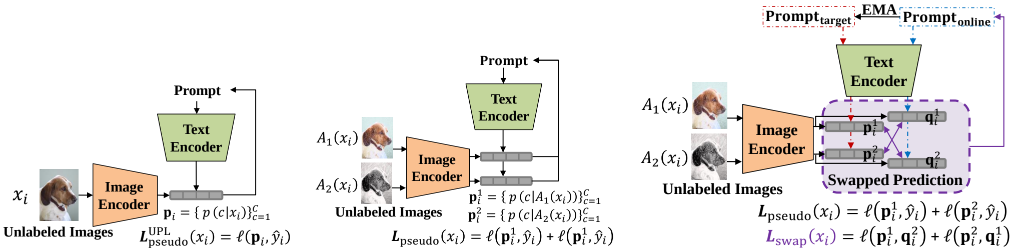
A Analysis on Only Use One Online Prompt

In section 4.3 ‘Analysis on the Decay Rate of Target Prompt.’ of the paper, we mentioned that a gradient collapse will happen if $\boldsymbol{L}_{\texttt{swap}}$ is only used on an online prompt, thus it should be assisted by a target prompt which is consistent with the online prompt. We will discuss this case in detail here.

When there is only one online prompt, the text features generated by the online prompt interact with two image features respectively to get the predictions, which are then used to compute $\boldsymbol{L}_{\texttt{swap}}$ (i.e., $\boldsymbol{L}_{\texttt{swap}}(x_i) = \ell(\mathbf{p}_i^1, \mathbf{p}_i^2) + \ell(\mathbf{p}_i^2, \mathbf{p}_i^1)$), and at this time we want to compute two gradient and optimize the online prompt, not the image encoder which generates the image features (unlike SwAV [1], which optimizes the image encoder); the input of the image encoder is two different augmented image view, while the online prompt is only one. We can make the image encoder output two image features closer to each other for two different view, but we cannot change only one online prompt to make two predictions generated by it close to each other at the same time.

Thus the reason for the gradient collapse is that when we make two predictions close to each other, the online prompt can not calculate the gradient and apply to two fixed image features at the same time. Therefore, a simple method is letting the online prompt compute the gradient only with one image feature, and stop gradient at the other image feature (i.e., set a target prompt and $\epsilon = 0$). The experiments in the paper (Table 3) show that the performance of the model in this case is worse than that where the decay rate is between 0.9 and 0.999, indicating that it is better to keep some historical information for test time adaptation.

B Additional Illustration on Objective Functions


Prompt
Text Encoder
x_i
Image Encoder
Unlabeled Images
$\mathbf{p}_i = \{p(c|x_i)\}_{c=1}^{C}$
$\boldsymbol{L}_{\text{pseudo}}^{\text{UPL}}(x_i) = \ell(\mathbf{p}_i, \hat{y}_i)$
Prompt
Text Encoder
$A_1(x_i)$
$A_2(x_i)$
Image Encoder
Unlabeled Images
$\mathbf{p}_i^1 = \{p(c|A_1(x_i))\}_{c=1}^{C}$
$\mathbf{p}_i^2 = \{p(c|A_2(x_i))\}_{c=1}^{C}$
$\boldsymbol{L}_{\text{pseudo}}(x_i) = \ell(\mathbf{p}_i^1, \hat{y}_i) + \ell(\mathbf{p}_i^1, \hat{y}_i)$
$\text{Prompt}_{\text{target}}$
EMA
$\text{Prompt}_{\text{online}}$
Text Encoder
$A_1(x_i)$
$A_2(x_i)$
Image Encoder
$\mathbf{p}_i^1$
$\mathbf{p}_i^2$
$\mathbf{q}_i^1$
$\mathbf{q}_i^2$
Swapped Prediction
Unlabeled Images
$\boldsymbol{L}_{\text{pseudo}}(x_i) = \ell(\mathbf{p}_i^1, \hat{y}_i) + \ell(\mathbf{p}_i^2, \hat{y}_i)$
$\boldsymbol{L}_{\text{swap}}(x_i) = \ell(\mathbf{p}_i^1, \mathbf{q}_i^2) + \ell(\mathbf{p}_i^2, \mathbf{q}_i^1)$

(a) UPL (b) UPL+AUG (c) SwapPrompt (Ours)

Figure 1: **Illustration of different methods.** (a) UPL only uses one image and its pseudo label to optimize one prompt. (b) UPL+AUG adds an augmented image view but still only use pseudo label loss. (c) Ours SwapPrompt leverages both self-supervised contrastive learning (i.e., $\boldsymbol{L}_{\texttt{swap}}$) and pseudo labels (i.e., $\boldsymbol{L}_{\texttt{pseudo}}$).

