# Empowering Source-Free Domain Adaptation via MLLM-Guided Reliability-Based Curriculum Learning

## Supplementary Material

This supplementary material provides additional details and results for RCL. Sec. 7 includes training details and hyper-parameters. Sec. 8 discusses extended results, including ablations, latency and backbone analyses, and sensitivity to the number and quality of MLLM teachers. Secs. 9–10 detail the MMR algorithm and MLLM pseudo-labeling pipeline, including prompt templates, STS alignment, and reliability distributions. Sec. 11 extends related works to further position RCL within SFDA/DA literature. Finally, Sec. 12 extends our discussions on RCL's choice for MLLMs and practicality as a general multi-teacher framework.

## 7. Training Details

The parameters used in the training process of RCL are shown in Table 6. We use the Adam optimizer [21] for all experiments, conducted in PyTorch on NVIDIA A100 GPUs due to availability. However, A100s are not required for training. RCL only finetunes lightweight backbones such as ResNet-50, ResNet-18, and ViT-B/32. Importantly, MLLMs are used solely for zero-shot inference to generate pseudo-labels, which are cached once before adaptation. Unlike other VLM-based methods that require repeated inference or prompt tuning during training, RCL is efficient and requires no tuning or retraining of the MLLMs.

Table 6. Parameters for different datasets and methods

| | Office-Home | | | DomainNet | | | VisDA | | |
|---|---|---|---|---|---|---|---|---|---|
| | RKT | SMKE | MMR | RKT | SMKE | MMR | RKT | SMKE | MMR |
| learning rate | 1e-04 | 1e-05 | 1e-05 | 1e-05 | 1e-05 | 1e-05 | 1e-05 | 1e-06 | 1e-06 |
| $\tau$ | – | 0.7 | 0.95 | – | 0.7 | 0.9 | – | 0.7 | 0.6 |
| batch size | 64 | 256 | 128 | 64 | 256 | 64 | 64 | 256 | 256 |
| max iter | 3000 | 5000 | 5000 | 8000 | 10000 | 5000 | 6000 | 6000 | 5000 |

## 8. Extended Results

### 8.1. Main Results

Tables 7, 8, and 9 provide extended results for the Office-Home, DomainNet, and VisDA datasets, respectively. RCL achieves the highest average accuracy across all datasets. For Office-Home and DomainNet, RCL attains the highest performance across all 12 adaptation tasks compared to existing methods. For VisDA, RCL achieves top results in 9 out of 12 categories and ranks second in the remaining three. In most scenarios, the second-best performance is primarily achieved using either DIFO-C-B32 [55] or PSAT-GDA [53]. As previously discussed in Section 5.1, DIFO employs task-specific prompt learning and requires tuning of the CLIP encoder. Conversely, PSAT-GDA uses a transformer-based SFDA approach, specifically training transformers for source guidance and domain alignment. Our method, RCL, does not require tuning or training of any large VLMs or transformers, instead relying solely on the zero-shot inference capabilities of MLLMs. Throughout the results, RCL consistently achieves significant performance improvements over existing methods.

### 8.2. Extended Ablation Studies

We provide additional ablation results, including hyper-parameter sensitivity (Sec. 8.2.1), inference latency for RCL and the MLLM teachers (Sec. 8.2.2), robustness to different MLLM/VLM teacher choices (Sec. 8.2.3), the impact of each RCL component (RKT, SMKE, MMR; Sec. 8.2.4), and knowledge transfer to smaller backbones (Sec. 8.2.5).

#### 8.2.1. Sensitivity of hyper-parameters

Fig. 9 shows the effect of the confidence threshold $\tau$ in SMKE, where $\tau$ values influence model performance by balancing self-correction with MLLM guidance. Higher $\tau$ values increase reliance on high-confidence pseudo-labels from the target model, while lower values depend more on MLLM-generated pseudo-labels. Fig. 9 highlights the optimal $\tau$ that enhances adaptation performance by effectively leveraging both reliable and less reliable pseudo-labels.

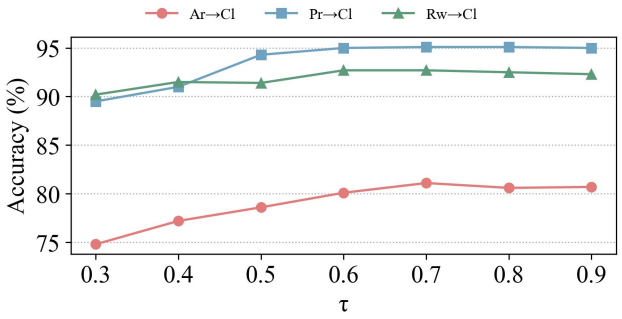


Figure 9. The effect of $\tau$ in SMKE.

For the MMR, Table 10 shows that $\tau = 0.95$ provided a slight edge in accuracy, ensuring the model effectively utilizes its predictions while still considering MLLM guidance.

In Table 11, we evaluate $\lambda$ values from 0 to 2.0. The results show that $\lambda = 0.5$ consistently yielded the highest average accuracy across multiple domain adaptation tasks, suggesting it strikes a good balance between supervised learning and regularization.

Table 7. Accuracy (%) on the **Office-Home**. SF denotes source-free, and CP, ViT denote the method uses CLIP, and ViT backbone respectively. We highlight the best result and underline the second-best one. (*) represents pre-trained CLIP/MLLM zero-shot performance. PADCLIP/ADCLIP ViT results as reported in respective paper, using ViT-B/16.

