP-IT presents a similar outcome, i.e. gives the lowest performance with L1 similarity and favors cosine similarity.

The LVP-CLIP-I, on the other hand, has very stable performance under different similarity functions, with L1 slightly outperforming the other two. Since L1 is also much easier to calculate than cosine similarity, we adopt it as the similarity function if the labeled vector is derived from image-embeddings. Otherwise, cosine similarity is used.

Similarity functions	L1	L2	Cos
LVP-CLIP-T	0.1	65.9	73.3
LVP-CLIP-I	**80.2**	80.0	80.1
LVP-CLIP-IT	76.8	81.8	**82.0**

Table 1. Ablation study on the effect of the similarity function, performed on CIFAR100 dataset.

2. Memory size of LVP

The memory size of the label vector pool (LVP) for each dataset is shown in Tab. 2. We represent the size by using different units, including the number of floating-point numbers and the equivalent number of images with dimensions 3x224x224 pixels. The total number of floating-point numbers needed to store the LVP can be calculated as $P \times K \times D$ where P is the pool size, K is the total number of classes, and $D = 768$. As can be seen, even with the four datasets combined, the memory needed for LVP is equivalent to only 13.6 frames of images (or 8.2MB).

3. Dataset without semantic labels

As discussed in our main paper, our proposed LVP enables CLIP [5]-based classification without solely relying on text embeddings. This is especially useful for datasets that lack meaningful text labels, such as CORe50. As shown in Fig. 1, CORe50 dataset has ten categories, represented by the 10 columns. Each category includes five distinct instances shown in 5 rows. Each instance is considered as an individual class. Hence, every small image in Fig. 1 is a unique class. These classes are labeled as $o1$, $o2$, ..., $o50$ with no inherent semantic meaning. Creating a set of meaningful semantic labels to distinguish these instances is challenging. Therefore, classifying the images by comparing their features to text embeddings of the labels becomes impractical. However, with the help of the proposed LVP, the LVP-I embeddings can be easily generated from the training images, facilitating accurate classification.

	CF100 [2]	IN100 [1]	DN [4]	CR50 [3]	CF100+IN100+DN+CR50
pool size P	1	1	6	8	mixed(1,1,6,8)
total class K	100	100	345	50	595
float number	76,800	76,800	1,589,760	307,200	2,050,560
images	0.5	0.5	10.6	2.0	13.6
Bytes	0.3MB	0.3MB	6.4MB	1.2MB	8.2MB

Table 2. Memory size required for LVP in terms of floating point numbers and the equivalent image size.

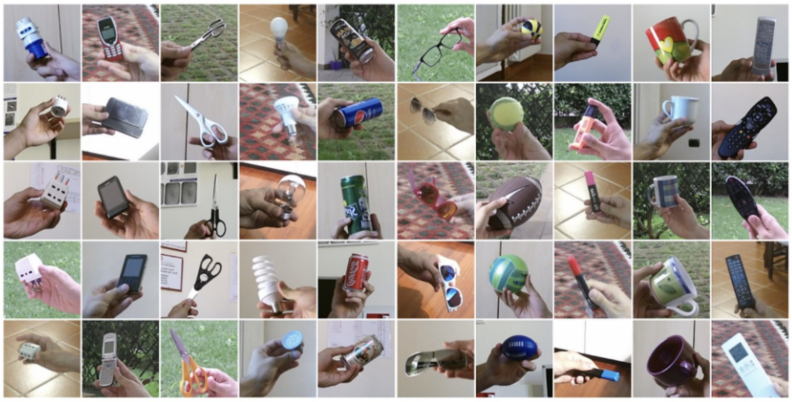
4. Unique advantages of LVP-CLIP

As illustrated in Fig. 2, parallel learning and retaining-free continual learning are two unique advantages of LVP-CLIP that most previous works cannot achieve. LVP-CLIP does not assume that the total number of classes is known in advance, and can learn new tasks by simply concatenating the label vector pools of each task. Moreover, since the LVPs of each task is completely independent of other tasks, the generation of LVPs can be processed on different machines in parallel.

5. Cross-Task Incremental Learning

Fig. 3 shows the T-SNE visualization of the label vector pools generated during cross-task incremental learning (CTIL). As can be seen, the LVPs for different datasets are well-separated in the feature space, with the exception of ImageNet100 and DomainNet datasets.

