

Supplementary Material

Mateusz Michalkiewicz¹, Sarah Parisot^{2,5*}, Stavros Tsogkas^{3,4*}, Mahsa Baktashmotlagh¹, Anders Eriksson¹, and Eugene Belilovsky²

¹ University of Queensland `m.michalkiewicz@uq.net.au`,
`{m.baktashmotlagh, a.eriksson}@uq.edu.au`

² Mila, University of Montreal `eugene.belilovsky@umontreal.ca`

³ University of Toronto `tsogkas@cs.toronto.edu`

⁴ Samsung AI Research Center, Toronto

⁵ Huawei Noah's Ark Lab. London `sarah.parisot@huawei.com`

We provide additional material to supplement our work. Appendix A verifies the accuracy of our re-implementation of [1], which is, to our knowledge, the only pre-existing work on few-shot 3D reconstruction. Appendix B further examines ways of incorporating shape priors into the encoder-decoder architecture of [1]. In Appendix C, we report performance on base classes for our three considered methods and Wallace et al. [1]. Appendix D shows learned attention maps obtained using the CGCE model, analyses similarities across classes, and the choice of hyperparameters. Finally, we provide more qualitative examples in Appendix E.

A Verifying Implementation of Wallace et al. [1]

In this section we validate that our re-implementation of [1] is correct. In Table 1 we observe performance to be very similar to the numbers reported in [1], with small variations that can be reasonably attributed to random initializations. Base class performance is not reported per class in [1]. Note that in the main paper we report the results obtained using our implementation (Wallace(ours)), including the results on classes not attempted in [1].

B Addition vs Concatenation of Shape Priors.

Since concatenation is a more widely used form of conditioning in deep models than addition, we also trained a variation of [1] where the shape embedding is concatenated to the 2D encoder but noticed a drop in performance and thus decided to compare against the originally proposed architecture.

C Base Class Performance

We report in Table 2 the performance on base classes for all methods. Note that performance is similar amongst methods. This is consistent with our observation

* Stavros Tsogkas and Sarah Parisot contributed to this article in their personal capacity as an Adjunct Professor at the University of Toronto and Visiting Scholar at Mila, respectively. The views expressed (or the conclusions reached) are their own and do not necessarily represent the views of Samsung Research America, Inc and Huawei Technologies Co., Ltd.

cat	Wallace [1]	Wallace(ours)
base		
plane	N/A	0.57
car	N/A	0.84
chair	N/A	0.49
monitor	N/A	0.50
cellphone	N/A	0.74
speaker	N/A	0.66
table	N/A	0.52
mean_base	0.62	0.62
novel		
bench	0.37 (0%)	0.37 (0%)
cabinet	0.66 (0%)	0.69 (0%)
lamp	0.19 (5%)	0.20 (5%)
firearm	0.19 (58%)	0.21 (58%)
couch	0.52 (4%)	0.54 (4%)
watercraft	0.38 (15%)	0.33 (16%)
mean_novel	0.39	0.39

Table 1. Comparison of Wallace et al. [1] and our re-implementation validates our experiments. N/A means numbers were not reported in [1]. Numbers in brackets indicate percentage improvement over baseline.

that nearest neighbor can solve the problem when large number of classes is provided. Thus most learning methods with enough capacity can expect to obtain similar performance. On the other hand, as shown in the main paper, novel class performance is improved for our proposals (GCE,CGCE,MCCE) demonstrating they generalize better about shapes.

cat	Wallace	GCE	CGCE	MCCE
base				
plane	0.57	0.58	0.59	0.59
car	0.84	0.84	0.84	0.84
chair	0.49	0.51	0.49	0.50
monitor	0.50	0.52	0.51	0.52
cellphone	0.74	0.71	0.69	0.71
speaker	0.66	0.67	0.66	0.66
table	0.52	0.54	0.54	0.53
mean_base	0.62	0.62	0.62	0.62

Table 2. Results on base classes for all methods. All methods perform similarly which is consistent with our observation that for large number of classes the problem is reduced to a simple nearest neighbor search.