| Method | SF | CP | ViT | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source | - | ✗ | ✗ | 44.7 | 64.2 | 69.4 | 48.3 | 57.9 | 60.3 | 49.5 | 40.3 | 67.2 | 59.7 | 45.6 | 73.0 | 56.7 |
| DAPL-RN [11] | ✗ | ✓ | ✗ | 54.1 | 84.3 | 84.8 | 74.4 | 83.7 | 85.0 | 74.5 | 54.6 | 84.8 | 75.2 | 54.7 | 83.8 | 74.5 |
| PADCLIP-RN [22] | ✗ | ✓ | ✗ | 57.5 | 84.0 | 83.8 | 77.8 | 85.5 | 84.7 | 76.3 | 59.2 | 85.4 | 78.1 | 60.2 | 86.7 | 76.6 |
| PADCLIP-ViT [22] | ✗ | ✓ | ✓ | 76.4 | 90.6 | 90.8 | 86.7 | 92.3 | 92.0 | 86.0 | 74.5 | 91.5 | 86.9 | 79.1 | 93.1 | 86.7 |
| ADCLIP-RN [48] | ✗ | ✓ | ✗ | 55.4 | 85.2 | 85.6 | 76.1 | 85.8 | 86.2 | 76.7 | 56.1 | 85.4 | 76.8 | 56.1 | 85.5 | 75.9 |
| ADCLIP-ViT [48] | ✗ | ✓ | ✓ | 70.9 | 92.5 | 92.1 | 85.4 | 92.4 | 92.5 | 86.7 | 74.3 | 93.0 | 86.9 | 72.6 | 93.8 | 86.1 |
| SHOT [32] | ✓ | ✗ | ✗ | 56.7 | 77.9 | 80.6 | 68.0 | 78.0 | 79.4 | 67.9 | 54.5 | 82.3 | 74.2 | 58.6 | 84.5 | 71.9 |
| NRC [62] | ✓ | ✗ | ✗ | 57.7 | 80.3 | 82.0 | 68.1 | 79.8 | 78.6 | 65.3 | 56.4 | 83.0 | 71.0 | 58.6 | 85.6 | 72.2 |
| GKD [49] | ✓ | ✗ | ✗ | 56.5 | 78.2 | 81.8 | 68.7 | 78.9 | 79.1 | 67.6 | 54.8 | 82.6 | 74.4 | 58.5 | 84.8 | 72.2 |
| AaD [64] | ✓ | ✗ | ✗ | 59.3 | 79.3 | 82.1 | 68.9 | 79.8 | 79.5 | 67.2 | 57.4 | 83.1 | 72.1 | 58.5 | 85.4 | 72.7 |
| AdaCon [1] | ✓ | ✗ | ✗ | 47.2 | 75.1 | 75.5 | 60.7 | 73.3 | 73.2 | 60.2 | 45.2 | 76.6 | 65.6 | 48.3 | 79.1 | 65.0 |
| CoWA [25] | ✓ | ✗ | ✗ | 56.9 | 78.4 | 81.0 | 69.1 | 80.0 | 79.9 | 67.7 | 57.2 | 82.4 | 72.8 | 60.5 | 84.5 | 72.5 |
| SCLM [51] | ✓ | ✗ | ✗ | 58.2 | 80.3 | 81.5 | 69.3 | 79.0 | 80.7 | 69.0 | 56.8 | 82.7 | 74.7 | 60.6 | 85.0 | 73.0 |
| ELR [65] | ✓ | ✗ | ✗ | 58.4 | 78.7 | 81.5 | 69.2 | 79.5 | 79.3 | 66.3 | 58.0 | 82.6 | 73.4 | 59.8 | 85.1 | 72.6 |
| PLUE [34] | ✓ | ✗ | ✗ | 49.1 | 73.5 | 78.2 | 62.9 | 73.5 | 74.5 | 62.2 | 48.3 | 78.6 | 68.6 | 51.8 | 81.5 | 66.9 |
| TPDS [52] | ✓ | ✗ | ✗ | 59.3 | 80.3 | 82.1 | 70.6 | 79.4 | 80.9 | 69.8 | 56.8 | 82.1 | 74.5 | 61.2 | 85.3 | 73.5 |
| C-SFDA [19] | ✓ | ✗ | ✗ | 60.3 | 80.2 | 82.9 | 69.3 | 80.1 | 78.8 | 67.3 | 58.1 | 83.4 | 73.6 | 61.3 | 86.3 | 73.5 |
| PSAT-GDA [53] | ✓ | ✗ | ✓ | 73.1 | 88.1 | 89.2 | 82.1 | 88.8 | 88.9 | 83.0 | 72.0 | 89.6 | 83.3 | 73.7 | 91.3 | 83.6 |
| LCFD-C-RN [54] | ✓ | ✓ | ✗ | 60.1 | 85.6 | 86.2 | 77.2 | 86.0 | 86.3 | 76.6 | 61.0 | 86.5 | 77.5 | 61.4 | 86.2 | 77.6 |
| LCFD-C-B32 [54] | ✓ | ✓ | ✓ | 72.3 | 89.8 | 89.9 | 81.1 | 90.3 | 89.5 | 80.1 | 71.5 | 89.8 | 81.8 | 72.7 | 90.4 | 83.3 |
| DIFO-C-RN [55] | ✓ | ✓ | ✗ | 62.6 | 87.5 | 87.1 | 79.5 | 87.9 | 87.4 | 78.3 | 63.4 | 88.1 | 80.0 | 63.3 | 87.7 | 79.4 |
| DIFO-C-B32 [55] | ✓ | ✓ | ✓ | 70.6 | 90.6 | 88.8 | 82.5 | 90.6 | 88.8 | 80.9 | 70.1 | 88.9 | 83.4 | 70.5 | 91.2 | 83.1 |
| CLIP-RN [44]* | - | ✓ | ✗ | 51.7 | 85.0 | 83.7 | 69.3 | 85.0 | 83.7 | 69.3 | 51.7 | 83.7 | 69.3 | 51.7 | 85.0 | 72.4 |
| LLaVA-34B [36]* | - | ✓ | ✓ | 78.3 | 93.7 | 89.5 | 87.0 | 93.7 | 89.5 | 87.0 | 78.3 | 89.5 | 87.0 | 78.3 | 93.7 | 87.2 |
| InstBLIP-XXL [7]* | - | ✓ | ✓ | 82.0 | 91.6 | 88.8 | 82.2 | 91.6 | 88.8 | 82.2 | 82.0 | 88.8 | 82.2 | 82.0 | 91.6 | 86.2 |
| ShrGPT4V-13B [4]* | - | ✓ | ✓ | 66.7 | 85.8 | 84.8 | 83.2 | 85.8 | 84.8 | 83.2 | 66.7 | 84.8 | 83.2 | 66.7 | 85.8 | 80.1 |
| **RCL (Ours)** | ✓ | ✗ | ✗ | 82.5 | 95.3 | **93.3** | 89.1 | **95.3** | **92.7** | **89.3** | **82.4** | 92.8 | 89.4 | 82.1 | 95.4 | 90.0 |
| **RCL-ViT (Ours)** | ✓ | ✗ | ✓ | **83.1** | **95.7** | 93.1 | **89.2** | **95.3** | 92.6 | 89.2 | 82.3 | **92.9** | **90.0** | **83.2** | **95.5** | **90.2** |

