MODALITY-AWARE ADAPTATION OF CONTRASTIVE LANGUAGE-IMAGE MODELS

Alexander Long, Thalaiyasingam Ajanthan, Anton van den Hengel
International Machine Learning
Amazon, Australia
{longlex,ajthal,hengelah}@amazon.com

ABSTRACT

Despite their high levels of robustness, Contrastive Language-Image Models (CLIP) still require some form of downstream adaptation when applied to tasks sufficiently out-of-domain with respect to their training set. Recent methods propose light-weight adapters on the model features, primarily focused on the few-shot domain. All such approaches however, require per-task hyperparameter tuning which necessitates access to a validation set; limiting their applicability in practice. As an alternative, we propose Modality Aware Tangent-space Retrieval (MATeR), a training-free, interpretable adapter which outperforms all recent methods when per-task hyperparameter tuning is prohibited. MATeR considers the manifold formed by CLIP embeddings when incorporating out of domain few-shot class information and its predictions are invariant to the *modality gap*; representing the first approach that considers the geometric structure of the CLIP latent space to inform downstream task adaptation. Additionally, we demonstrate a variant of MATeR has the ability to significantly increase zero-shot accuracy with only a handful of unlabelled images, much lower than the number of classes.

1 INTRODUCTION

Multi-Modal Foundation Models encode different modalities into a common vector space which can then be used in downstream tasks. Such models (Alayrac et al., 2022; Yuan et al., 2021; Li et al., 2022b; Jia et al., 2021; Radford et al., 2021) have achieved state-of-the art performance on many previously distinct Computer Vision (CV) tasks (Wang et al., 2022a; Ghiasi et al., 2022), as well as being at the core of recent image generation models (Ramesh et al., 2022; Crowson et al., 2022), however such models, and the representations they induce, remain poorly understood.

To better understand such models, we experiment with adapting the originally proposed CLIP (Radford et al., 2021), on downstream classification tasks while assuming no access to model weights. CLIP-like models are expensive to train, making the standard online learning paradigm of frequent retraining difficult and costly. In many cases, the model size makes fine-tuning out of reach for the majority of researchers, and only inference is possible for downstream use-cases. Due both the difficulty to distribute, and in an effort to recoup the cost of training such models, many organizations are moving towards making foundation models available through API calls only. In this scenario, fine-tuning is not possible as the weights of the model are not shared. Hence, there exists a strong need to facilitate precise control over such models for downstream use-cases without access to the full model weights. Due to the broad adoption (Gan et al., 2022) of CLIP in many other approaches, small, but consistent, increases in transfer accuracy have broad effects across multiple areas, and hence large practical impact. Finally, understanding the structure of the representation space, which is necessary when designing adapters with no access to model weights, provides insight that can improve the training scheme of the base models (Wang & Isola, 2020).

2 RELATED WORK

A recent work (Liang et al.) shows that there exists a modality-gap between the text and image embeddings in CLIP-like models and zero-shot performance of CLIP is usually superior to few-shot

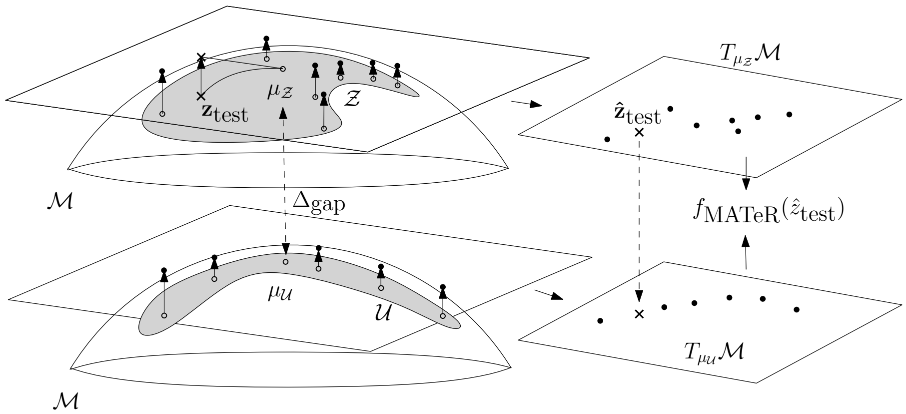

Figure 1: MATeR maps raw embeddings from the image modality $\mathcal{Z}$ and text modality $\mathcal{U}$ to the tangent space $T_x\mathcal{M}$ at the modality centers $\boldsymbol{\mu}_\mathcal{U}$, $\boldsymbol{\mu}_\mathcal{Z}$. At inference, a test embedding, $\mathbf{z}_{\text{test}}$, is mapped to the image tangent space and then copied to the text tangent space. Two k-NN like classifiers are then used as scorers, and the outputs are ensembled. The transport across the modality gap, Δ_{gap}, is critical as the magnitude of the gap is larger than the span of embeddings, and hence otherwise only the image modality influences the scorer. We visualize the modalities as two hemispheres; in reality the two distributions are present on the same hyper-sphere, with Δ_{gap} orthogonal to $\mathcal{U}$ and $\mathcal{Z}$.

(up to 4-shot) accuracy (Radford et al., 2021). To address this deficit, many adaptation approaches are developed but they rely on task-specific tuning requiring a labelled validation set per task. Popular Tip-Adapter (Zhang et al., 2021) averages zero-shot logits with a k-NN-like classifier of image encodings using task-specific mixing coefficients. Tip-X (Udandarao et al., 2022) improves over this using external information via large text-image datasets and generated images. Differently, CALIP (Guo et al., 2022) uses an attention mechanism between the token embeddings of the two-modalities and shows strong performance when the attention layers are learned. Similarly, Elevator (Li et al., 2022a) proposes improved initialization mechanisms for the projection layer to boost few-shot performance.

An alternative approach is to do prompt engineering via language models (Pratt et al., 2022) or learning (Zhou et al., 2022; Lu et al., 2022; Wang et al., 2022b). Prompt learning has shown significant improvements in multiple tasks including few-shot learning (Zhou et al., 2022) and various continual learning settings (Khattak et al., 2022). Nevertheless, these approaches require backpropagating through the full pretrained model, making them slow and limiting their applicability. In contrast, we consider a restricted (and widely applicable) setup where the pretrained models are treated as black boxes and no task-specific validation set is available.

3 Method

3.1 Problem Setup

Consider a K-shot downstream classification task with dataset $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{CK}$ with K examples from each class $c \in \mathcal{C} = \{1, 2, \ldots, C\}$. Here, $\mathbf{x}_i \in \mathcal{X}$ denotes the i-th training image and $y_i \in \mathcal{C}$ denotes the associated label. Let $\mathcal{S}, \mathcal{P} \subset \mathcal{T}$ be the set of class strings (note there may be multiple strings per class) and the prompt templates, where $\mathcal{T}$ denotes the text domain. We use the standard human-constructed templates from (Cherti et al., 2022). The text encoder $\mathbf{T} : \mathcal{T} \to \mathbb{R}^D$ is used to generate the text encodings corresponding to each combination of class strings and prompt templates. Let $\mathcal{U}_c = \{\mathbf{u}_c^{s,p} \mid s \in \mathcal{S}, p \in \mathcal{P}\}$ be the set of text encodings corresponding to class $c \in \mathcal{C}$. To create a single encoding for each class, these encodings are typically combined using some aggregation function (e.g., mean) $\mathbf{u}_c = \text{aggregate}(\mathcal{U}_c)$ where $\mathbf{u}_c \in \mathbb{R}^D$. Similarly, image encodings are produced from raw images $\mathcal{X}$ using the image encoder $\mathbf{I} : \mathcal{X} \to \mathcal{Z} \in \mathbb{R}^D$. Note that the dimension, D, of the label and image encodings is the same. In addition, these encodings are L2-normalized and hence they lie in the D-dimensional hypersphere. In the following section, we consider this geometry when designing our adaptation.