D Further Analysis of Compositional GCE

Attention Maps. As described in the main text, a learned attention vector α_i selects the most relevant codes from each of the 5 available codebooks. We visualize these selections for each category as a heat map in Figure 1. Note that we have previously obtained a similarity metric between classes (using nearest neighbor proximity) as shown in Figure 7 in the main text. In Table 3 we further illustrate some pairs which show high similarity. We observe in Figure 1 that *similar classes will often share codes*. We further illustrate this by selecting 3 pairs of similar categories and 3 pairs of distant categories (see Table 3 and Figure 2). Indeed this shows that CGCE model is learning to assign general structure to each class which can be reused in similar classes. As expected, however, not all codebooks for similar classes share exactly the same codes, thus they can learn distinctions across classes.

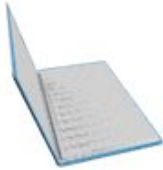








object	similar object	distant object
		
		
		

Table 3. We illustrate the categories found to be similar and dissimilar (based on our nearest neighbor proximity metric). For example, the first row shows that laptops are similar to monitors but distant from cars. Second row indicates sofas are similar to chairs but distant from phones. Third row shows watercrafts are similar to cars but distant from monitors.

Choice of Hyperparameters. Most of the common hyperparameters (e.g. conditioning size, architecture, optimization) are taken directly from [1]. For CGCE, we empirically set $M = 5$ codebooks with $c = 6$ codes each. Our preliminary experiments showed that: (a) c has to be smaller than the number of categories. Larger c values can result in each category being associated with a separate set of codes, preventing parameter sharing and accurate modelling of inter-class variability. (b) If M is very large (e.g. 1000), the information contained in each code could be degraded due to summation. Future work may consider alternative operators.

E Additional Qualitative Examples

We provide more visualizations of our reconstructions as compared to Zero-Shot baseline and Wallace which demonstrate higher quality predictions obtained using our proposed method.