Table 8. Accuracy (%) on the **DomainNet**.

| Method | SF | CP | ViT | C→P | C→R | C→S | P→C | P→R | P→S | R→C | R→P | R→S | S→C | S→P | S→R | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source | - | ✗ | ✗ | 42.6 | 53.7 | 51.9 | 52.9 | 66.7 | 51.6 | 49.1 | 56.8 | 43.9 | 60.9 | 48.6 | 53.2 | 52.7 |
| DAPL-RN [11] | ✗ | ✓ | ✗ | 72.4 | 87.6 | 65.9 | 72.7 | 87.6 | 65.6 | 73.2 | 72.4 | 66.2 | 73.8 | 72.9 | 87.8 | 74.8 |
| ADCLIP-RN [22] | ✗ | ✓ | ✗ | 71.7 | 88.1 | 66.0 | 73.2 | 86.9 | 65.2 | 73.6 | 73.0 | 68.4 | 72.3 | 74.2 | 89.3 | 75.2 |
| SHOT [32] | ✓ | ✗ | ✗ | 63.5 | 78.2 | 59.5 | 67.9 | 81.3 | 61.7 | 67.7 | 67.6 | 57.8 | 70.2 | 64.0 | 78.0 | 68.1 |
| NRC [62] | ✓ | ✗ | ✗ | 62.6 | 77.1 | 58.3 | 62.9 | 81.3 | 60.7 | 64.7 | 69.4 | 58.7 | 69.4 | 65.8 | 78.7 | 67.5 |
| GKD [49] | ✓ | ✗ | ✗ | 61.4 | 77.4 | 60.3 | 69.6 | 81.4 | 63.2 | 68.3 | 68.4 | 59.5 | 71.5 | 65.2 | 77.6 | 68.7 |
| AdaCon [1] | ✓ | ✗ | ✗ | 60.8 | 74.8 | 55.9 | 62.2 | 78.3 | 58.2 | 63.1 | 68.1 | 55.6 | 67.1 | 66.0 | 75.4 | 65.4 |
| CoWA [25] | ✓ | ✗ | ✗ | 64.6 | 80.6 | 60.6 | 66.2 | 79.8 | 60.8 | 69.0 | 67.2 | 60.0 | 69.0 | 65.8 | 79.9 | 68.6 |
| PLUE [34] | ✓ | ✗ | ✗ | 59.8 | 74.0 | 56.0 | 61.6 | 78.5 | 57.9 | 61.6 | 65.9 | 53.8 | 67.5 | 64.3 | 76.0 | 64.7 |
| TPDS [52] | ✓ | ✗ | ✗ | 62.9 | 77.1 | 59.8 | 65.6 | 79.0 | 61.5 | 66.4 | 67.0 | 58.2 | 68.6 | 64.3 | 75.3 | 67.1 |
| LCFD-C-RN [54] | ✓ | ✓ | ✗ | 75.4 | 88.2 | 72.0 | 75.8 | 88.3 | 72.1 | 76.1 | 75.6 | 71.2 | 77.6 | 75.9 | 88.2 | 78.0 |
| LCFD-C-B32 [54] | ✓ | ✓ | ✓ | 77.2 | 88.0 | 75.2 | 78.8 | 88.2 | 75.8 | 79.1 | 77.8 | 74.9 | 79.9 | 77.4 | 88.0 | 80.0 |
| DIFO-C-RN [55] | ✓ | ✓ | ✗ | 73.8 | 89.0 | 69.4 | 74.0 | 88.7 | 70.1 | 74.8 | 74.6 | 69.6 | 74.7 | 74.3 | 88.0 | 76.7 |
| DIFO-C-B32 [55] | ✓ | ✓ | ✓ | 76.6 | 87.2 | 74.9 | 80.0 | 87.4 | 75.6 | 80.8 | 77.3 | 75.5 | 80.5 | 76.7 | 87.3 | 80.0 |
| LLaVA-34B [36]* | - | ✓ | ✓ | 84.4 | 91.0 | 83.7 | 85.5 | 91.0 | 83.7 | 85.5 | 84.4 | 83.7 | 85.5 | 84.4 | 91.0 | 86.1 |
| InstBLIP-XXL [7]* | - | ✓ | ✓ | 82.5 | 89.0 | 83.0 | 86.7 | 89.0 | 83.0 | 86.7 | 82.5 | 83.0 | 86.7 | 82.5 | 89.0 | 85.3 |
| ShrGPT4V-13B [4]* | - | ✓ | ✓ | 79.7 | 87.9 | 79.2 | 79.9 | 87.9 | 79.2 | 79.9 | 79.7 | 79.2 | 79.9 | 79.7 | 87.9 | 81.7 |
| **RCL (Ours)** | ✓ | ✗ | ✗ | 87.6 | 92.8 | 87.9 | 89.2 | 92.7 | 87.8 | 89.6 | 87.7 | 87.6 | 89.4 | 87.5 | 92.7 | 89.4 |
| **RCL-ViT (Ours)** | ✓ | ✗ | ✓ | **88.1** | **93.3** | **88.0** | **89.7** | **93.3** | **88.0** | **89.7** | **88.0** | **87.8** | **89.7** | **88.1** | **93.3** | **89.7** |

Table 9. Accuracy (%) on the **VisDA-C**.

