We sincerely thank all reviewers for their valuable comments and suggestions. We will improve and update the draft according to the reviews. Below we respond to specific comments and concerns.

R1: The missing training details. We indeed are using the same training methods for fair comparisons with existing methods. Details are given in l.20 to l.26 in the supplementary material.

R1, R2 & R4: The novelty and contribution. We feel encouraged that most reviewers give positive evaluations, like 'simple and effective' (**R3**) and 'well motivated and yields good results' (**R1**). It is a novel contribution because we are the first to apply this method to deal with the problems of acquiring global context in the 3D sparse point clouds. Comprehensive experiments are conducted to evaluate the effectiveness and generalizability of our method and new state of the art has been established on multiple 3D benchmarks.

R1: Splitting up the vector & R4: Solving data sparsity. This technique has been originally introduced in (Wu et al., 2018) and is cited in the Related Works section (l.90 to l.100). In our case, by splitting up the vectors, we expand data for the "contextual encoding layer", which solves the "data-sparsity" and facilitates the learning of the global context in the sparse 3D point clouds.

R2: The group of the vectors. We have conducted a comprehensive analysis of the performance w.r.t. the group number (see l.28 to l.40 and Table 2 in the supplementary for details). In our experiment, the group follows the rule of "locality" that for feature of each individual point, represented by different colors, the vectors with neighboring channels will be grouped together. We actually tried other methods such as "channel shuffle" in ShuffleNet [25], seeking to weaken the constraint of "locality", but no significant performance improvement is gained.

R3: More results for G and K. The performance of the original Encoding layer ($G = 1$) will saturate quickly with the code words, and the performance depends entirely on the code-word number K. In our approach, We mitigate this problem by introducing the "group contextual encoding" to boost the performance by exploiting "G". The results in the following Table A show that our method ($C \times 3$, $G = 2$) can lead to the increase on accuracy without saturation when the number of code words is increased up to 32. We will add more experimental analysis in the final version.

Table A: Ablation studies of SA2$'$ layer w.r.t. G and K on Sun RGB-D V1. C is fixed to be $C \times 3$.

K	8	16	24	32
$C \times 3, G = 1$	55.8	55.5	**56.2**	55.4
$C \times 3, G = 2$	55.8	56.2	56.4	**56.7**

R3: Comparison with ImVoteNet. ImVoteNet is published very recently (after NeurIPS deadline). It is only evaluated on the benchmark of SUN-RGBD V1 while our method has been evaluated on other several benchmarks. Also, the results reported by ImVoteNet in that paper are not directly comparable to ours because it used additional RGB information while our method, only utilized the geometric information.

R3: The improvement on ScanNet. The details of the gain and variance w.r.t. the seed layer on ScanNet can be found in Table 4 of supplementary material. Actually, when measured on a more strict evaluation metrics of mAP @ 0.5, the improvement is 6.57 mAP, which can be found in Table 4, 5 in the paper. For the task of ScanNet Voxel labeling, the accuracy has been improved significantly by 1.5 (%) , surpassing the previous SOTA, PointWeb and LG-PointNet++.

R3: Global Average Pooling. We were following the EncNet [23], which uses $K = 0$ to denote the "global average pooling (GAP)". We will clarify this in the final version.

R4: PointNetVLAD. Our method is extended from the Encoding Layer [23, 24] instead of the NetVLAD [1]. The difference and the advantage of Encoding Layer [23, 24] have been clarified in [23, 24] that the encoding weight of the Encoding Layer is based on the residual. While in the NetVLAD, the weight is solely based on the input instead of the dictionary. As s result, the code words of NetVLAD are not learned from the distribution of the descriptors, making it inferior to the Encoding Layer [23, 24]. Therefore, we chose Encoding Layer as our targeting baseline. We will add a comparison with the PointNetVLAD in the final version. The concerns about the novelty and contribution regarding our method have been answered in a previous question.

R4: Comparison with SOTA on ScanNetV2 . It should be noted that the results in the paper are for 3D detection rather than segmentation, please refer to recent work of VoteNet [12][1]. In Table 6, 7 of [12] , only the results of VoteNet and 3DSIS are listed for 3D detection on ScanNetV2. These are the only two recent reported "Geo Only" results on this task and we included them all.

R4: Experiments to verify the effectiveness. To verify the effectiveness of "Grouping", we have conducted ablation experiments to study specifically the impact of "grouping". Details can be found in l.251 to l.259 with the title "Comparison with Encoding layer" and Table 2 in the paper. The results has showed that our method has outperformed the non-grouping counterparts, thus verifying the effectiveness of our "Grouping".

[1] https://arxiv.org/pdf/1904.09664.pdf