Table 1: Accuracy of methods, averaged over our 29-dataset testbench, to reduce the modality gap for CLIP-RN50 in the 1-shot case. MATeR is the only approach which *increases* accuracy over zeroshot. Rel. Rep. refers to relative representations (Moschella et al., 2022).

Method Distance	Zeroshot Cosine	L2 shifted L2	L2 shifted Cosine	Rotated Cosine	Rel. Rep. Cosine	MATeR Tangent L2
Mean	40.18	35.62	33.71	32.41	33.82	**41.89**
Median	38.84	35.66	28.12	31.51	28.61	**43.23**

3.2 Modality Aware Tangent Space Retrieval

Modality Aware Tangent Space Retrieval (MATeR) converts the normalized embeddings of each modality to their tangent-space representations, performs a fixed attention-like operation over each modality given the test image, and combines the result into a single prediction.

To motivate the use of the embedding space geometry, we first consider only the process of prompt ensembling. Typically, the aggregation function to combine individual prompt encodings for a given class is the Euclidean mean followed by L2-normalization. The motivation is to create a single point $\mathbf{u}_c$ that is representative of the true class center in the CLIP latent space. Prompt averaging in this way consistently increases zero-shot accuracy by 2-3% in comparison to simply using the class string alone (Radford et al., 2021). However, all encodings are restricted to the hypersphere due to the L2-normalization, while resultant $\mathbf{u}_c$ is not (as the averaging is performed in euclidean space). Hence, before post-normalizing, $\mathbf{u}_c$ corresponds to a point that *could not be produced* by any encoded text from $\mathcal{T}$ alone. What is desired, is the point on the hypersphere at minimum distance to all other points in the set. This is the Fréchet mean (Lee & Lee, 2012).

Let $(\mathcal{M}, \rho)$ be a Riemannian manifold equipped with an inner-product ρ on all tangent spaces $T_{\mathbf{x}}\mathcal{M}$ at $\mathbf{x}$. ρ induces a norm in each tangent space $T_{\mathbf{x}}\mathcal{M}$, which we denote as $\|\mathbf{v}\|_\rho = \sqrt{\rho_{\mathbf{x}}(\mathbf{v}, \mathbf{v})}$ for any $\mathbf{v} \in T_{\mathbf{x}}\mathcal{M}$. $d(\mathbf{x}, \mathbf{y})$ is the geodesic distance. The Fréchet mean $\boldsymbol{\mu} \in \mathcal{M}$ of a set of points $\mathcal{B} = \left\{\mathbf{x}^1, \cdots, \mathbf{x}^t\right\}$ with each $\mathbf{x}^l \in \mathcal{M}$ is defined as:

$$\boldsymbol{\mu} = \underset{\mathbf{m} \in \mathcal{M}}{\arg\min} \frac{1}{t} \sum_{l=1}^{t} d\left(\mathbf{x}^l, \mathbf{m}\right)^2 \ . \tag{1}$$

The Fréchet mean can be used as a drop-in replacement for the Euclidean prompt averaging to obtain the class centers $\mathbf{u}_c$. This alone results in a minor (0.01-0.5%) but consistent zero-shot accuracy gain across multiple datasets, model sizes, and pretraining datasets (see Table 2).

In MATeR, we construct a geometry-aware adapter when a small number image embeddings are also present. The obvious approach is to follow the zero-shot procedure and simply assign the class label to the closest training embedding (from either modality), however this fails due to the *modality gap* (Liang et al.), with the two modalities occupying distinct regions of the embedding space. Due to the gap, test images are always closest to training images and text encodings (which in the few-shot case are typically more accurate) have no effect. Naive approaches to reduce the gap which operate in euclidean space are not effective (Table 1).

The core of MATeR is the use of modality-dependent tangent-space representations. These representations have two key properties; 1) L2 distances in this space are proportional to distances over the manifold of original embeddings, 2) They are invariant to the modality gap. These representations are calculated by converting the normalized embeddings of each modality to their tangent-space representations via the logarithmic map at the Fréchet mean of that modality.

For a curve $\gamma : [a, b] \to \mathcal{M}$, we define the length of γ to be $L(\gamma) = \int_a^b \|\gamma'(t)\|_\rho \, dt$. For $\mathbf{x}, \mathbf{y} \in \mathcal{M}$, the distance $d(\mathbf{x}, \mathbf{y}) = \inf L(\gamma)$ where γ is any curve such that $\gamma(a) = \mathbf{x}, \gamma(b) = \mathbf{y}$. A geodesic $\gamma_{\mathbf{xy}}$ from $\mathbf{x}$ to $\mathbf{y}$, is a curve that minimizes this length.

For each point $\mathbf{x} \in \mathcal{M}$ and vector $\mathbf{v} \in T_{\mathbf{x}}\mathcal{M}$, there exists a unique geodesic $\gamma : [0, 1] \to \mathcal{M}$ where $\gamma(0) = \mathbf{x}, \gamma'(0) = \mathbf{v}$. The exponential map $\exp_{\mathbf{x}} : T_{\mathbf{x}}\mathcal{M} \to \mathcal{M}$ is defined as $\exp_{\mathbf{x}}(\mathbf{v}) = \gamma(1)$. The logarithmic map $\log_{\mathbf{x}} : \mathcal{M} \to T_{\mathbf{x}}\mathcal{M}$ is the inverse of $\exp_{\mathbf{x}}$. The per-modality tangent-space representations are then;

$$\hat{\mathbf{z}} = \log_{\boldsymbol{\mu}_{\mathcal{Z}}}(\mathbf{z}), \qquad \hat{\mathbf{u}} = \log_{\boldsymbol{\mu}_{\mathcal{U}}}(\mathbf{u}). \tag{2}$$

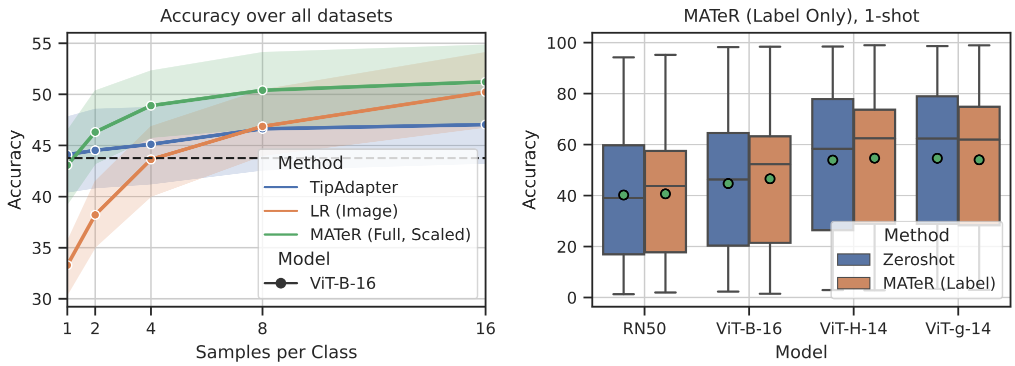

Figure 2: **Left:** Few-shot performance. Dashed line indicates mean zero-shot accuracy, LR is the Linear Regression baseline. Mean accuracy is shown over 29 datasets for various samples per class. The 50% confidence interval over datasets is shown. Note this is not a measure of error - it captures the distribution of performance of each method over the range of tasks. **Right:** Label-only MATeR vs. zero-shot across multiple architectures. Median accuracy increases 3.6% on average (see Table 4 for numeric results), an improvement comparable to the change in zero-shot accuracy between ViT-H-14 to ViT-g-14, an additional 400M parameters. Accuracy quartiles are shown over the 29 datasets in our evaluation benchmark, with means as green circles.