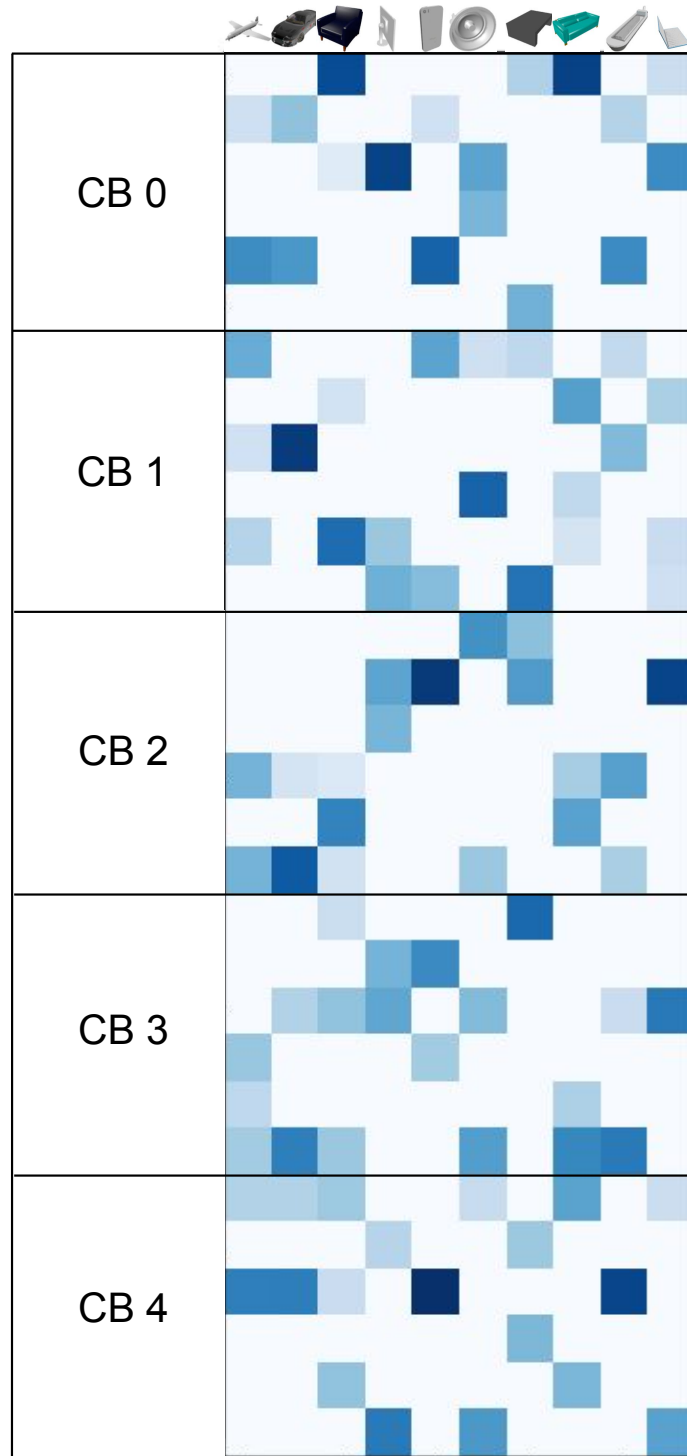


Fig. 1. Attention heat map depicting code selection for all base classes (columns 1-7) and 3 novel ones (columns 8-10). We have used 5 codebooks each one having 6 codes. Darker squares indicate higher attention given to a particular code. Note each codebook J is indicated by CB J .

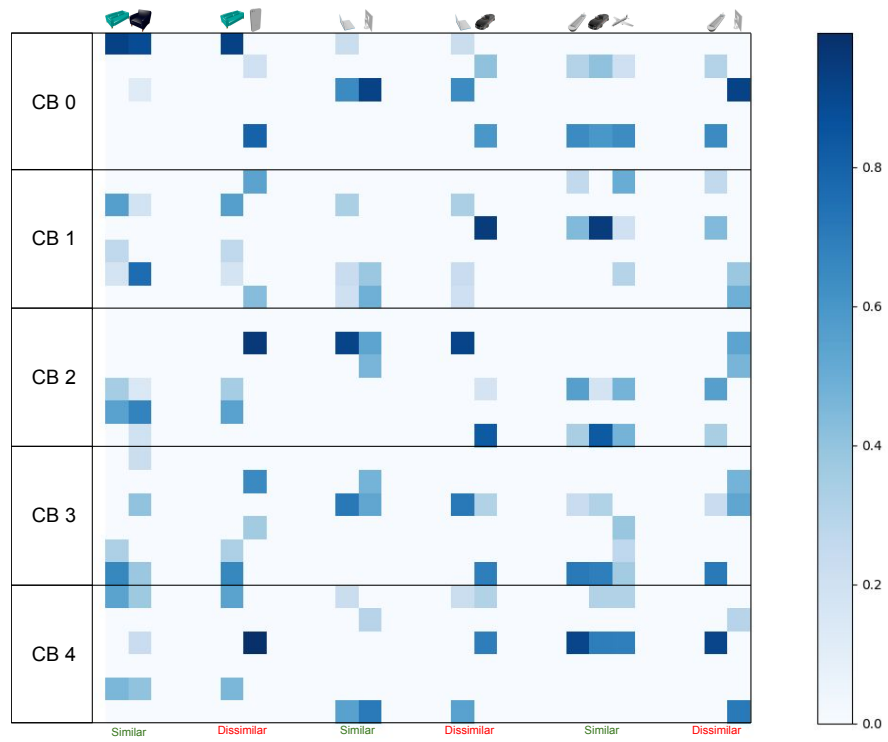


Fig. 2. Attention heat map of similar and distant categories. Columns feature 3 similar cases: (sofa and chair), (laptop and monitor), (watercraft and plane and car), and 3 distant ones: (sofa and phone), (laptop and car), and (watercraft and monitor). One can see that for similar categories, similar codes are chosen as opposed to distant ones.