| Method | SF | CP | ViT | plane | bcyle | bus | car | horse | knife | mcycl | person | plant | sktbrd | train | truck | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Source | - | ✗ | ✗ | 60.4 | 22.5 | 44.8 | 73.4 | 60.6 | 3.28 | 81.3 | 22.1 | 62.2 | 24.8 | 83.7 | 4.81 | 45.3 |
| DAPL-RN [11] | ✗ | ✓ | ✗ | 97.8 | 83.1 | 88.8 | 77.9 | 97.4 | 91.5 | 94.2 | 79.7 | 88.6 | 89.3 | 92.5 | 62.0 | 86.9 |
| PADCLIP-RN [22] | ✗ | ✓ | ✗ | 96.7 | 88.8 | 87.0 | 82.8 | 97.1 | 93.0 | 91.3 | 83.0 | 95.5 | 91.8 | 91.5 | 63.0 | 88.5 |
| ADCLIP-RN [48] | ✗ | ✓ | ✗ | 98.1 | 83.6 | 91.2 | 76.6 | 98.1 | 93.4 | 96.0 | 81.4 | 86.4 | 91.5 | 92.1 | 64.2 | 87.7 |
| SHOT [32] | ✓ | ✗ | ✗ | 95.0 | 87.4 | 80.9 | 57.6 | 93.9 | 94.1 | 79.4 | 80.4 | 90.9 | 89.8 | 85.8 | 57.5 | 82.7 |
| NRC [62] | ✓ | ✗ | ✗ | 96.8 | 91.3 | 82.4 | 62.4 | 96.2 | 95.9 | 86.1 | 90.7 | 94.8 | 94.1 | 90.4 | 59.7 | 85.9 |
| GKD [49] | ✓ | ✗ | ✗ | 95.3 | 87.6 | 81.7 | 58.1 | 93.9 | 94.0 | 80.0 | 80.0 | 91.2 | 91.0 | 86.9 | 56.1 | 83.0 |
| AaD [64] | ✓ | ✗ | ✗ | 97.4 | 90.5 | 80.8 | 76.2 | 97.3 | 96.1 | 89.8 | 82.9 | 95.5 | 93.0 | 92.0 | 64.7 | 88.0 |
| AdaCon [1] | ✓ | ✗ | ✗ | 97.0 | 84.7 | 84.0 | 77.3 | 96.7 | 93.8 | 91.9 | 84.8 | 94.3 | 93.1 | 94.1 | 49.7 | 86.8 |
| CoWA [25] | ✓ | ✗ | ✗ | 96.2 | 89.7 | 83.9 | 73.8 | 96.4 | **97.4** | 89.3 | **86.8** | 94.6 | 92.1 | 88.7 | 53.8 | 86.9 |
| SCLM [51] | ✓ | ✗ | ✗ | 97.1 | 90.7 | 85.6 | 62.0 | 97.3 | 94.6 | 81.8 | 84.3 | 93.6 | 92.8 | 88.0 | 55.9 | 85.3 |
| ELR [65] | ✓ | ✗ | ✗ | 97.1 | 89.7 | 82.7 | 62.0 | 96.2 | 97.0 | 87.6 | 81.2 | 93.7 | 94.1 | 90.2 | 58.6 | 85.8 |
| PLUE [34] | ✓ | ✗ | ✗ | 94.4 | 91.7 | 89.0 | 70.5 | 96.6 | 94.9 | 92.2 | 88.8 | 92.9 | 95.3 | 91.4 | 61.6 | 88.3 |
| TPDS [52] | ✓ | ✗ | ✗ | 97.6 | 91.5 | 89.7 | 83.4 | 97.5 | 96.3 | 92.2 | 82.4 | 96.0 | 94.1 | 90.9 | 40.4 | 87.6 |
| C-SFDA [19] | ✓ | ✗ | ✗ | 97.6 | 88.8 | 86.1 | 72.2 | 97.2 | 94.4 | 92.1 | 84.7 | 93.0 | 90.7 | 93.1 | 63.5 | 87.8 |
| PSAT-GDA [53] | ✓ | ✗ | ✓ | 97.5 | 92.4 | 89.9 | 72.5 | 98.2 | 96.5 | 89.3 | 55.6 | 95.7 | **98.2** | 95.3 | 54.8 | 86.3 |
| DIFO-C-RN [55] | ✓ | ✓ | ✗ | 97.7 | 87.6 | 90.5 | 83.6 | 96.7 | 95.8 | 94.8 | 74.1 | 92.4 | 93.8 | 92.9 | 65.5 | 88.8 |
| DIFO-C-B32 [55] | ✓ | ✓ | ✓ | 97.5 | 89.0 | 90.8 | 83.5 | 97.8 | 97.3 | 93.2 | 83.5 | 95.2 | 96.8 | 93.7 | 65.9 | 90.3 |
| LLaVA-34B [36]* | - | ✓ | ✓ | 99.4 | 97.3 | 94.8 | 83.9 | 98.9 | 95.8 | 95.9 | 80.9 | 92.7 | 98.8 | 97.4 | 68.9 | 92.1 |
| InstBLIP-XXL [7]* | - | ✓ | ✓ | 99.2 | 89.6 | 82.0 | 69.8 | 97.9 | 91.0 | 97.5 | 84.3 | 73.6 | 99.3 | 96.7 | 60.0 | 86.7 |
| ShrGPT4V-13B [4]* | - | ✓ | ✓ | 99.2 | 94.7 | 90.8 | 87.9 | 98.3 | 92.1 | 97.3 | 68.0 | 96.3 | 95.6 | 96.8 | 68.2 | 90.4 |
| **RCL (Ours)** | ✓ | ✗ | ✗ | **99.5** | **96.1** | **92.6** | **89.4** | **99.1** | 97.1 | **97.0** | 85.8 | **96.6** | 98.1 | **97.3** | **70.0** | **93.2** |

Table 10. The effect of confidence threshold $\tau$ in MMR.

| $\tau$ | 1.0 | 0.95 | 0.85 | 0.75 | 0.65 |
|---|---|---|---|---|---|
| Ar→Cl | 82.3 | 82.5 | 82.2 | 82.0 | 81.9 |
| Ar→Pr | 95.4 | 95.3 | 95.3 | 95.3 | 95.3 |
| Ar→Rw | 93.1 | 93.3 | 93.3 | 93.3 | 93.4 |

Table 11. Average accuracy for different $\lambda$ values in MMR.

| $\lambda$ | →C | →P | →R | →A | Avg. |
|---|---|---|---|---|---|
| 0 | 82.0 | 95.3 | 92.8 | 89.1 | 89.8 |
| 0.5 | 82.3 | 95.3 | 92.9 | 89.3 | **90.0** |
| 1 | 82.3 | 95.3 | 92.8 | 89.2 | 89.9 |
| 1.5 | 82.2 | 95.2 | 92.8 | 89.2 | 89.9 |
| 2 | 82.3 | 95.2 | 92.8 | 89.1 | 89.9 |

### 8.2.2. Latency of deployed model

Table 12 illustrates the inference latency comparison of MLLMs and our proposed RCL. While LLaVA-34B, InstructBLIP-XXL, and ShareGPT4v-13B require latency over 1800ms per sample, RCL achieves a significantly faster inference speed of 5ms per sample. When combining all three MLLMs for ensembling, over 7000ms latency is required. This over 378x improvement in computational efficiency validates our approach of distilling MLLM knowledge into a compact model rather than relying on direct MLLM inference, enabling practical deployment while achieving even better performance. All SFDA methods using ResNet-50 have comparable inference latency. RCL adds complexity only during initial pseudo-label generation with MLLMs but maintains efficient training and inference post-deployment while achieving superior performance. In contrast, DIFO [55] incorporates additional models (e.g., CLIP-RN, CLIP-ViT) in every training iteration, increasing computational overhead throughout training.