Here, $\hat{\mathbf{z}}$ and $\hat{\mathbf{u}}$ are relative to the respective modality centers (*i.e.*, Fréchet means). Intuitively, these representations can be viewed as modality-relative L2 coordinates of the 'flattened' embedding manifold at each modality center. See Sec. A.3 for additional background to the logarithmic map.

Standard learning algorithms can now be applied to $\hat{\mathbf{z}}$ and $\hat{\mathbf{u}}$. We use a k-NN-like classifier where scores are computed over the entire training dataset. For a test image $\mathbf{z}^{\text{test}}$ we obtain $\hat{\mathbf{z}}^{\text{test}}$ from Eq. (2), similarly all text and image embeddings are converted to the respective tangent representations. Given the tangent representations of the images, the score of the test image in the image modality for a given class is computed as the mean inverse distance to an image of that class.[1] Precisely,

$$a^z_c = \frac{1}{K}\sum_{j=1}^{K} \|\hat{\mathbf{z}}_{\text{test}} - \hat{\mathbf{z}}^j_c\|_\rho^{-1} , \quad \forall c \in \mathcal{C} , \tag{3}$$

where $\hat{\mathbf{z}}^j_c$ denotes the tangent representation of j-th image belonging to class c. Now, the scores in image and text domains can be written in vectorized form as:

$$\mathbf{a}^z = [a^z_1, \ldots, a^z_C] , \quad \mathbf{a}^u = [\|\hat{\mathbf{z}}_{\text{test}} - \hat{\mathbf{u}}_1\|_\rho, \ldots, \|\hat{\mathbf{z}}_{\text{test}} - \hat{\mathbf{u}}_C\|_\rho] . \tag{4}$$

We then combine these distances into a single scorer:

$$f_{\text{MATeR}}(\mathbf{z}_{\text{test}}) = \sigma\left(\mathbf{a}^z\right) + \sigma\left(\alpha\, \mathbf{a}^u\right) , \tag{5}$$

where σ is the softmax function and $\alpha > 0$ balances the two scores.[2] The label with maximum softmax score is chosen as the prediction. The above scoring function is invariant to the magnitude of the modality gap as the gap is orthogonal to the span of the modality embeddings (Zhang et al.). However, the relative dispersion of each modality can still negatively affect the final scorer.

4 Experiments

4.1 Few-Shot Adaptation

We evaluate downstream accuracy for 1, 2, 4, 8 and 16 samples per class for over 29 datasets. All results are averaged over 5 random seeds. We evaluate over a range of encoders from RN50 to ViT-g-14. For model details please refer to Sec. C.2. In all cases we report accuracy distributions

[1]Instead of inverse mean, other functions such as the negative min or inverse min can be used. See Fig. 5.

[2]In all our experiments $\alpha = 2$ works well – sharpening the label logits. However, as shown in the appendix (Fig. 6), $\alpha = 2$ is far from the optimal value. Additionally, without α ($\alpha = 1$) MATeR's performance remains high.

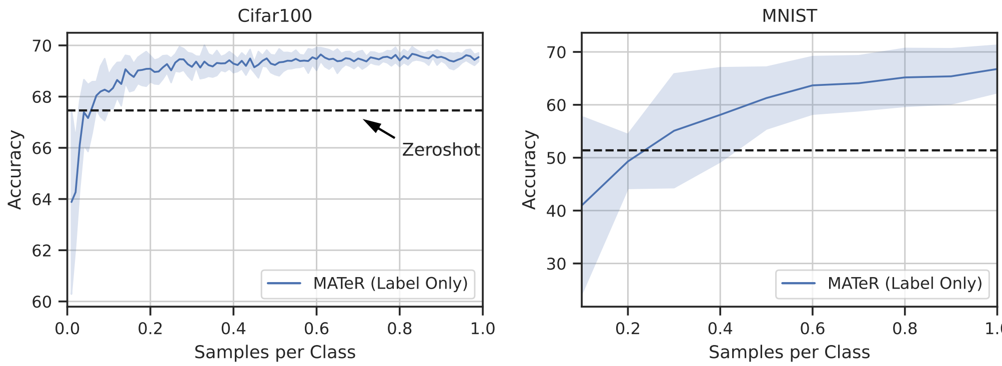

Figure 3: Unlabeled 'less-than-one-shot' MATeR accuracy with a ViT-B-16 variant on CIFAR100 and MNIST. The shaded region represents the standard deviation over 10 seeds. Significant improvement over zero-shot is achieved with only a small number of unlabeled images. In contrast, a linear probe requires several samples per class to match zero-shot performance.

over datasets rather than point estimates such as the mean, as these can significantly obscure true performance (Agarwal et al., 2021). Our primary comparison is to linear regression (we sometimes refer to this as a linear probe) on the image features which remains a strong baseline (Li et al., 2022a) and requires little tuning. As shown in Fig. 2, MATeR outperforms zero-shot, Tip-Adapter and LR, for all sample budgets.

We note an additional intriguing result; MATeR's label-logit only performance is significantly higher than standard zero-shot (see Fig. 2 right) . That is, $f_{\text{MATeR (Label only)}}(\mathbf{z}_{\text{test}}) = \sigma\left(\mathbf{a}^u\right)$; this can be thought of as performing zero-shot on the tangent representations. Images are not considered as part of this scorer, other than to find the Fréchet mean of the image manifold in order to calculate $\hat{\mathbf{z}}_{\text{test}}$. Consequently, accuracy does not improve as additional images are provided. When averaged across all datasets, this approach outperforms standard zero-shot by 5.97% in median accuracy with the ViT-B-16 backbone. On individual datasets, the improvement is as large as a 21% in absolute percentage improvement (see Table 5).

4.2 Less than 1-shot

MATeR's strong label-only accuracy leads to an interesting question; is it sufficient to use fewer than a single sample per class to identify the image manifold Fréchet mean? We carry out the following experiment to answer this question; we sub-sample the dataset to one image per class then select, without replacement, n of these C images. Tangent image representations $\hat{\mathbf{z}}_{\text{test}}$ are then computed using the Fréchet mean derived from this subset. As no image labels are provided, the scorer is only informed by the tangent label encodings. We change n within the range $\{1, 2, \ldots, C\}$ and repeat the sub-sampling process over 10 random seeds. As shown in Fig. 3, with only a handful of unlabeled images (10 for CIFAR100 and 3 for MNIST), MATeR is able to significantly improve on zero-shot accuracy.

This provides MATeR a unique property; less than 1-shot, *unlabeled* adaptation whereas all other methods require at least one labelled image per class. In practice, this means for a domain of interest simply defining the labels to be classified and collecting a small number of (unlabeled) sample images is sufficient to significantly boost zero-shot performance. Additionally, if we allow the means to be calculated from the test set, no training images are required *at all* as the set of unlabeled test images is sufficient to define the Fréchet mean of the image manifold.

