We appreciate the time invested by the reviewers in carefully reading our paper and providing very helpful and detailed comments. We sincerely thank the reviewers for their positive feedback on: paper writing [R3, R4, R5], extensive empirical evaluation [R1, R3, R4, R5], novelty of our idea [R1, R3, R4, R5], and its simplicity [R5].

Dependence of our approach on the first task [R1, R4]: Our method does not require choosing "good" classes for the first task or choosing a "good" first task. As mentioned in the paper (Line# 229), we run experiments for 5 randomly chosen task orders (with randomly chosen classes per task) and report the average accuracy. Therefore, our results are an average of different task orders with different first tasks.

Table 1: Experimental results on SVHN dataset with ResNet-18. We report the average accuracy of 5 tasks (A_5). Calibration modules are trained for each task.

Base module setting	Base module finetuned on	SVHN (A_5)
Trained from scratch	Only on the first SVHN task	98.2%
Pre-trained on CIFAR-10	No task (Frozen for all)	97.9%
Pre-trained on CIFAR-10	Only on the first SVHN task	98.4%

Our method is based on re-calibrating the base module to learn new tasks instead of training the full network on new tasks. Re-calibration needs a trained model of some sort. Therefore, a randomly initialized (untrained) base module (_SF) cannot be expected to work well for any task by using re-calibration. The base module does not have to be trained on the first task and can even be pre-trained on another dataset. We have shown in Table 1 above that a base module pre-trained on CIFAR-10 can be used to successfully perform incremental learning (A_5: 97.9%) on SVHN using our method by only training the calibration parameters for each SVHN task. In fact, if we also train the CIFAR-10 pre-trained base module on a randomly chosen first SVHN task, then the incremental performance is even higher (A_5: 98.4%). Please note that CIFAR-10 is significantly different from SVHN. *Please also note that in our paper, we have not used a pre-trained base-module in any experiment.*

If the data setting is such that classes have few samples, then our method can be modified to use the pre-trained base module setting described above. However, this is a different problem setting in itself. We will include all the above-discussed points in the revised version. Our response to other comments from reviewers is provided below.

Reviewer 1 (R1)

Comparison with other approaches: Our approach uses task labels and follows the continual learning with task labels setting. We compared our approach with methods (APD [8] ICLR-20, HNET [6] ICLR-20, GEM [36], LwF [15], and several others) that use same task labels setting. For the sake of completeness, we *additionally* compared our approach with various other continual learning setting approaches. In particular, we also compared our approach with replay/memory-based (with and without task labels) approaches such as iCaRL [16], and others. These methods store old task data (additional information) that are used during training for new tasks so that catastrophic forgetting is reduced for older tasks. Please note that our method does not store any data from the previous tasks and, therefore, does not need any dedicated memory to store task exemplars. We will mention all these details for all the compared methods in the revised version. The numbered citations refer to the references of the main paper.

Upper-bound: Our results are very close to the upper-bound (learning on all the tasks jointly with task labels setting). The upper-bounds on ImageNet-100/10 and MS-Celeb-10K/10 experiments are 98.8% and 98.4% (final accuracy), respectively, which are very close to our reported results for these experiments. We will include the upper-bounds for all the experiments in the revised version.

Additional suggestions: We will incorporate all your suggestions in the revised version.

Reviewer 3 (R3)

Piggyback: We will add more discussion on the Piggyback method (reference [5]) in the revised version.

Paper claims in line 78-79: We have justified this claim using an ablation study of setting SC (Scratch Calibration), as mentioned in Lines# 280-282. We will further highlight this point in the revised version.

Reviewer 4 (R4)

No error bar is reported: We run experiments for 5 random task orders and report the mean accuracy in the paper for all experiments. We will also mention variance in the revised version. Thanks for your suggestion.

Broader impact statement: We will improve it in the revised version. The code will be released after the publication.

Reviewer 5 (R5)

Experiments on object recognition/detection: We will include results on these tasks in the revised version.

Method setting: Thanks for pointing this out. Yes, our approach is limited to continual learning (CL) with task labels setting. We will be happy to mention it in the revised version.