Table 12. Latency comparison for inferencing

| **Model** | **Avg Latency (ms / sample)** |
|---|---|
| LLaVA-34B (w/STS) | ∼2850 |
| ShrGPT4v-13B (w/STS) | ∼1890 |
| InsBLIP-XXL (w/STS) | ∼2740 |
| RCL (RN50) | ∼5 |

### 8.2.3. Sensitivity to the capability of MLLMs

Table 13 provides a detailed version of Figs. 6 and 8. The upper part of the table shows the learning ability of the target model when learning from only one of the MLLMs, with the upper bound being the zero-shot performance of the individual MLLM. In contrast, RCL outperforms all the MLLMs it learns from, demonstrating that it effectively combines their knowledge and adapts it to the target domain through the curriculum learning process. The lower part of Table 13 illustrates the impact of replacing the best-performing MLLM (LLaVA-34B) with CLIP-RN50. Despite CLIP-RN50 having a 14.8% lower accuracy on the target task and a 4.9% lower average accuracy when combined with InstructBLIP-XXL and ShareGPT4V-13B, the performance

Table 13. RCL's sensitivity to MLLMs.

| MLLMs/VLMs | | | | | | Office-Home | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| LLaVA-34B | LLaVA-7B | InstBLIP-XXL | ShrGPT4V-13B | ShrGPT4V-7B | CLIP-RN | →A | →C | →P | →R | Avg. |
| ✗ | ✗ | ✗ | ✓ | ✗ | ✗ | 74.7 | 60.0 | 81.4 | 77.4 | 73.4 |
| ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | 82.2 | 81.1 | 88.4 | 88.6 | 81.1 |
| ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | 87.0 | 76.6 | 93.6 | 89.6 | 86.7 |
| ✗ | ✓ | ✗ | ✗ | ✓ | ✓ | 82.3 | 66.1 | 88.8 | 87.0 | 81.1 |
| ✗ | ✗ | ✓ | ✓ | ✗ | ✓ | 88.1 | 79.5 | 94.2 | **93.1** | 88.7 |
| ✓ | ✗ | ✓ | ✓ | ✗ | ✗ | **89.3** | **82.3** | **95.3** | 92.9 | **90.0** |

Table 14. Distribution (%) of pseudo-labeled target samples by reliability on Office-Home: $D_R$ (all teachers agree), $D_{LR}$ (2/3 agree), and $D_{UR}$ (all disagree). Reported values show the proportion of each bin within the domain, alongside the resulting RCL accuracy. Total images per domain: A: 2,427, C: 4,365, P: 4,439, R: 4,357. Ensembles are the same as Table 13.

| MLLM/VLM Teacher Ensemble | Domain | $D_R$ | $D_{LR}$ | $D_{UR}$ | **Acc** |
|---|---|---|---|---|---|
| LLaVA-7B + ShrGPT4V-7B + CLIP-RN | →A | 53.7 | 34.4 | 11.9 | 82.3 |
| | →C | 35.1 | 43.4 | 21.7 | 66.1 |
| | →P | 64.5 | 27.3 | 8.2 | 88.8 |
| | →R | 63.9 | 28 | 8.1 | 87.0 |
| InstBLIP-XXL + ShrGPT4V-13B + CLIP-RN | →A | 60.3 | 29.5 | 10.2 | 88.1 |
| | →C | 41.1 | 35.6 | 25.5 | 79.5 |
| | →P | 71.4 | 22.0 | 6.6 | 94.2 |
| | →R | 70.6 | 22.8 | 6.6 | **93.1** |
| LLaVA-34B + InstBLIP-XXL + ShrGPT4V-13B | →A | 72.3 | 21.8 | 5.9 | **89.3** |
| | →C | 58.1 | 31.1 | 11.0 | **82.3** |
| | →P | 79.3 | 18.1 | 2.6 | **95.3** |
| | →R | 76.1 | 20.0 | 3.9 | 92.9 |

of RCL only decreases by 1.3%. Furthermore, even when using MLLM/CLIP models with an average accuracy of 72.9%, RCL can still maintain an 81.1% accuracy, comparable to SOTA methods. This demonstrates that RCL is robust to the choice of MLLMs, highlighting its effectiveness in leveraging knowledge from multiple sources and adapting to the target domain.

#### 8.2.4. Impact of RCL components

In Table 15, we present the full results on Office-Home to study the impact of RCL components. Across all adaptation tasks, RKT achieves the lowest performance, utilizing only reliable data. It is important to note that even with only RKT applied, RCL already reaches similar performance levels to existing methods. We further find that applying SMKE and MMR significantly boosts RCL performance across every task. This highlights the importance of our reliability scheme and the efficient utilization of all target data.

#### 8.2.5. Knowledge transfer to a smaller backbone

Table 16 provides the complete results on Office-Home for the choice of backbone architecture. We employ RCL with a ResNet18 backbone and find that, across all adaptation tasks, RCL with ResNet18 outperforms the existing state-of-the-art methods and performs comparably to RCL with a ResNet50 backbone. Our method is adept at efficiently leveraging knowledge from multiple MLLMs and using self-correction to distill it into a smaller architecture, which is vital for deployments with resource constraints.

## 9. Multi-hot Masking Refinement Details

Multi-hot Masking Refinement (MMR) (Sec. 4.3) refines pseudo-labels for unreliable samples via Multi-hot Masking and consistency loss, ensuring robust learning from all samples. Full pseudocode is provided in Alg. 1.