5 Conclusion

We introduce MATeR, a lightweight adapter for CLIP-like models that outperforms strong baselines on a wide variety of datasets and base models. In addition to providing a state-of-the-art approach for transfer learning for such models, MATeR demonstrates the novel capability of outperforming the zero-shot accuracy with only a handful of unlabeled images, much fewer than the number of classes.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron C Courville, and Marc Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in neural information processing systems*, 34:29304–29320, 2021.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. *arXiv preprint arXiv:2212.07143*, 2022.

Katherine Crowson, Stella Biderman, Daniel Kornis, Dashiell Stander, Eric Hallahan, Louis Castricato, and Edward Raff. Vqgan-clip: Open domain image generation and editing with natural language guidance. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVII*, pp. 88–105. Springer, 2022.

Basura Fernando, Amaury Habrard, Marc Sebban, and Tinne Tuytelaars. Subspace alignment for domain adaptation. *arXiv preprint arXiv:1409.5241*, 2014.

Zhe Gan, Linjie Li, Chunyuan Li, Lijuan Wang, Zicheng Liu, Jianfeng Gao, et al. Vision-language pre-training: Basics, recent advances, and future trends. *Foundations and Trends® in Computer Graphics and Vision*, 14(3–4):163–352, 2022.

Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI*, pp. 540–557. Springer, 2022.

John C Gower and Garmt B Dijksterhuis. *Procrustes problems*, volume 30. OUP Oxford, 2004.

Ziyu Guo, Renrui Zhang, Longtian Qiu, Xianzheng Ma, Xupeng Miao, Xuming He, and Bin Cui. Calip: Zero-shot enhancement of clip with parameter-free attention. *arXiv preprint arXiv:2209.14169*, 2022.

Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. OpenCLIP, 7 2021.

Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International Conference on Machine Learning*, pp. 4904–4916. PMLR, 2021.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. *arXiv preprint arXiv:2210.03117*, 2022.

John M Lee and John M Lee. *Smooth manifolds*. Springer, 2012.

Chunyuan Li, Haotian Liu, Liunian Harold Li, Pengchuan Zhang, Jyoti Aneja, Jianwei Yang, Ping Jin, Yong Jae Lee, Houdong Hu, Zicheng Liu, et al. Elevater: A benchmark and toolkit for evaluating language-augmented visual models. *arXiv preprint arXiv:2204.08790*, 2022a.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. *arXiv preprint arXiv:2201.12086*, 2022b.

Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Zou. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. In *Advances in Neural Information Processing Systems*.

Yuning Lu, Jianzhuang Liu, Yonggang Zhang, Yajing Liu, and Xinmei Tian. Prompt distribution learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5206–5215, 2022.

Nina Miolane, Nicolas Guigui, Alice Le Brigant, Johan Mathe, Benjamin Hou, Yann Thanwerdas, Stefan Heyder, Olivier Peltre, Niklas Koep, Hadi Zaatiti, et al. Geomstats: a python package for riemannian geometry in machine learning. *Journal of Machine Learning Research*, 21(223):1–9, 2020.

Luca Moschella, Valentino Maiorca, Marco Fumero, Antonio Norelli, Francesco Locatello, and Emanuele Rodolà. Relative representations enable zero-shot latent space communication. *arXiv preprint arXiv:2209.15430*, 2022.

Sarah Pratt, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. *arXiv preprint arXiv:2209.03320*, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Vishaal Udandarao, Ankush Gupta, and Samuel Albanie. Sus-x: Training-free name-only transfer of vision-language models. *arXiv preprint arXiv:2211.16198*, 2022.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pp. 9929–9939. PMLR, 2020.

Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022a.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 139–149, 2022b.

Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.

Renrui Zhang, Rongyao Fang, Peng Gao, Wei Zhang, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free clip-adapter for better vision-language modeling. *arXiv preprint arXiv:2111.03930*, 2021.

Yuhui Zhang, Jeff Z HaoChen, Shih-Cheng Huang, Kuan-Chieh Wang, James Zou, and Serena Yeung. Drml: Diagnosing and rectifying vision models using language. In *NeurIPS ML Safety Workshop*.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, 2022.

A Additional Background

A.1 Modality Gap

The modality gap is an observed phenomenon where CLIP-like models do not perfectly align points between modalities. Instead, there exists a large, and constant shift, with the two modalities occupying distinct regions of the embedding space. Liang et al. first observed this phenomenon and demonstrated empirically that reducing the gap by shifting and re-normalizing one modality towards the other modalities class center *reduces* zero-shot accuracy. Zhang et al. show empirically that for many datasets 1) The modality gap between corresponding image and text embeddings can be approximated by a constant vector, particularly at the class level and 2) The modality gap is orthogonal to the span of image embeddings and text embeddings, and image embeddings and text embeddings have zero mean in the subspace orthogonal to the modality gap.

The modality gap creates a problem when performing downstream classification. Standard (i.e in the ambient space) 1-NN functions by assigning the class label of the closest point. Given the modality gap is much larger than the class-to-class distance, this results in label encodings (which inform the zero-shot model) to never be considered during classification, and a multi-modal, single-index k-NN reverts to an image-only k-NN. Similar effects distort linear classifiers. Ideally, closing the gap would allow a combination of text and image encodings to inform classification via a single model.

A.2 zero-shot CLIP and Tip-Adapter, Tip-X, CALIP

In the standard CLIP zero-shot setting, the class predictions are scored via cosine similarity to the test image encoding; $f_{\text{zero-shot}}(\mathbf{z}_{\text{test}}) = \mathbf{z}_{\text{test}}^T \mathbf{U}$ where $\mathbf{z}_{\text{test}} = \mathbf{I}(\mathbf{x}_{\text{test}})$, $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_C]^T$ and $\mathbf{u}_c$ is the class encoding. The prediction rule is standard, $y_{\text{test}}^{\text{pred}} = \arg\max_{c \in \mathcal{C}}(f_{\text{zero-shot}}(\mathbf{z}_{\text{test}}^T))$

Remarkably, the original CLIP paper found zero-shot performance (where no example images are provided) outperforms linear probe few-shot classification until 4 images per class are provided (averaged over multiple datasets). Several works attempt to address this via ensembling few-shot logits with zero-shot logits, however this is difficult as zero-shot logits model are poorly calibrated due to the modality gap (see sec. A.1).

Tip-Adapter (Zhang et al., 2021) is a recent method that displays monotonically increasing accuracy as images are added to the training set.

$$f_{\text{TIP-A}}(\mathbf{z}_{\text{test}}) = \alpha \exp\left(-\beta(1 - \mathbf{z}_{\text{test}} \mathbf{Z}^T\right) \mathbf{L} + \exp(\tau) \mathbf{z}_{\text{test}} \mathbf{U}^T \tag{6}$$

where $\mathbf{L} \in \mathbb{R}^{CK \times C}$ is a row-wise one-hot matrix indicating the training image class labels, $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_C]^T \in \mathbb{R}^{C \times D}$ and $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{CK}]^T \in \mathbb{R}^{CK \times D}$, and α and β are hyperparameters which modulate the few-shot score distribution. The first term is equivalent to a distance (in this case cosine similarity as $\mathbf{z}_{\text{test}}$ is normalized and $\mathbf{Z}$ is row-wise normalized) weighted k-NN with $k = C \times K$, and logits transformed by an exponential-like activation, summed over classes, and re-weighted. The second term is the temperature-scaled CLIP zero-shot 'logits'. The $\exp(\tau)$ term is clipped to 100 in the original CLIP training, and Tip-Adapter uses this value in all experiments.