Tab. 3 provides a detailed comparison between the ideal and actual performance of the three variants of the LVP-CLIP for each learning task. Ideal performance is defined as the test accuracy for each task when the four datasets in the CTIL setting are learned and tested independently. Entries highlighted in red indicate tasks where ideal and actual performance are closely aligned (within a difference of 0.1). As shown in Fig. 3 , the LVPs of ImageNet100 and DomainNet are intermixed, and not well separated. This



```
[o1, ..., o5] -> plug adapters
[o6, ..., o10] -> mobile phones
[o11, ..., o15] -> scissors
[o16, ..., o20] -> light bulbs
[o21, ..., o25] -> cans
[o26, ..., o30] -> glasses
[o31, ..., o35] -> balls
[o36, ..., o40] -> markers
[o41, ..., o45] -> cups
[o46, ..., o50] -> remote controls
```

Figure 1. Images and labels from the CORe50 [3] dataset. There are a total of 50 classes but only 10 object names. Each object has five different instances as five classes. Since the class names are very close to each other as text, it is nearly impossible to separate them by zero-shot learning.

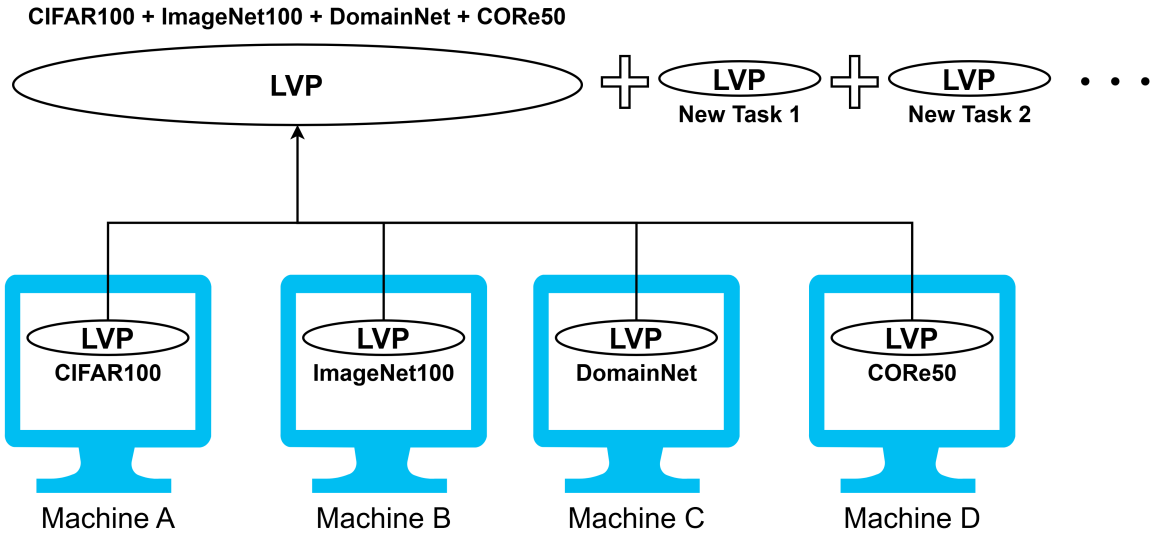


Figure 2. The illustration of the parallelizability and retaining-free continual learning ability of LVP-CLIP. Four machines conduct the experiments independently and in parallel and store the LVPs for each dataset. By simply concatenating all of LVPs, the continual learning of the four datasets is achieved. Moreover, as the new tasks arrive, the concatenation is simply repeated to store the knowledge from new tasks.

explains the higher offsets observed between the Ideal and actual performances on the ten IN100 test tasks and the DN-5 task(the 'real' domain) compared to the other test tasks. It is clear that, the distribution of ImageNet100 dataset is close to the 'real' domain of DomainNet.