**MMR Complexity and Effectiveness.** MMR demonstrates enhanced effectiveness for challenging adaptation tasks, with performance gains correlating to the proportion of unreliable samples. For example, the (→ Clipart) achieves the most significant improvement of +1.4% (80.9% to 82.3% in Table 3), despite having the highest proportion of unreliable (R=0) samples (see Fig. 10). These results indicate MMR's particular utility when dealing with large portions of unreliable labels, which aligns with real-world scenarios where MLLMs possess divergent knowledge spaces. MMR enables comprehensive data utilization through its ability to learn from all samples, rather than being limited to only high-confidence instances. Our lightweight inference model ensures deployment efficiency, achieving a practical latency

Table 15. Accuracy (%) of various components of RCL training for select adaptation tasks.

| RCL | | | Office-Home | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RKT | SMKE | MMR | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P | Avg. |
| ✓ | ✗ | ✗ | 73.5 | 89.3 | 88.2 | 82.6 | 89.1 | 87.9 | 82.7 | 73.2 | 88.2 | 83.1 | 73.2 | 89.4 | 83.3 |
| ✓ | ✗ | ✓ | 80.3 | 93.9 | 91.9 | 87.3 | 94.2 | 91.8 | 87.8 | 80.1 | 92.2 | 88.0 | 80.4 | 91.9 | 88.3 |
| ✓ | ✓ | ✗ | 81.1 | 95.1 | 92.7 | 88.5 | 95.0 | 92.3 | 88.5 | 80.8 | 92.4 | 88.6 | 80.8 | 95.3 | 89.3 |
| ✓ | ✓ | ✓ | **82.5** | **95.3** | **93.3** | **89.1** | **95.3** | **92.7** | **89.3** | **82.4** | **92.8** | **89.4** | **82.1** | **95.4** | **90.0** |

Table 16. Impact of Backbone.

| Method | BB | Office-Home | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | A→C | A→P | A→R | C→A | C→P | C→R | P→A | P→C | P→R | R→A | R→C | R→P | Avg. |
| DIFO-C-RN | RN50 | 62.6 | 87.5 | 87.1 | 79.5 | 87.9 | 87.4 | 78.3 | 63.4 | 88.1 | 80.0 | 63.3 | 87.7 | 79.4 |
| DIFO-C-ViT | RN50 | 70.6 | 90.6 | 88.8 | 82.5 | 90.6 | 88.8 | 80.9 | 70.1 | 88.9 | 83.4 | 70.5 | 91.2 | 83.1 |
| RCL (Ours) | RN18 | 81.2 | 95.3 | 92.8 | 88.9 | 94.9 | 92.4 | 88.8 | 81.7 | 92.4 | **89.5** | 81.6 | 95.1 | 89.6 |
| RCL (Ours) | RN50 | **82.5** | **95.3** | **93.3** | **89.1** | **95.3** | **92.7** | **89.3** | **82.4** | **92.8** | 89.4 | **82.1** | **95.4** | **90.0** |

of 5ms per sample. Regarding the effectiveness of MMR components, we have conducted detailed ablation studies (Tables 3, 11). These results demonstrate that both multi-hot masking and consistency loss contribute to the performance. Note that the MMR cannot be applied without Multi-hot Masking since the unreliable samples cannot be used by traditional pseudo-labeling methods.

## 10. MLLM Pseudo-labeling Details

In this section, we provide details on prompt templates (Sec. 10.1), and discussions on pseudolabel distribution (Sec. 10.3) and prompt sensitivity (Sec. 10.2).

### 10.1. Prompt Templates

To facilitate pseudo-labeling, we design specific prompt templates for different MLLMs:

**LLaVA models:** *"Question: What is the closest name from this list to describe the object in the image? Return the name only. <class names>"*

**ShareGPT4V models:** *"Question: What is the closest name from this list to describe the object in the image? List: <class names> Return the closest name from the list only. Use *exact* names from the list only. Answer:"*

**InstructBLIP models:** *"Question: What is the closest name from this list to describe the object in the image? <class names>. Use the closest name from the list only. Pick the answer from the list only. Answer:"*

For the STS calculations, we utilize all-MiniLM-L6-v2 to generate vector representations of both the outputs from the MLLMs and the text options [45].

### 10.2. Prompt Sensitivity

To evaluate robustness of our STS strategy, we tested LLaVA-34B on OfficeHome under multiple prompt templates. Beyond the default template, we considered (i) a variant with added *domain information*, (ii) a *naive template* phrased as *"What is this image from given list below {class_list}?"*, (iii) a *paraphrased class list*, and (iv) a list with injected *distractor labels*. For *paraphrased class names*, we manually curated synonyms or simplified forms of Office-Home categories (e.g., *"alarm clock" → "clock"*, *"computer monitor" → "monitor"*, *"filing cabinet" → "cabinet"*, *"push pin" → "thumbtack"*). While MLLM outputs differ since they pick from modified class lists, STS reliably maps predictions back to the original label set, showing consistency under lexical variation. For *distractor labels*, we manually injected unrelated category names sampled from DomainNet and VisDA into the candidate list, simulating noisier prompts where irrelevant options appear alongside correct classes. These variations stress-test both STS alignment and consensus-driven reliability, showing that RCL remains stable despite prompt perturbations.

Results are shown in Table 17. Across all four Office-Home transfers, performance varies by less than 2% absolute between prompt templates. While domain information slightly reduces accuracy, and paraphrases/distractors introduce minor drops, the overall averages remain stable.

Table 17. Prompt sensitivity analysis on Office-Home using STS. Zero-shot+STS accuracy with different prompt templates for LLaVA-34B show small fluctuations.

| **Prompt Template** | **Ar→Cl** | **Ar→Pr** | **Ar→Rw** | **Avg.** |
|---|---|---|---|---|
| Naive prompt | 76.80 | 92.10 | 87.45 | 85.71 |
| **Default prompt (D)** | **78.35** | **93.78** | **89.58** | **87.19** |
| D + Domain info | 77.54 | 93.08 | 88.73 | 86.25 |
| D + Paraphrased classes | 76.20 | 92.45 | 87.90 | 85.20 |
| D + Distractor labels | 76.90 | 92.70 | 88.25 | 85.79 |

**Algorithm 1** Multi-hot Masking Refinement (MMR)

**Input:** Unlabeled dataset $\mathcal{D}_t$, confidence threshold $\tau$, model $f_{\theta_t}$, MLLM outputs $\hat{y}^{mi}$
**for** each sample $x_t^i \in \mathcal{D}_t$ **do**
  Generate augmented views: $x_{t,weak}^i$ (weak) and $x_{t,strong}^i$ (strong)
  Obtain model predictions $\mathbf{z}_{t,weak}^i$ and $\mathbf{z}_{t,strong}^i$
  Compute confidence score $p_t^i = \max_c \mathbf{z}_{t,weak}^i$
  **if** $p_t^i \geq \tau$ **then**
    Assign pseudo-label $\tilde{y}^i = \text{argmax}_C \, \mathbf{z}_{t,weak}^i$
  **else**
    Compute multi-hot mask $\mathbf{m}^i$ from MLLM outputs:

$$\mathbf{m}^i = 1 - \prod_{m=1}^{M} (1 - \mathbb{1}(\hat{y}^{mi}))$$

    Refine pseudo-label:

$$\tilde{y}^i = \begin{cases} \text{argmax}_C(\mathbf{z}_t^i), & \text{if } p_t^i \geq \tau, \\ \text{argmax}_C(\mathbf{z}_t^i \odot \mathbf{m}^i), & \text{if } p_t^i < \tau, \end{cases}$$