The α and β terms present a problem in the few-shot case as they have a large effect on prediction accuracy and must be tuned with a validation set, which is not present. In our implementation, we use $\alpha = 0.5$ and $\beta = 1$ as global default values (see fig. 4 for impact of this alteration). An alternative would be to tune these values on a training set of tasks, and evaluate on unseen test tasks, however we leave this to future work.

TIP-X (Udandarao et al., 2022) adds an additional term that attempts to improve calibration by making the attention relative to the training image encodings affinity to the test image encodings, via a KL-divergence term.

$$f_{\text{TIP-X}}(\mathbf{z}_{\text{test}}) = f_{\text{TIP-A}}(\mathbf{z}_{\text{test}}) + \gamma \psi(-\mathbf{M}) \mathbf{L} \tag{7}$$

where $M_{ij} = D_{KL}(\sigma(\mathbf{z}_{\text{test}} \mathbf{U}^T) \mid\mid \sigma(\mathbf{Z} \mathbf{U}^T))$, σ is the softmax function, and ψ is a re-scales $\mathbf{M}$ to have magnitudes equal to the few-shot logits from Tip-Adapter.

CALIP Guo et al. (2022) is similar;

$$f_{\text{CALIP}}(\mathbf{z}_{\text{test}}) = \alpha_1 \mathbf{z}_{\text{test}} \mathbf{U}^T + \alpha_2 \mathbf{z}_{\text{test}} \sigma(\mathbf{A}/\tau_1) \mathbf{U} + \alpha_3 \mathbf{z}_{\text{test}} \sigma(\mathbf{A}^T/\tau_2) \mathbf{Z} \tag{8}$$

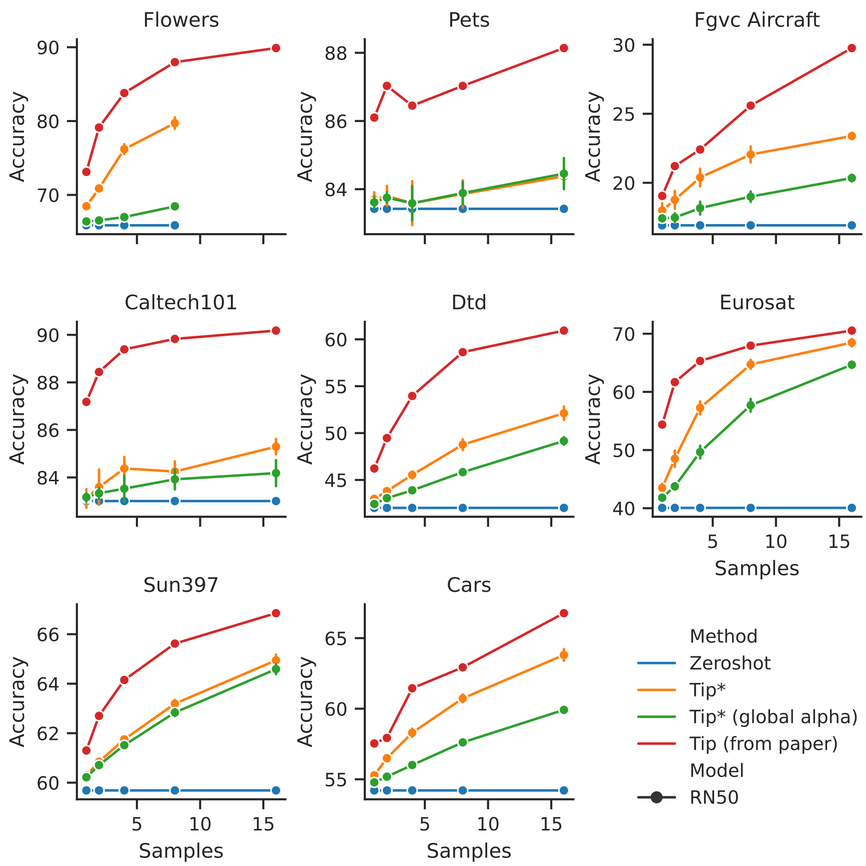

Figure 4: Tip-Adapter Replication. 'Tip*' is our re-implementation with the original per-dataset α values, Tip* (global alpha) is with $\alpha = 1$ for all datasets. Re-implementation results are the mean over 5 seeds, with standard deviation shown as error bars. We were unable to perfectly replicate Tip's performance on our version of the datasets and with our inference pipeline. This may be due to different prompt templates (Tip uses a single prompt), or the fact that Tip uses a small amount of image augmentation when constructing image features. For the flowers dataset, we do not report the 16-shot case as not all classes have 16 samples. We do see that both the reported numbers and the re-implementation improve on the zero-shot performance for any number of samples, across all datasets. In addition, it is clear that the inability to tune α has a detrimental effect on accuracy.

Table 2: Fréchet prompt averaging provides a consistent boost in Zeroshot accuracy over various models and pretraining datasets. Accuracy is averaged over all datasets with > 1 prompt. We also report the 'None' case where no averaging is applied, and multiple label encodings per class are present.

		Mean Accuracy			Median Accuracy		
Model	Pretraining	Frechet	Euclidean	None	Frechet	Euclidean	None
RN50	openai	41.35	41.31	40.46	39.54	39.45	39.00
ViT-B-16	openai	48.27	48.24	48.44	49.06	49.05	48.24
ViT-B-32	openai	45.38	45.39	45.18	46.78	46.84	46.03
ViT-H-14	laion2b_s32b_b79k	59.05	59.04	58.36	62.56	62.56	61.70
ViT-L-14	laion2b_s32b_b82k	56.98	56.95	56.44	59.60	59.41	58.85
	openai	53.90	53.84	52.85	52.94	52.97	51.97
ViT-g-14	laion2b_s12b_b42k	63.47	63.46	62.91	67.00	67.03	65.62

where $\mathbf{A} = \mathbf{Z}\mathbf{U}^T$.

All approaches also include 'fine-tuning' variants, where some parameters in the adapter are unfrozen and the adapter trained using a standard cross entropy loss.

A.3 Riemmannian Geometry Background and Notation

An n-dimensional manifold $\mathcal{M}$ is a topological space that is locally homeomorphic to $\mathbb{R}^n$. The tangent space $T_{\mathbf{x}}\mathcal{M}$ at $\mathbf{x}$ is defined as the vector space of all tangent vectors at $\mathbf{x}$. For a manifold $\mathcal{M}$, a Riemannian metric $\rho = (\rho_{\mathbf{x}})_{\mathbf{x}\in\mathcal{M}}$ is a smooth collection of inner products $\rho_{\mathbf{x}} : T_{\mathbf{x}}\mathcal{M} \times T_{\mathbf{x}}\mathcal{M} \to \mathbb{R}$ on the tangent space of every $\mathbf{x} \in \mathcal{M}$. The resulting pair $(\mathcal{M}, \rho)$ is called a Riemannian manifold. Note that ρ induces a norm in each tangent space $T_{\mathbf{x}}\mathcal{M}$, given by $\|\mathbf{v}\|_\rho = \sqrt{\rho_{\mathbf{x}}(\mathbf{v}, \mathbf{v})}$ for any $\mathbf{v} \in T_{\mathbf{x}}\mathcal{M}$.