6. Classes in ImageNet100

We have selected 100 classes from ImageNet [1] following [6]. The label ID and class names of the 100 classes are as follows: [15, 'American robin'], [45, 'Gila monster'], [54, 'eastern hog-nosed snake'], [57, 'garter snake'], [64, 'green mamba'], [72, 'European garden spider'], [90, 'lorikeet'], [99, 'goose'], [119, 'rock crab'], [120, 'fiddler crab'], [122, 'American lobster'], [131, 'little blue heron'], [137, 'American coot'],[151, 'Chihuahua'], [155, 'Shih Tzu'], [157, 'Papillon'], [158, 'toy terrier'], [166, 'Treeing Walker Coonhound'], [167, 'English foxhound'], [169, 'borzoi'], [176, 'Saluki'], [180, 'American Staffordshire Terrier'], [209, 'Chesapeake Bay Retriever'], [211, 'Vizsla'], [222, 'Kuvasz'], [228, 'Komondor'], [234, 'Rottweiler'], [236, 'Dobermann'], [242, 'Boxer'], [246, 'Great Dane'], [267, 'Standard Poodle'], [268, 'Mexican hairless dog [xoloitzcuintli]'], [272, 'coyote'], [275, 'African wild dog'], [277, 'red fox'],[281, 'tabby cat'],[299, 'meerkat'],[305, 'dung beetle'], [313, 'stick insect'], [317,

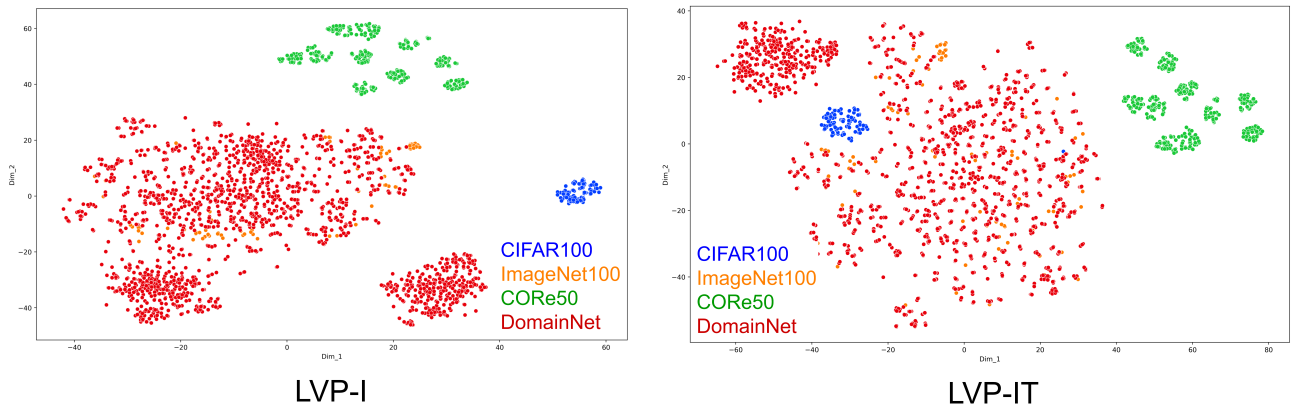


Figure 3. The T-SNE visualization of LVP-I and LVP-IT. Thanks to the remarkable feature extraction of CLIP, different dataset can be well-separated in the feature space except the ImageNet100 and DomainNet.

