We thank the reviewers for their positive and constructive feedbacks of this work. Now, a detailed broader impact analysis is already included in our current draft. Then, we address the comments as follows.

1 Common comments

R1(Additional): Why K should follows the Eq. (6)? How different K impact the performance? **R2(Q2)**: Is Eq. (6) based on experiments or theory? Is it robust for different K? **R3(Q1)**: a further explanation for K would help the reader. **R**: In our framework, the samples with top K confidence in each class will be added as template samples in the next epoch. So a large K would make the class center too dependent on the additional data. It will create a negative effect when the K samples contain some wrong predictions. Eq. (6) defines the K based on our experiment. That is, the additional samples are controlled at two times of the labeled data. In the range of K that we set up, the proposed method is robust. Besides, we will further elaborate on this mechanism in the revision according to the reviewers' comments.

2 Response for R1: Thanks for the helpful comments.

Q1: How this work differs from related works? **R**: The differences are: 1) Design T-MI loss to achieve semantic labels clustering for unlabeled data compared to other SSL frameworks. 2) Generate proxy labels from the feature space while others rely on the prediction scores. We will clarify the differences more clearly in the revision.

Q2: Why T-MI performs more stable and what characteristics make DTM outperforms FixMatch should be discussed. **R**: T-MI loss performs more stable because the weak data augmentation is introduced to constitute three pairs of MI loss. So it is more robust than single pair MI loss. DTM outperforms FixMatch because the feature matching can capture more hard samples than confidence based methods. We will add more analyses for the two comparison experiments.

3 Response for R2: Thanks for the careful consideration.

Q1: Do the two CNNs share the same parameters in Fig. 2? **R**: Yes, we will note it in the revision.

Q2: How to update the memory bank? **R**: Memory bank is a matrix of $C \times K$ that stores the image ID and confidence. For a unlabeled sample, we query the corresponding row according to the predicted category. If the query sample has higher confidence than the lowest sample in the existing confidence values, then the query sample will replace it.

Q4: What the difference and relation between the proposed method and contrastive learning [1]? **R**:1) **Difference**: [1] learns a classifier for each sample. Unsupervised training is achieved by identifying instances with contrastive loss, while ours is achieved by maximizing T-MI loss of the sample and its two augmentation images. They are two different methods. 2) **Relation**: we both use a queue to store past samples, but our method only saves the features of the labeled and the $C \times K$ additional samples. We will also include this discussion in the revision.

4 Response for R3: Thanks for the positive comments.

Q1,3,4: How to confirm K when class number is not 10 or 100? The tiny question in Fig. 2, the citation of MNIST. **R**:According to our experiment, Eq. (6) is applicable when the number of categories is not 10 or 100. We will add an explanation for fc in the revised version. MNIST is now cited in the current draft.

5 Response for R4: Thanks for the carefully checking.

Q1: It seems trivial to extend the Triplet Mutual Information and its code [2]. Please explain the difference in TMI. **Q4**: Please explain the difference of Deformable Template Matching (DTM) between your work and [3, 4] separately.

R: We clarify that our work is not to extend the [2] and its code. We are sorry that the T-MI and DTM are named the same as the previous works, so it confuses the reviewer. In fact, they are different in tasks, structures, and implementations, and the codes are developed by ourselves. **For T-MI**, the differences between ours and [2] are: 1) TMI in [2] measures the MI between a sample and its positive and negative samples. Our T-MI loss evaluates the MI among the original image and its weakly and strongly augmentation images. 2) In [2], the feature map is used to calculate the MI while we use the prediction scores in the output layer to calculate the MI. 3) MI in [2] is obtained indirectly through discriminator while we use a simple and effective calculation method based on statistics. **For DTM,** 1) DTM in [3] is achieved by a set of pre-defined basic rules. 2) In [4], the new template is generated by updating the template descriptor and adding new keypoints with the matching process in pixels. 3) Our DTM refers to updating the class centers in the training phase. They are completely different implementations.

Q2: Are Table 3's parameters the same for different tasks? Is it possible to do a 2D search for hyper-parameters? **R**: Yes, we select a best parameter group ($\alpha = 0.1, \tau = 0.85$) according to the ablation experiments in Table 3. Then, the parameter group is fixed on other tasks. The reason we fix one parameter and search another is that our computation resource is limited. We will add a figure to show the results of the 2D search in the revised version.

Q3: For the comparison, how were the parameters of other methods tuned?
R: The results of the other methods shown in the paper are under the best parameters in the original papers. They select the best hyper-parameters by a series of comparative experiments. We follow the same selection strategy.

Reference

[1] He, Kaiming, et al. "Momentum contrast for unsupervised visual representation learning," in CVPR, 2020.
[2]Wu, Jianlong, et al. "Deep comprehensive correlation mining for image clustering," ICCV, 2019.
[3] Lee, Hyungtae, et al. "DTM: Deformable template matching," ICASSP, 2016.
[4] Xu, Yuhao, et al. "Partial descriptor update and isolated point avoidance based template update for high frame rate and ultra-low delay deformation matching," ICPR, 2018.