Escaping Plato's Cave: Towards the Alignment of 3D and Text Latent Spaces

Supplementary Material

This supplementary document provides additional technical details and experimental results to complement the main material. Specifically, we first outline some technical details pertinent to our framework in Suppl. A. Then, we evaluate in Suppl. B the alignment performance using a different text encoder, T5. Next, we extend the matching and top-5 downstream results from Table 1 in the main paper by including top-1 and top-10 retrieval scores in Suppl. C. Furthermore, in Suppl. D, we examine the impact of chosen subspace dimensions across different pairs of text and 3D encoders. These additional results provide a more comprehensive understanding of our alignment approach and its robustness across different configurations.

A. Implementation Details

In our framework, we use mean pooling for text encoders to obtain a fixed-size text representation. While we also experimented with using a class token for text encoding, it yielded consistently similar results. For 3D encoders, we extract the global output feature as the final representation.

The embedding size is set to 512 whenever possible, with the option to use projection layers when necessary. However, for multi-modal pre-trained models and certain text encoders, the embedding dimension may vary. In such cases, we adopt two distinct strategies: (1) for Canonical Correlation Analysis (CCA)-based approaches (Ours), the maximum subspace dimension is determined as the minimum of the embedding sizes of the 3D and text encoders; (2) for affine transformation-based alignment, we follow prior work by padding the lower-dimensional representation to match the higher-dimensional one.

For training parameters and dataset configurations related to uni-modal encoders, we adhere to the OpenShape settings, ensuring consistency with existing benchmarks.

B. Evaluation with an Additional Text Encoder

To further evaluate the generalization of our alignment approach, we test it using an additional text encoder, T5. Unlike BERT, RoBERTa, and CLIP's text encoder, which are encoder-only architectures, T5 follows an encoder-decoder structure. We average its encoder output embeddings and use the resulting vector as the text representation.

As shown in Tab. 3, while T5 does not achieve the same performance as CLIP's text encoder, it consistently outperforms other uni-modal text encoders, such as BERT and RoBERTa, in alignment tasks. These results suggest that the encoder-decoder structure may provide richer text representations for cross-modal alignment, although multi-modal encoders like CLIP text encoder remain superior for this task.

C. Downstream Results

We extend our evaluation by including top-1 and top-10 retrieval metrics, which complement the matching and top-5 results presented in the main paper by offering additional perspectives. As shown in Tab. 2, these results emphasize the consistency of our findings: the combination of local CKA and our proposed subspace projection method consistently achieves superior performance in retrieval tasks, whereas the affine approach demonstrates better results in matching tasks (Table 1). This highlights that method performance can vary significantly depending on the downstream task, reflecting the distinction between overall assignment accuracy (matching) and query-specific precision (top-1 retrieval). Among uni-modal 3D encoders, PointBERT performs best. Meanwhile, CLIP continues to excel as the most effective text encoder, which shows its generalizability across modalities.

The alignment approaches studied thus far exhibit limited generalization to the zero-shot classification downstream task. In particular, top-1 accuracy on Objaverse-LVIS remains below 3% even for the best-performing uni-modal 3D encoders when aligned with text encoders which is way lower to the 40% and more attained by OpenShape. This performance bottleneck can be attributed primarily to the repetitive nature of the captions used in this task: instances within the same class often share identical or nearly identical textual descriptions, leading to duplicated text embeddings. This opens up a new direction to enhance these approaches with zero-shot classification capabilities.