Test Tasks	CF100-1	CF100-2	CF100-3	CF100-4	CF100-5	CF100-6	CF100-7	CF100-8	CF100-9	CF100-10
Ideal-I	81.9	80.7	81.1	81.2	79.3	79.6	74.9	80.3	83.8	79.3
Ideal-IT	85.7	81.8	82.0	82.7	81.1	79.5	77.1	80.9	87.3	81.6
Ideal-C	83.1	81.4	82.5	79.9	80.2	79.3	75.3	80.5	86.1	81.1
LVP-CLIP-I	82.0	81.0	81.0	81.0	79.3	79.6	74.9	80.3	83.8	79.2
LVP-CLIP-IT	85.3	79.3	81.1	82.5	80.5	79.1	76.5	80.5	85.5	81.0
LVP-CLIP-C	84.7	79.4	81.7	80.8	76.6	78.5	73.6	80.6	84.5	79.1
Test Tasks	IN100-1	IN100-2	IN100-3	IN100-4	IN100-5	IN100-6	IN100-7	IN100-8	IN100-9	IN100-10
Ideal-I	93.0	84.2	88.2	97.0	94.4	92.0	92.0	90.6	91.2	95.4
Ideal-IT	93.2	86.2	90.2	96.4	94.6	92.0	93.6	90.2	92.6	96.2
Ideal-C	94.4	85.4	90.8	96.2	94.8	91.6	93.4	90.0	93.6	95.6
LVP-CLIP-I	89.0	81.0	87.2	94.6	84.0	88.6	84.0	85.0	86.2	85.6
LVP-CLIP-IT	90.0	82.6	89.4	95.2	85.6	86.8	81.2	81.8	83.2	84.6
LVP-CLIP-C	90.2	77.8	83.6	92.0	81.2	81.2	81.2	83.8	80.0	81.2
Test Tasks	DN-1	DN-2	DN-3	DN-4	DN-5	DN-6	CR50-1	CR50-2	CR50-3	ALL
Ideal-I	82.2	56.1	76.0	46.1	87.0	73.3	87.0	85.1	86.3	82.7
Ideal-IT	82.4	58.6	77.1	42.1	88.2	74.5	-	-	-	83.7
Ideal-C	80.1	60.7	75.5	33.5	87.7	74.5	90.3	88.8	89.6	83.3
LVP-CLIP-I	82.2	56.1	75.9	46.1	86.0	73.3	87.0	85.1	86.3	80.9
LVP-CLIP-IT	81.9	58.2	77.0	42.1	86.0	74.2	86.6	84.2	86.1	81.0
LVP-CLIP-C	79.9	59.8	74.3	31.0	84.4	73.9	89.6	87.5	89.5	79.4

Table 3. Results of all the cross-task incremental learning experiments. The ideal result is the test accuracy of each test task when the learning and testing are done on a given dataset independently. The LVP-CLIP results are the result of each test task across the four-datasets. The numbers for which the offset from the ideal performance is less than or equal to 0.1 are highlighted in red indicating nearly zero forgetting.

'leafhopper'], [331, 'hare'], [342, 'wild boar'], [368, 'gibbon'], [374, 'langur'], [407, 'ambulance'], [421, 'baluster handrail'],[431, 'bassinet'], [449, 'boathouse'], [452, 'poke bonnet'], [455, 'bottle cap'], [479, 'car wheel'], [494, 'bell or wind chime'], [498, 'movie theater'], [503, 'cocktail shaker'], [508, 'computer keyboard'], [544, 'Dutch oven'], [560, 'football helmet'], [570, 'gas mask or respirator'], [592, 'hard disk drive'],[593, 'harmonica'], [599, 'honeycomb'], [606, 'clothes iron'], [608, 'jeans'], [619, 'lampshade'],[620, 'laptop computer'], [653, 'milk can'], [659, 'mixing bowl'], [662, 'modem'], [665, 'moped'], [667, 'graduation cap'], [674, 'mousetrap'], [682, 'obelisk'],[703, 'park bench'], [708, 'pedestal'], [717, 'pickup truck'], [724, 'pirate ship'],[748, 'purse'], [758, 'fishing casting reel'], [765, 'rocking chair'], [766, 'rotisserie'],[772, 'safety pin'], [775, 'sarong'], [796, 'balaclava ski mask'], [798, 'slide rule'],[830, 'stretcher'], [854, 'front curtain'], [857, 'throne'], [858, 'tile roof'],

[872, ‘tripod’],[876, ‘hot tub’], [882, ‘vacuum cleaner’], [904, ‘window screen’], [908, ‘airplane wing’], [936, ‘cabbage’], [938, ‘cauliflower’], [953, ‘pineapple’], [959, ‘carbonara’],[960, ‘chocolate syrup’], [993, ‘gyromitra’], [994, ‘stinkhorn mushroom’]]

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1, 2

[2] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 1

[3] Vincenzo Lomonaco and Davide Maltoni. Core50: a new dataset and benchmark for continuous object recognition. In *Conference on robot learning*, pages 17–26. PMLR, 2017. 1, 2

[4] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1406–1415, 2019. 1

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1

[6] Runqi Wang, Xiaoyue Duan, Guoliang Kang, Jianzhuang Liu, Shaohui Lin, Songcen Xu, Jinhu Lü, and Baochang Zhang. Attriclip: A non-incremental learner for incremental knowledge learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3654–3663, 2023. 2