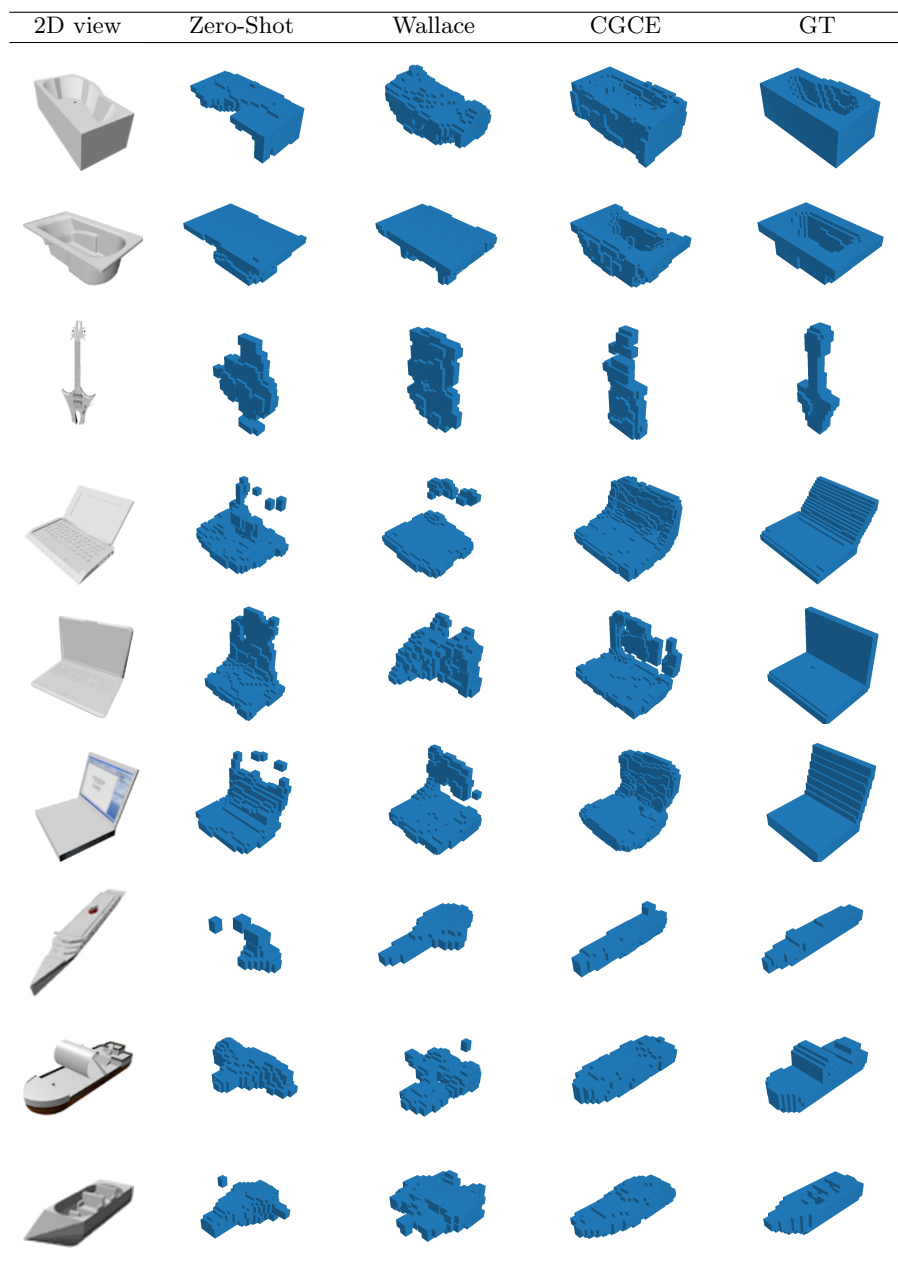


Fig. 3. Qualitative examples of shape inference obtained by Zero-Shot baseline, Wallace and proposed CGCE approach. We have used 2D views from random angles, but for visualization purposes the views are aligned to the same angle.

References

1. Wallace, B., Hariharan, B.: Few-shot generalization for single-image 3d reconstruction via priors. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3818–3827 (2019)