D. Additional Ablations

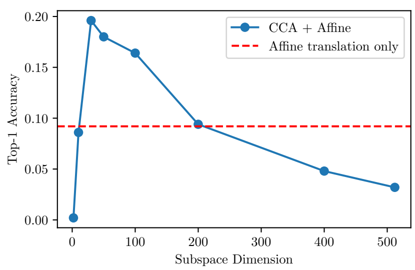
Dimensionality's impact on alignment. We generalize the dimension analysis to additional pairs of text and 3D encoders in Fig. 8, extending the findings presented in the main paper. The results confirm that our method consistently achieves better alignment in low-dimensional subspaces across all evaluated pairs, which reaffirms the importance of dimensionality reduction to enable our subspace projection approach. The optimal subspace dimension is often consistent across different encoders, but exceptions are observed. For example, MinkowskiNet exhibits improved performance at higher dimensions (e.g. 200 vs. 50), which shows that encoders have representations that might align differently. This variability highlights that the ideal subspace dimension for balancing geometric and semantic features, while being low, is not fixed but encoder-dependent.

Method	3D Encoder	CLIP		RoBERTa		BERT	
		Top-1 retrieval	Top-10 retrieval	Top-1 retrieval	Top-10 retrieval	Top-1 retrieval	Top-10 retrieval
Multi-modal 3D encoder							
Affine + Subspace Projection	OpenShape	56.4	90.8	38.8	81.8	32.4	78.8
Affine + Subspace Projection	ULIP-2	54.2	90.2	37.0	80.8	29.2	69.2
Affine + Subspace Projection	Uni3D	47.0	89.0	29.4	73.8	19.0	59.4
Uni-modal 3D encoder							
Affine	PointBert	9.8	37.2	42	22.2	3.4	22.6
Affine	SparseConv	10.6	46.2	8.0	29.4	3.2	20.4
Affine	Pointnet	7.0	30.2	3.4	22.0	3.0	20.0
Affine + Subspace Projection (Ours)	PointBert	18.0	57.4	10.8	36.6	7.6	25.8
Affine + Subspace Projection (Ours)	SparseConv	13.0	58.0	8.2	29.4	5.2	21.0
Affine + Subspace Projection (Ours)	Pointnet	14.0	44.8	7.0	35.8	6.6	23.8
Local CKA	PointBert	5.4	24.4	0.2	4.0	0.8	7.19
Local CKA	SparseConv	3.4	23.59	0.2	3.2	0.6	6.4
Local CKA	Pointnet	4.2	28.19	0.0	3.59	1.0	8.0
Local CKA + Subspace Projection (Ours)	PointBert	**30.0**	**70.8**	**17.8**	**54.4**	13.6	**51.0**
Local CKA + Subspace Projection (Ours)	SparseConv	21.2	64.0	14.79	42.4	11.4	41.4
Local CKA + Subspace Projection (Ours)	Pointnet	23.79	62.6	15.8	49.2	**14.6**	45.4

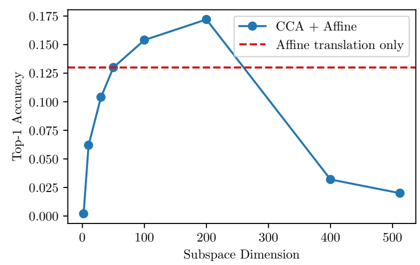
Table 2. **Top-1 and top-5 retrieval accuracy across 3D and text encoders using different alignment approaches.** We use 30,000 anchors for subspace projection and affine transformation approaches, and 1,000 anchors for local CKA. A query set of 500 is uniformly sampled, with results averaged over 3 different seeds. The subspace dimension is fixed at 50. Our approach (Ours) consistently demonstrates improved retrieval performance, with multi-modal 3D encoders setting the upper bound for performance.

