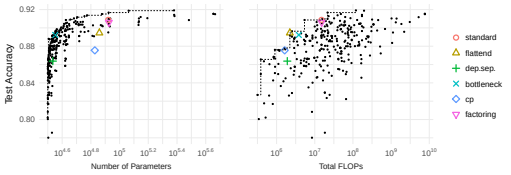
We thank all the reviewers for their dedication to reading the paper and providing helpful comments.

R1 Thank you for your positive comments. As your suggestion regarding improvements, we are planning additional experiments of a segmentation task for 3D medical data, which will be included in the camera-ready version if they make it in time.

R2 Thank you for telling us a typo and a suggestion to improve the figure. We will fix them. For your two concerns,
1) CP and Flattened were mistakenly excluded from Fig 5. We updated the results (see the right figure). Note that Flattened was not included in the 3D result (Fig 4) as we intended, because Flattened were not defined for 3D. Also, Factoring did not appear in Figs 3&4 because Factoring could not decompose 3x3 filter in terms of convolution (it was for 5x5 or more lager filters).



2) Each black dot in figures indicate either an hyperedge in the network diagram (see Eq. (6)) or some unnamed tensor decomposition found by enumeration/GA search.

R4 Thank you for your questions and comments. We will clarify all of them and will revise the paper accordingly, which will enhance the quality of the paper.

> **1-a. The propositions 1-4 are given without any explanations about its content.** Each of Propositions 1–4 was collectively explained right after Proposition 4.

> **1-b. The final result in Theorem 1 is not very informative.** We respectfully disagree with this claim. If we do not consider the redundancy in terms of representation carefully, we can generate an infinite number of equivalent networks (e.g., adding many 1x1 convolutions). To prove Theorem 1, we have to formulate the definition of redundancy, etc., a part of which have not been studied. Also, Theorem 1 has an additional value that its proof in supplementary material can be used as an algorithm of enumeration (see Line 196). We will explicitly explain this in the revised manuscript.

> **2. Enumeration of 3D convolutions having at most two inner indices.** We could not train them because its size is too huge. We could enumerate them, which are 10793 decompositions in total. Training all of them requires roughly 0.1 million GPU days, which is infeasible.

> **3-a. Standard method has best performance.** This is not true for Fig 5. The accuracy of the standard convolution was 0.91, but the most accurate one achieved nearly 0.92.

> **3-b. CP performed well. What is the main information the authors try to convey to reader.** Yes, CP performed well, especially for 3D convolution. However, CP is just one of the Pareto solutions, and we have to use other solutions when the computational resources are more limited. The main message of our results is the following. The existing tensor decompositions can be Pareto optimal, but they are very sparse; However, our method can densify them.

> **4. A lot of contents are presented for the well known introduction, e.g., CNN, tensor networks, Einconv layer.** The light-weight architecture of CNN is well-known in the CNN community. Tensor network is also well-known in the tensor/physics community. However, the intersection of them – the number of people who know both – is incredibly small, we believe. Connecting the different communities and introducing the new viewpoint for light-weight CNNs is one of our main contributions, which are also recognized by other reviewers.

Note that we have first introduced the notion of Einconv layers, so it should not have been well known.

> **5-a. How to enumerate many different tensor decompositions?** As we answered above, the proof of Theorem 1 forms an algorithm for enumeration. We will publish the real code.

> **5-b. Why the proposed method can achieve good results?** Compare to the entire space of tensor decomposition, the existing tensor decompositions such as CP represent just the tip of the iceberg. So it is reasonable to think there exist better decompositions in the iceberg, which can be found by our method.

Improvements: Comparison with other tensor network such as TT, HT etc. Thank you for the suggestion. We conducted additional experiments for TT and HT; please check the updated results (the above figure). Note that `{tt,ht}_relu` are the variants of having ReLU activation. Overall, both TT and HT are not better than CP (TT is close to CP, though).