  **end if**
  Compute supervised loss:

$$\mathcal{L}_{\text{sup}} = -\tilde{y}^i \cdot \log f_{\theta_t}(x_{t,weak}^i)$$

  Compute consistency loss:

$$\mathcal{L}_{\text{cons}} = \frac{1}{M} \sum_{m=1}^{M} \sum_{i=1}^{N_t} \text{H}(\tilde{y}^i, \mathbf{z}_{st}^i)$$

  Update model by minimizing:

$$\mathcal{L}_{\text{MMR}} = \mathcal{L}_{\text{sup}} + \lambda_{\text{cons}} \mathcal{L}_{\text{cons}}$$

**end for**

### 10.3. Pseudo-label Distributions

Fig. 10 demonstrates the distribution of sample reliability across different classes within the Office-Home dataset. We observe that for various classes across domains, there are few samples with R=1 (high reliability). This emphasizes the importance of utilizing less reliable data, a fundamental aspect of our approach, which is integrated into our SMKE and MMR techniques. To highlight our point, if we observe Table 3, it is evident that using only R=1 data (applying only RKT) results in notably lower accuracies for the Clipart domain (while adaptations to other domains achieve over 80%, Clipart is lowest at 73.3%). As Fig. 10 indicates, the

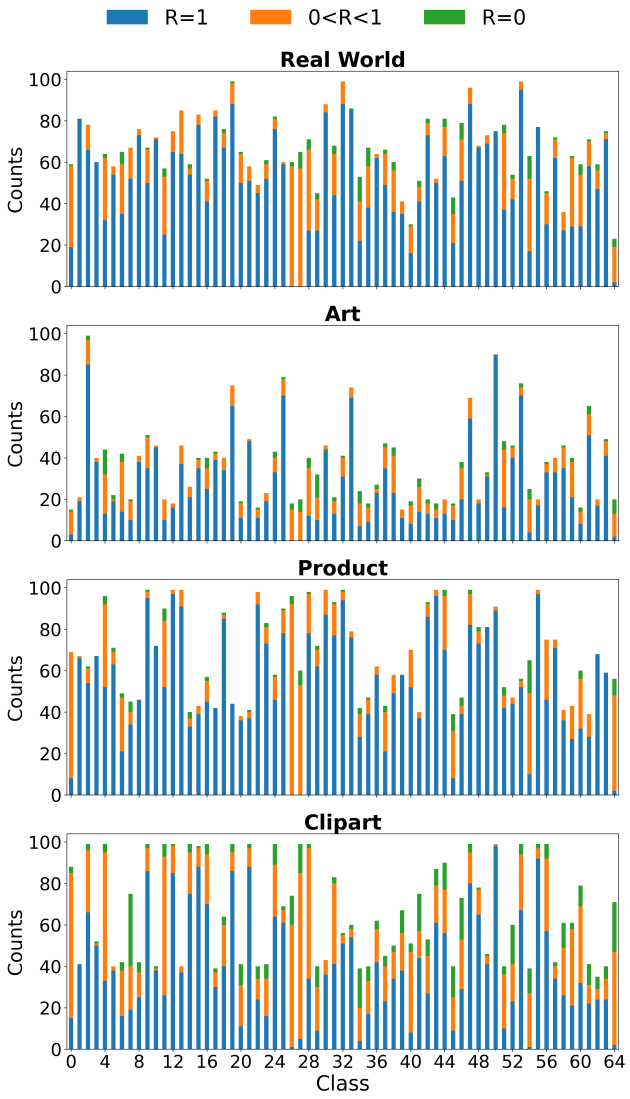


Figure 10. Per-class counts of pseudo-labels on Office-Home using LLaVA-34B, InstructBLIP-XXL, and ShareGPT4v-13B.

Clipart domain has a higher proportion of pseudo-labels where 0<R<1. By employing SMKE and MMR, we manage to enhance performance in adapting to the Clipart domain by +9% (to 82.3%). We also provide the distributions for $D_R$, $D_{LR}$, and $D_{UR}$ across various teacher ensembles and respective RCL accuracy in Table 14.

## 11. Extended Related Work on SFDA and DA

**SFDA Methods.** Early SFDA methods adapted ImageNet-pretrained models using pseudo-label refinement, such as SHOT [32], NRC [16], and G-SFDA [63]. These works emphasize clustering, entropy minimization, or distribution alignment under the no-source setting. More recent SFDA studies refine pseudo-labeling with neighborhood consistency [50, 51] or target-structure guidance [5, 33].

TPDS [23] introduced target-prototype distribution smoothing for improved calibration but still relies on strong pretrained features (e.g., CLIP). DIFO [55], the current SOTA, adapts CLIP through prompt learning and distillation, progressively aligning CLIP with target data before transferring to a task-specific model. While effective, these works typically exploit a single teacher or finetune a large VLM, making them less scalable when source data is unavailable or when deploying lightweight students is required.

**Prompt-based DA with CLIP.** Although not strictly SFDA, prompt-based DA methods highlight another direction. Chen et al. [3] proposed multi-prompt alignment for multi-source DA; Phan et al. [13] improved cross-domain transfer by aligning prompt gradients; Ge et al. [11] developed prompt learning for domain adaptation; and Vuong et al. [56] preserved clusters in prompt learning for UDA. These approaches require source data or the ability to tune CLIP prompts, unlike SFDA where such access is not allowed. We include them here to show that our reliability-driven framework complements prompt-based DA by demonstrating how frozen teachers (MLLMs or CLIP) can be exploited without modification.