Method	3D Encoder	T5	
		Matching accuracy	Top-5 retrieval
Multi-modal 3D encoder			
Affine + Subspace Projection (Ours)	OpenShape	65.0	82.6
Affine + Subspace Projection (Ours)	ULIP-2	51.8	73.2
Affine + Subspace Projection (Ours)	Uni3D	53.6	67.0
Uni-modal 3D encoder			
Affine + Subspace Projection (Ours)	PointBert	21.6	28.4
Affine + Subspace Projection (Ours)	SparseConv	22.8	23.2
Affine + Subspace Projection (Ours)	Pointnet	21.6	22.0

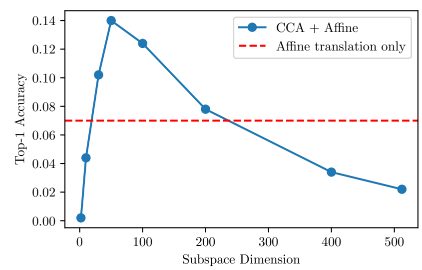
Table 3. **Matching and Top-5 retrieval accuracy using T5 text encoder and different 3D encoders.** The Affine + Subspace Projection (Ours) method is evaluated across both multi-modal and uni-modal 3D encoders. T5 is better aligned to 3D encoders compared to other uni-modal text encoders as presented in Table 1 of the main paper.



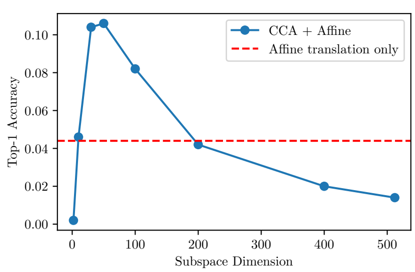
(a) CLIP and PointBert



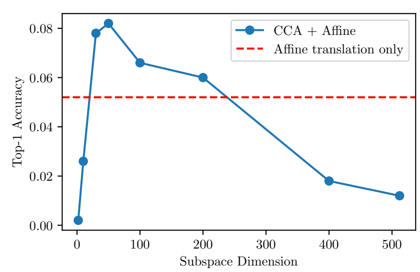
(b) CLIP and MinkowskiNet



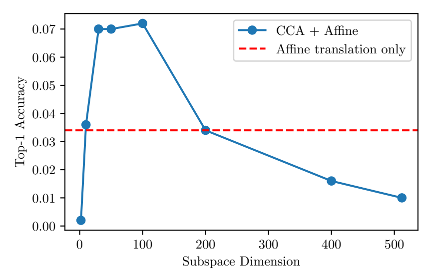
(c) CLIP and PointNet



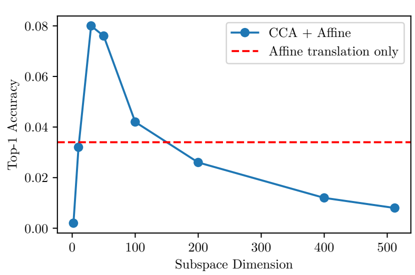
(d) RoBERTa and PointBert



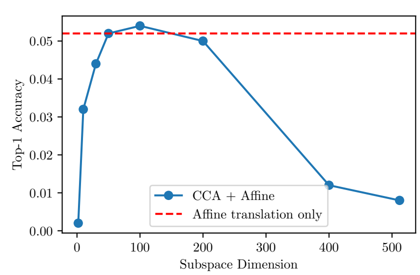
(e) RoBERTa and MinkowskiNet



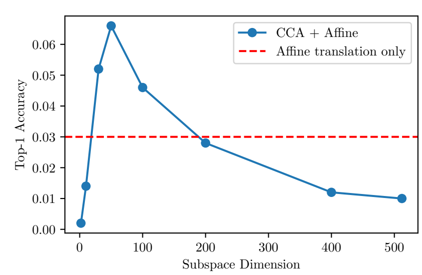
(f) RoBERTa and PointNet



(g) BERT and PointBert



(h) BERT and MinkowskiNet



(i) BERT and PointNet

Figure 8. **Impact of subspace dimensionality on retrieval performance.** Comparison of two approaches: our proposed CCA + affine translation method (blue) and affine translation without subspace projection (red). Each plot corresponds to a pair of Text Encoder and 3D Encoder. Optimal downstream performance is obtained with low-dimensional subspace projection, although the exact dimension differs from encoder to another.