For a curve $\gamma : [a, b] \to \mathcal{M}$, we define the length of γ to be $L(\gamma) = \int_a^b \|\gamma'(t)\|_\rho \, dt$. For $\mathbf{x}, \mathbf{y} \in \mathcal{M}$, the distance $d(\mathbf{x}, \mathbf{y}) = \inf L(\gamma)$ where γ is any curve such that $\gamma(a) = \mathbf{x}, \gamma(b) = \mathbf{y}$. A geodesic $\gamma_{\mathbf{xy}}$ from $\mathbf{x}$ to $\mathbf{y}$, is a curve that minimizes this length.

For each point $\mathbf{x} \in \mathcal{M}$ and vector $\mathbf{v} \in T_{\mathbf{x}}\mathcal{M}$, there exists a unique geodesic $\gamma : [0, 1] \to \mathcal{M}$ where $\gamma(0) = \mathbf{x}, \gamma'(0) = \mathbf{v}$. The exponential map $\exp_{\mathbf{x}} : T_{\mathbf{x}}\mathcal{M} \to \mathcal{M}$ is defined as $\exp_{\mathbf{x}}(\mathbf{v}) = \gamma(1)$. Note that this is an isometry, *i.e.*, $\|\mathbf{v}\|_\rho = d\left(\mathbf{x}, \exp_{\mathbf{x}}(\mathbf{v})\right)$. The logarithmic map $\log_{\mathbf{x}} : \mathcal{M} \to T_{\mathbf{x}}\mathcal{M}$ is the inverse of $\exp_{\mathbf{x}}$.

B Further Results

B.1 Why can't we simply rotate encodings?

Liang et al. demonstrated that attempting the modality gap by shifting one modality to overlap the other in L2, then re-normalizing the encoding does not improve accuracy. This is perhaps not surprising given such a transformation is non-linear and does not preserve inter-modality distance (we confirm the conclusion holds when L2 is used as the distance to classify in Table 3 however). A reasonable distance preserving approach is to rotate one modality over to the other by solving an Orthogonal Procrustes (Gower & Dijksterhuis, 2004) problem that minimizes the distance between modality pairs. In the one-shot case we have $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_C]^T$ and $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_C]^T$,

$$\mathbf{R} = \arg\min_{\mathbf{\Omega}} \|\mathbf{\Omega}\mathbf{U} - \mathbf{Z}\|_F \ , \quad \text{subject to} \quad \mathbf{\Omega}^T\mathbf{\Omega} = \mathbf{I} \ . \tag{9}$$

$\mathbf{U}^* = \mathbf{R}\mathbf{U}$ can then be combined with $\mathbf{Z}$ as a single index, and standard classifiers can be applied. However, in practice this significantly reduces accuracy (see Fig. **??**). It is no immediately clear why, as this is approach is equivalent to subspace alignment (Fernando et al., 2014), as points represent the Principal Components (PC) in the few shot case as $K \ll D$. Using Tangent PCA we show why. Figures 9, 10 show the cosine similarity of the top 10 text PC's (rows) with image PC's (columns),

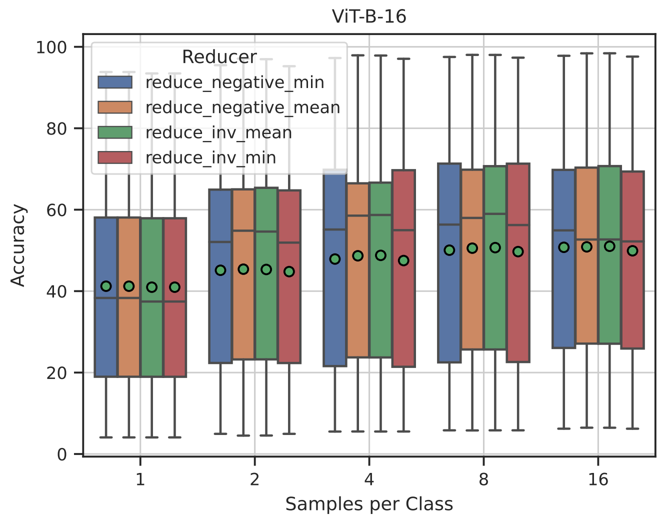

Figure 5: Comparison of various reduction functions to transform the collection of distances into logits for the image samples. The distribution over 29 datasets, averaged over 3 seeds, for each sample budget is shown. Means are green circles, medians center-lines. We use the inverse mean.

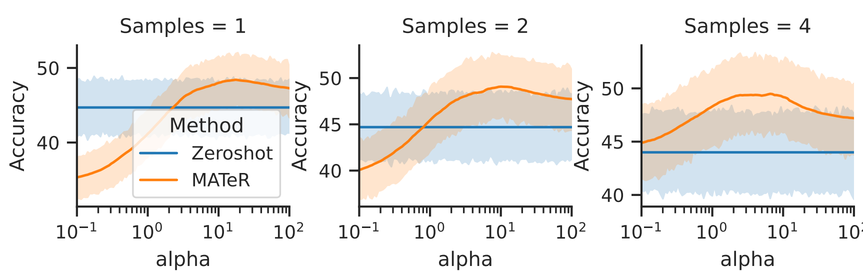

Figure 6: Aggregate effect of sharpness α, over all datasets, for various sample budgets (ViT-B-16 model used). Our chosen value $\alpha = 2$ is far from the optimal, and tuning would increase MATeR's accuracy several percentage points.

Table 3: Mean-shifting baseline with an RN50 encoder, for the 1-shot case only. Classification is done via minimum $l2$ distance on various index's. U' is the text encodings mean shifted to lie on the image manifold; $U' = U - \left(\bar{U} - \bar{Z}\right)$. '$U'$ and Z' refers to the case where both encodings are present in the same index and the standard (closet point) classifier is used. 'Ensemble Z, U' refers to the case where two index's are created and queried separately, and their resulting logits are added with equal weighting. These results differ from previous work (Liang et al.) as they preformed all classification with inner products only, and hence re-normalized after the mean shift; a non-linear transform which does not preserve relative distances between classes and reduces accuracy.

