R1: Thank you for recognizing this method as simple and novel (though combining existing concepts). We will provide line or bar graphs in the experiments and add more details on the specific setups in supplementary materials. **[Reverse transfer]** The issue you raised about reverse-transferring to previous models is important and interesting. Existing method (like DEN[46]) solves it by sharing and fine-tuning partial old-task weights with the new task. However, lacking the old training data causes the accuracy degradation of old tasks in subsequent learning. We plan to integrate memory-replay principle to address the reverse-transfer issue. Current replaying methods suffer from re-training which requires memory and choosing an architecture suitable for all tasks. Our method could address these issues by recording partially key data complemented to the preserved weights for fine-tuning old-task models. Finding complemented data for reserve-transfer would be an interesting topic and will be discussed as a future work of our study. **[Math. motivation]** Our method is not motivated by mathematics actually. Instead, it is more bio-motivated. Compacting the model can be seen as a 'consolidation' step in our brain; consolidation of recent memories into long-term storage occurs during REM sleep (Gais et al. 2007). Growing the model by increasing the neurons corresponds to 'neurogenesis' in the human brain. Adult neurogenesis contributes to the formation of new memories (Eriksson et al. 1998, Gage 2000). Hippocampal neurogenesis sustains human-specific cognitive function throughout life (Boldrini et al. 2018). Picking mechanism results in different cognitive functions relying on canonical neural circuits replicated across multiple brain areas (Douglas et al. 1995). Thus, compacting, picking and growing modules are effectively and sustainably combined.

R2: Thank you for leaning more towards accept. **1.1** The accuracy goals are set based on that obtained by fine-tuning the current task from an old task model. So, in Table 6, the goal of task k ($k = 2 \cdots 6$) is set as the accuracy of fine-tuning the ImageNet model (task 1) to task k. On Table 7, the goals are determined by fine-tuning the FaceRec model of task 1. We set a maximum size for growing; While not meeting the accuracy goal, the CPG process will stop when running out the maximum size. **1.2** Assume the iteration time is T, our gradual pruning trains the 10% weights-pruned model with $0.1T$ iterations, then 20% weights-pruned model with a further $0.1T$ iterations, and so on. The total iterations T is set the same as a common pruning procedure, which has not to be particularly large. **1.3** The "Finetune" column is the accuracy target. In this setting, the CPG model achieves six tasks with the only size of weights equal to one model. It is compressed to 0.6 in the first task, then expand to 0.62, 0.81, 0.8195, 0.98195, and 1.The pickup ratios from the first to fifth tasks are 0.756, 0.765, 0.822, 0.581, and 0.667. If we allow the model to double, the accuracy will increase a little. **1.4** We randomly split tasks on CIFAR and use reduced ResNet18 as in A-GEM and obtain average accuracy of 63.4 at the end of the 17-th task, better than A-GEM with 62.2, GEM with 61.2, and Prog-Net with 59.1. The memory overhead of CPG is the binary masks, which costs 2.59 MB, and that of A-GEM is the episodic memory, which costs 3.80 MB. **2.1** A distinction of our approach is the "picking" step. This has been ablated in Exp.-CIFAR, where PAE (using all the previous weights during learning) is a special case of our CPG; As seen, PAE (without picking but using all) performs worse on both model size and accuracy. Our new results also show that PAE is less favorable on larger-scale data. **2.2** We compare our method with most related works, including (1) the methods using also an iterating mechanism of compression and growing, and (2) the methods able to avoid forgetting. As for (1), to our best knowledge, DEN is a representative one among the few. There are merely few works achieving (2) too, where PackNet, Piggback, and ProgressiveNet are compared in our exps. To enhance the CIFAR experiments, we also conduct additional analysis using the CIFAR setting in A-GEM and obtain average accuracy of 70.6, 68.4, 68.6, 69.9, 67.6, 63.4, 65.2, 73.1, 62.4, 68.6, 72.8, 67.4, 65.8, 70.6, 66.6, 67.1, 63.4 at the end of each task. The average accuracy at the later task shows the forgetting measurements of CPG method. To make a fair comparison on the same architecture, we have carefully implemented DEN since its public codes only support feed-forward networks, whereas convolution and other layers in a typical CNN are missing. In our experience, DEN does not avoid unforgetting and thus a "Split&Duplication" step is enforced to lessen forgetting. However, this step is difficult to tune for balancing the current and old tasks, causing its performance degradation. **3** We would like to claim that model compression does not necessarily degrade the accuracy. E.g., in Exp. Fine-grained, the accuracy before pruning is 76.1, 83.3, 92.7, 96.7, 77.1, 80.3, and after gradual pruning is 75.8, 83.5, 92.8, 96.6, 77.1, 80.3, for the 6 tasks respectively. Thus there are only insignificant accuracy drop during pruning and sometimes the accuracy increases. Our recorded model is the compressed one for an old task and 'unforgetting' is defined in terms of the recorded models. **4, 5** Will be refined accordingly.

R3: We thank your comments. Though you fell novelty of the approach is limited, we would like to emphasize that a mix of other approaches is not necessarily un-novel. We have done critical analyses to the pros and cons of the combination, e.g., ablating our method with ProgressiveNet (without compression), PAE (omit picking), PAC (omit picking and growing), and Piggyback (with picking only); our method can achieve unforgetting, model compactness, sustainability, and high accuracy via an effective way to reuse previous knowledge; this is the first continual lifelong method attaining these goals simultaneously to our knowledge. Our work follows a common sequential task-based setup, and we plan to extend it to the case without task boundaries in the future. We choose gradual pruning because we have tried l_1-regularizations but found more iterations are needed to converge (this coincides with the survey of Cheng et al. 2018). HAT is conceptually similar to PackNet but compressing in neuron or filter level with a different mechanism; we will cite it. Our current implementation loads the entire model and the mask for each task, yet our method allows loading only the filters associated with the masked weights to speed up the individual task test.