**Positioning of Our Work.** Our RCL framework sits at the intersection of these lines: (1) like SFDA methods, it respects the no-source constraint, but unlike prior work, it incorporates multiple teachers; (2) like CLIP-based DA methods, it leverages large pretrained multimodal models, but without prompt tuning or finetuning; and (3) unlike DIFO or TPDS, it distills supervision into compact students by explicitly modeling teacher reliability and progressively incorporating all target data through a curriculum. This distinction is critical: RCL is not limited to a single CLIP teacher or large MLLMs, but provides a general strategy for multi-teacher SFDA under noisy pseudo-labels.

# 12. Extended Discussions

## 12.1. Why use MLLMs?

**MLLMs as Teachers.** A key motivation for adopting MLLMs is their role as foundation models that pair strong visual encoders with instruction-tuned LLMs, yielding broader semantics and instruction-following ability than contrastive VLMs like CLIP. Large-scale visual instruction tuning has produced general-purpose MLLMs with strong zero-shot performance across diverse image understanding tasks, as shown in multi-discipline benchmarks [2, 27, 39, 57, 67]. Beyond general benchmarks, domain-specific MLLMs such as LLaVA-Med [29] and biomedical/industrial evaluations like MicroVQA [38], and others [24, 47], show their practical availability where task-specific models are scarce. Importantly, recent studies show that MLLMs have now caught up with and in many cases surpassed CLIP on standard image classification benchmarks [37], highlighting their promise as future-proof teachers for SFDA. RCL provides a recipe to distill such broad supervision into compact, task-specific students without prompt tuning, teacher training, multiple inferences per sample, or source data access. In our pipeline, noisy MLLM predictions or hallucinations are naturally handled as unreliable labels, mitigated through reliability-based multi-teacher distillation.

**Robustness to Teacher Variants.** At the same time, RCL is not contingent on large MLLMs. Our framework is fundamentally teacher-agnostic, focusing on multi-teacher curriculum learning under noisy pseudo-labels. Stress tests confirm this: replacing an MLLM with CLIP-RN50 reduces Office-Home accuracy by only 1.3% (Supp. Table 10), while ensembles of weaker or smaller teachers (72.9% average zero-shot accuracy) still achieve 81.1% with RCL, surpassing strong CLIP-only SFDA baselines. Moreover, with modest student backbones (RN18/50), RCL outperforms ViT-based CLIP methods (Table 4, main), showing that the gains stem from reliability-driven distillation rather than teacher scale. Thus, MLLMs serve as a strong upper-bound case that demonstrates the potential of foundation models for SFDA, while RCL remains robust and effective with CLIP or heterogeneous teacher ensembles. We summarize key differences in using MLLM vs. CLIP/VLM models as techers in Table 18.

## 12.2. Practicality of RCL.

Although MLLMs provide strong zero-shot classification, their direct use for classification is computationally expensive and unsuitable for large-scale or real-time deployment. RCL addresses this by requiring only a single inference pass of MLLMs (or other teachers) over the target dataset. These pseudo-labels are cached and used throughout training, meaning the heavy models need not be invoked again. In contrast, CLIP-based prompt learning often requires repeated forward passes per image across multiple prompts, and direct MLLM inference is prohibitive for deployment. By distilling cached pseudo-labels into a lightweight target model, RCL produces compact students that retain the benefits of foundation knowledge while being orders of magnitude cheaper to train and deploy. This design enables real-world SFDA usage: practitioners can harness the strengths of foundation models without maintaining them in production, instead deploying efficient students adapted with RCL.

## 12.3. RCL as a General Multi-Teacher SFDA Framework

While we highlight MLLMs in the main paper, RCL can be seen as a general multi-teacher curriculum distillation framework for SFDA. The core components (RKT, SMKE, and MMR) are teacher-agnostic, working on pseudo-label reliability rather than specific architectures. Our proposed STS is what makes MLLMs viable teachers by converting free-form text outputs into closed-set predictions; for models

Table 18. Motivation and comparisons for MLLM vs. CLIP/VLM as teacher models.

| **Category** | **MLLM Teachers** | **CLIP / VLM Teachers** |
|---|---|---|
| Efficacy | Richer semantics, reasoning, instruction-following. Zero-shot MLLMs w/ STS already surpass CLIP baselines (Tab. 1,2). RCL w/ MLLMs improves +6.4% over CLIP-only RCL (Tab. 4). | Strong contrastive features, but limited to alignment. Require prompt tuning / finetuning for strong SFDA (e.g., DIFO). RCL with CLIP substitution still competitive (Supp. Tab. 10). |
| Computation | High per-query latency (1800–2800ms, Supp. Tab. 9). In RCL, used *once* for caching pseudo-labels; no repeated inference. | Lower latency per query, but prompt-learning methods require repeated inference per epoch, increasing total cost. |
| Deployment | Not suitable for on-device deployment due to size. Distillation via RCL yields compact students ($\sim$5ms/sample). | More deployable, but adaptation pipelines (e.g., DIFO) still depend on CLIP during training, raising training-time overhead. |
| Robustness | Aggregating diverse MLLMs mitigates hallucination; reliability scoring filters noise. Gains persist even with weaker MLLMs (Supp. Fig. 6, Tab. 10). | Ensembles can cover variance, but raw ensembling underperforms RCL distillation (Supp. Fig. 8). |
| Generalizability | Encodes broad, multi-domain knowledge (general + domain-specific MLLMs). Strong upper bound for SFDA and beyond. | Strong on vision tasks but limited cross-domain reasoning. Less effective in medical/industrial applications where MLLM knowledge is valuable. |

like CLIP or ViTs, this conversion is not required and the same RCL can be applied.

We validate this generality through multiple experiments:

1. CLIP substitution in Table 13. Replacing an MLLM with CLIP-RN50 reduces accuracy by only 1.3%.
2. Weaker/smaller teachers in Table 13. Even ensembles with an average of 72.9% zero-shot accuracy achieve 81.1% with RCL, surpassing strong CLIP-based SFDA methods.
3. Comparison to ViT-based baselines (TPDS [52], LCFD [54], DIFO [55]) in Table 4 (main paper). RCL-trained ResNet student outperforms ViT-based CLIP prompt-learning methods, highlighting that the framework (not teacher scale) is the main driver.

Thus, while MLLMs provide richer semantics and broad applicability (e.g., medical, industrial domains where task-specific teachers are unavailable), RCL is not confined to MLLMs. It is a robust, scalable framework that can integrate diverse teacher families, making it broadly useful for SFDA across settings. Future work can assess its viability and adoption across different tasks beyond SFDA.