	Accuracy (5 seeds)					
RN50	Z	U	U'	U' and Z	Ensemble Z, U	Gain from U'
cars	23.34%	54.21%	49.37%	35.66%	37.77%	-4.84%
country211	4.00%	15.45%	12.66%	6.68%	5.89%	-2.78%
fer2013	19.24%	34.62%	45.84%	39.19%	23.57%	11.22%
fgvc_aircraft	13.26%	16.92%	16.82%	15.37%	16.47%	-0.10%
gtsrb	18.55%	35.15%	25.74%	21.01%	25.53%	-9.42%
mnist	38.67%	57.86%	60.84%	57.40%	52.41%	2.98%
renderedsst2	49.26%	55.74%	55.90%	54.37%	49.56%	0.16%
stl10	77.40%	94.21%	94.96%	93.23%	87.73%	0.74%
sun397	29.75%	59.69%	57.70%	37.13%	41.40%	-1.99%
voc2007	24.68%	64.31%	56.67%	45.86%	34.09%	-7.64%
caltech101	56.08%	83.01%	77.84%	69.22%	71.11%	-5.16%
cifar10	45.03%	72.08%	69.04%	65.75%	59.57%	-3.04%
cifar100	16.48%	38.84%	35.84%	24.84%	28.54%	-3.00%
clevr_closest_object_dist	23.41%	14.20%	22.08%	23.57%	23.87%	7.88%
clevr_count_all	19.79%	22.34%	16.90%	20.27%	20.47%	-5.45%
diabetic_retinopathy	17.04%	17.71%	6.47%	9.51%	16.57%	-11.24%
dmlab	18.88%	14.76%	14.20%	15.24%	18.05%	-0.56%
dsprites_label_orientation	9.80%	1.33%	2.06%	9.50%	9.32%	0.74%
dsprites_label_x_position	4.32%	3.30%	3.31%	4.18%	4.27%	0.02%
dtd	28.11%	41.97%	40.28%	38.59%	36.69%	-1.69%
eurosat	47.45%	40.06%	40.01%	49.21%	52.49%	-0.04%
flowers	53.18%	65.88%	51.51%	60.53%	69.06%	-14.37%
kitti_closest_vehicle_dist	39.95%	22.22%	22.32%	28.65%	42.13%	0.09%
pcam	65.74%	64.11%	66.54%	67.80%	66.48%	2.42%
pets	33.23%	83.42%	75.00%	57.66%	59.92%	-8.42%
resisc45	39.34%	46.25%	46.39%	44.35%	47.99%	0.14%
smallnorb_label_azimuth	6.70%	5.70%	5.74%	6.60%	6.67%	0.05%
smallnorb_label_elevation	13.27%	10.88%	10.95%	13.29%	13.24%	0.07%
svhn	11.56%	29.03%	24.77%	18.44%	14.29%	-4.26%
Mean	29.22%	40.18%	38.20%	35.62%	35.69%	-1.98%
Median	23.41%	38.84%	40.01%	35.66%	34.09%	-0.56%

ordered by explained variance. The direction of variance do not align; the embeddings contain information additional to the class being considered, and this additional information is not consistent across modalities. Aligning modalities via Orthogonal Procrustes will fit towards these directions of greater variance, reducing accuracy.

C IMPLEMENTATION DETAILS

We use no image augmentation in all experiments to facilitate comparison. Extending our approach in include image augmentation is straightfoward and likely to increase performance. For logistic regression, we use scikitlearn's implementation with the LBFGS solver and otherwise default hyper-parameters. Given all datasets are few-shot and hence low-sample, optimizing on CPU is very fast, and we found no need to learn the classifier on GPU.

To calculate the Fréchet norms and logarithmic maps we use the excellent `geomstats` (Miolane et al., 2020) package.

Table 4: Numerical results for Figure 2 right.

	Mean		**Median**	
	MATeR (Label only)	Zeroshot	MATeR (Label only)	Zeroshot
RN50	40.657	40.205	43.780	39.000
ViT-B-16	46.565	44.697	52.264	46.294
ViT-H-14	54.665	53.908	62.431	58.361
ViT-g-14	54.008	54.618	61.960	62.366
Mean	48.974	48.357	55.109	51.505
Median	50.286	49.302	57.112	52.328
Std. Dev.	6.651	7.068	8.888	10.778

Table 5: MATeR label only accuracy per dataset when using a single shot per class to inform the Fréchet mean only using the CLIP-ViT-B-16 backbone. Results averaged over 5 seeds.

	MATeR (Label only)	Zeroshot	Difference
Caltech101	86.078	88.562	-2.484
Cars	58.906	64.582	-5.676
Cifar10	91.468	90.760	0.708
Cifar100	69.580	67.480	2.100
Clevr Closest Object Distance	15.594	14.814	0.780
Clevr Count All	27.249	20.371	6.877
Country211	20.812	22.858	-2.045
Diabetic Retinopathy	24.078	3.026	21.053
Dmlab	18.684	15.945	2.739
Dsprites Label Orientation	1.570	2.317	-0.746
Dsprites Label X Position	3.123	2.939	0.184
Dtd	44.383	44.840	-0.457
Eurosat	61.241	54.852	6.389
Fer2013	52.711	46.294	6.417
Fgvc Aircraft	22.550	24.362	-1.812
Flowers	62.333	71.078	-8.745
Gtsrb	46.215	43.413	2.803
Kitti Closest Vehicle Distance	39.716	22.222	17.494
Mnist	67.900	51.390	16.510
Pcam	54.090	51.834	2.256
Pets	81.168	87.364	-6.196
Renderedsst2	56.716	60.461	-3.745
Resisc45	63.171	59.603	3.568
Smallnorb Label Azimuth	5.541	5.646	-0.105
Smallnorb Label Elevation	10.808	11.374	-0.566
Stl10	98.275	98.263	0.013
Sun397	60.835	64.290	-3.455
Svhn	34.854	27.559	7.295
Voc2007	70.735	77.704	-6.970
Mean	46.565	44.697	1.868
Median	52.711	46.294	6.417
Std. Dev.	27.535	29.523	-1.989

Table 6: Mean zero-shot performance across architectures and pretraining methods for all datasets with > 1 prompt ($N = 18$). Fréchet averaging results in a minor, but consistent gain over the euclidean mean re-projected to the unit hyper-sphere. This is expected as prompt templating should not spread label encodings across a large area (at least, not greater than the inter-modality class-to-class distance).

Model, Pretraining	Fréchet	Euclidean w/ L2-Norm
RN50, openai)	**41.35**	41.31
ViT-B-32, openai	45.38	**45.39**
ViT-B-16, openai	**48.27**	48.24
ViT-L-14, openai	**53.90**	53.84
ViT-L-14, laion2b_s32b_b82k	**56.98**	56.95
ViT-H-14, laion2b_s32b_b79k	**59.05**	59.04
ViT-g-14, laion2b_s12b_b42k	**64.08**	64.06
Mean	**52.71**	52.69
Median	**53.90**	53.84

Table 7: Comparison of Fréchet prompt averaging with the standard approach by mean zero-shot performance for CLIP-RN50 broken down by dataset. Only datasets with multiple prompt templates are listed.

Prompt Averaging Method Dataset	Frechet	Linear Reprojected	Difference
Caltech101	83.007	83.007	0.000
Cars	54.334	54.210	0.124
Cifar10	72.080	72.080	0.000
Cifar100	39.000	38.840	0.160
Country211	15.445	15.445	0.000
Dsprites Label Orientation	1.267	1.325	-0.058
Dtd	41.915	42.021	-0.106
Eurosat	40.074	40.056	0.019
Fer2013	34.675	34.620	0.056
Fgvc Aircraft	16.922	16.922	0.000
Gtsrb	35.154	35.154	0.000
Pcam	64.114	64.111	0.003
Resisc45	46.286	46.254	0.032
Smallnorb Label Azimuth	5.720	5.695	0.025
Smallnorb Label Elevation	10.947	10.881	0.066
Stl10	94.213	94.213	0.000
Sun397	59.687	59.687	0.000
Svhn	29.375	29.034	0.341
Mean	41.345	41.309	0.037
Median	39.537	39.448	0.002
Std. Dev.	26.372	26.381	0.096