In section 4.3 ‘Analysis on the Decay Rate of Target Prompt.’ of the paper, we introduced the objective functions of UPL, UPL+AUG and proposed SwapPrompt. We compare those three methods by framework in this section. As shown in Figure 1, our SwapPompt not only takes advantage of the augmented view of image but also maintains two prompts to construct a self-supervised contrastive learning framework. Our framework can leverage the representation capabilities of pre-trained models to optimize the online prompt.

C Additional Experimental Results on ViT-B/16

In this section, We compare SwapPrompt with state-of-the-art baselines on ViT-B/16 visual encoder on 5 datasets. The implementation details are the same as the experiment in section 4.2 except the image backbone model. Table 1 show the empirical results of SwapPrompt along with other baselines. It is obvious that SwapPrompt still provides superior test-time adaptation performance than baselines on ViT-B/16, which verifies the advantages of our self-supervised contrastive learning framework.

Table 1: Results on ViT-B/16.

Method	Caltech101	DTD	Flowers102	Oxford-Pets	UCF101	Average
CLIP [2]	92.86	44.44	71.34	89.13	66.69	72.89
UPL [3]	93.01	47.53	73.10	93.13	73.25	75.80
TPT [4]	93.42	47.51	72.70	88.95	69.46	74.41
SwapPrompt	**93.79**	**48.46**	**74.23**	**92.58**	**74.94**	**76.80**

D Analysis on Over-Confidence Risk of Predictions

Entropy minimization is a promising method in test-time adaptation. Recently, Shu et al. [4] propose test-time prompt tuning (TPT) to extend the old entropy minimization method to vision-language model. Nevertheless, it may lead to a over-confidence risk in the model's predictions (i.e., generating high confidence for a wrong result) from directly minimizing the entropy to tuning instance-specific prompts. Reliable prediction confidence is important because it provides a measure to help gauge how much we should trust the adapted model.

In this section, we use the same setting as the experiment in section 4.2. We calculate the confidence of the final classification of all test data prediction by softmax (with a temperature coefficient of 1), and classify them into two categories according to whether they are consistent with the ground truth. Then we obtain the average prediction confidence of correct results and wrong results, respectively. Table 2 demonstrates the average prediction confidence of TPT and our SwapPrompt. It is observed that TPT has excessive confidence regardless of the correct results or the wrong results. For the correct results, TPT's prediction confidence far exceeds its accuracy. For the wrong results, they are too close to the correct results' confidence. On the other side, the confidence of correct results in SwapPrompt is much closer to its accuracy and the wrong results have a much lower confidence.

Table 2: Average prediction confidence of TPT and SwapPrompt on 5 datasets. 'Correct' denotes the average confidence of correct classification result, while 'Wrong' denotes the average confidence of wrong classification result.

	Caltech101		DTD		Flower102		Oxford-Pets		UCF101	
Method	Correct	Wrong	Correct	Wrong	Correct	Wrong	Correct	Wrong	Correct	Wrong
TPT	0.9492	0.8276	0.9058	0.7559	0.9492	0.6890	0.6621	0.7666	0.9058	0.7173
SwapPrompt	0.8452	0.4946	0.6968	0.4521	0.7480	0.4395	0.7954	0.4890	0.7422	0.3901
TPT Acc.	87.22		42.17		65.42		84.60		61.18	
SwapPrompt Acc.	89.90		47.34		70.22		89.14		65.66	

References

[1] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.

[2] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[3] Tony Huang, Jack Chu, and Fangyun Wei. Unsupervised prompt learning for vision-language models. *arXiv preprint arXiv:2204.03649*, 2022.

[4] Manli Shu, Weili Nie, De-An Huang, Zhiding Yu, Tom Goldstein, Anima Anandkumar, and Chaowei Xiao. Test-time prompt tuning for zero-shot generalization in vision-language models. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.