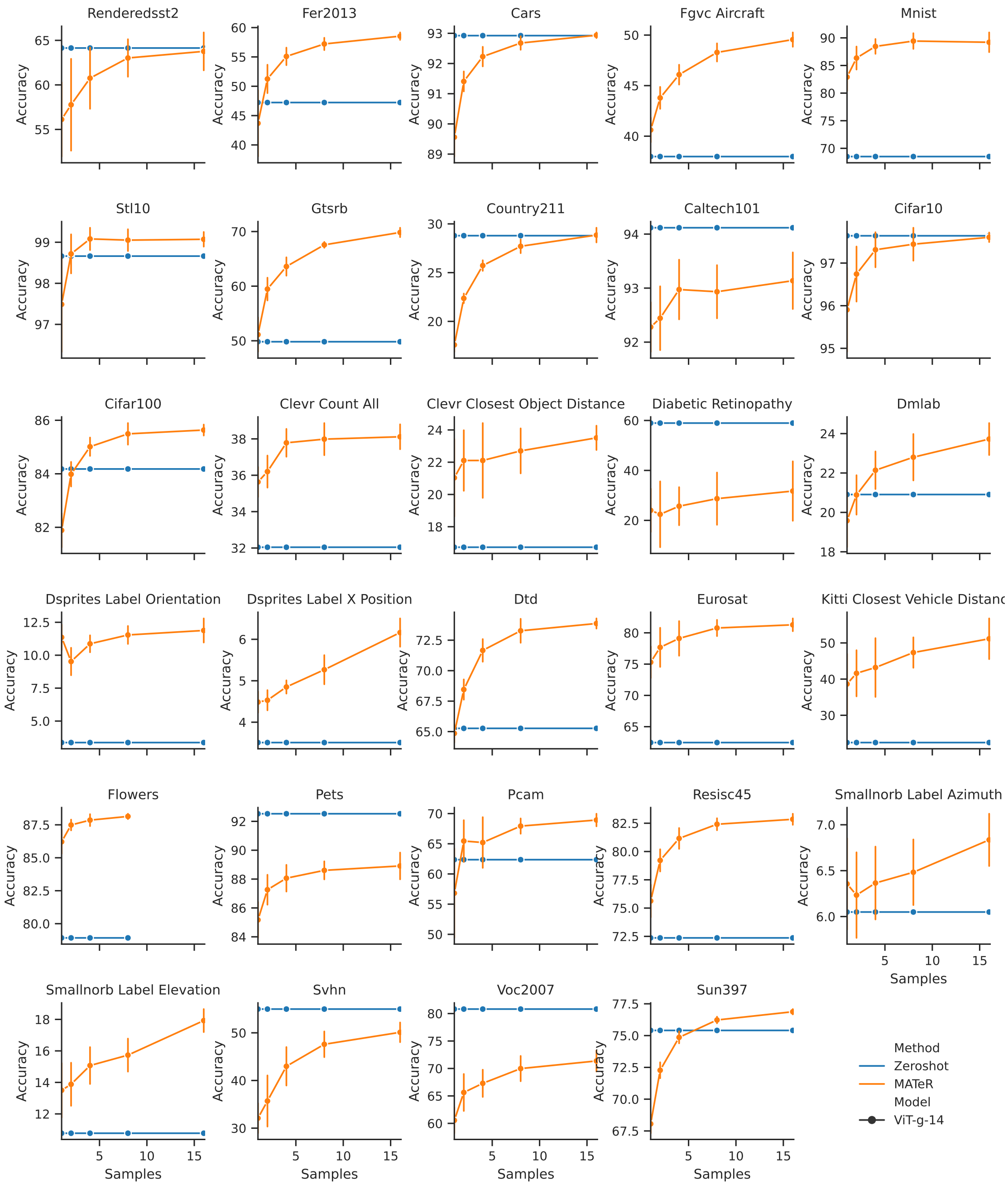

Figure 7: MATeR vs zero-shot accuracy over 29 benchmark datasets for CLIP-ViT-g-14. Error bars show standard deviation over 10 seeds. Note y-axis is not consistent across subplots.

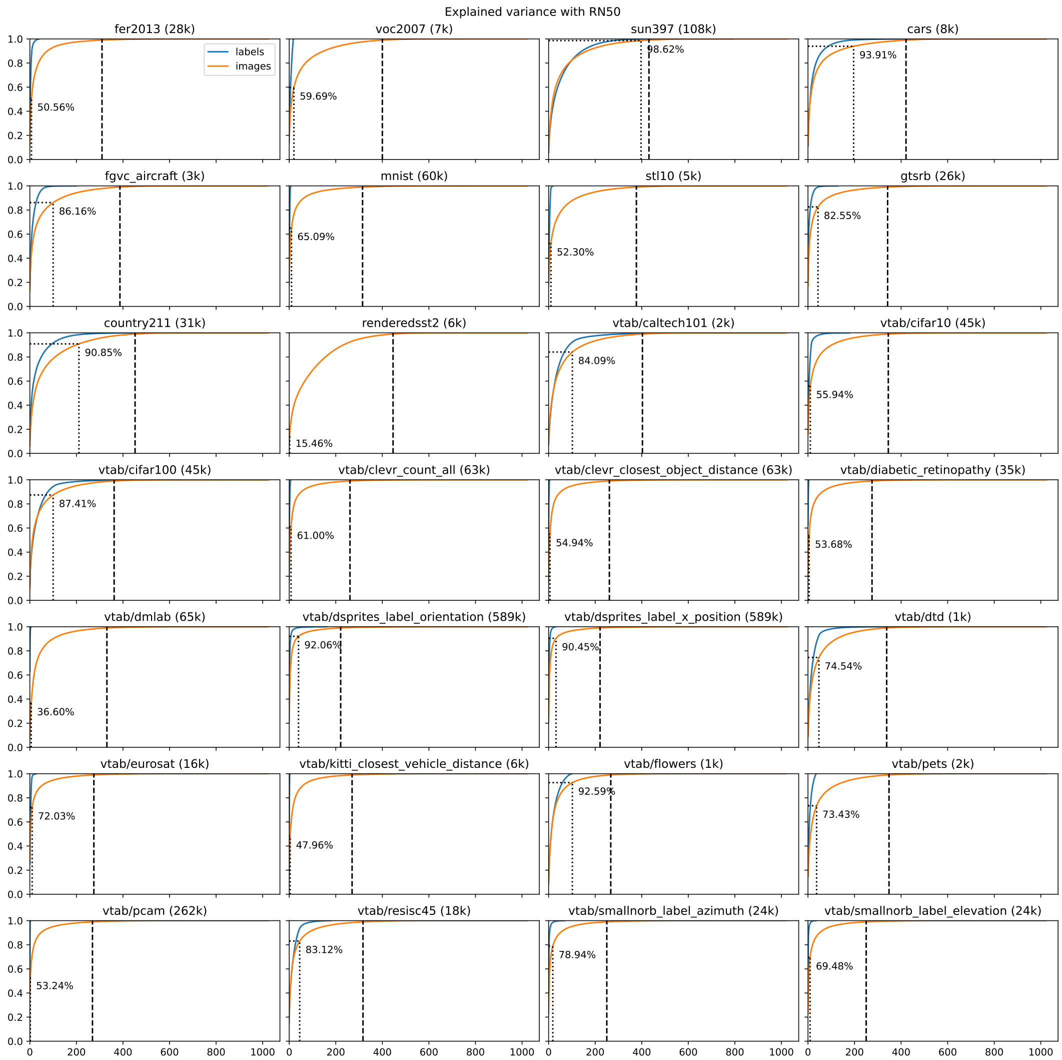

Figure 8: Explained variance (y-axis) vs number of principal components (x-axis) of OpenAI trained CLIP embeddings for various downstream datasets. Image encoder is RN50. Dotted line shows explained variance when number of components is equal to the number of classes. Dashed line is number of components where explained variance is >99%. Number of samples per dataset shown in brackets in subplot titles.

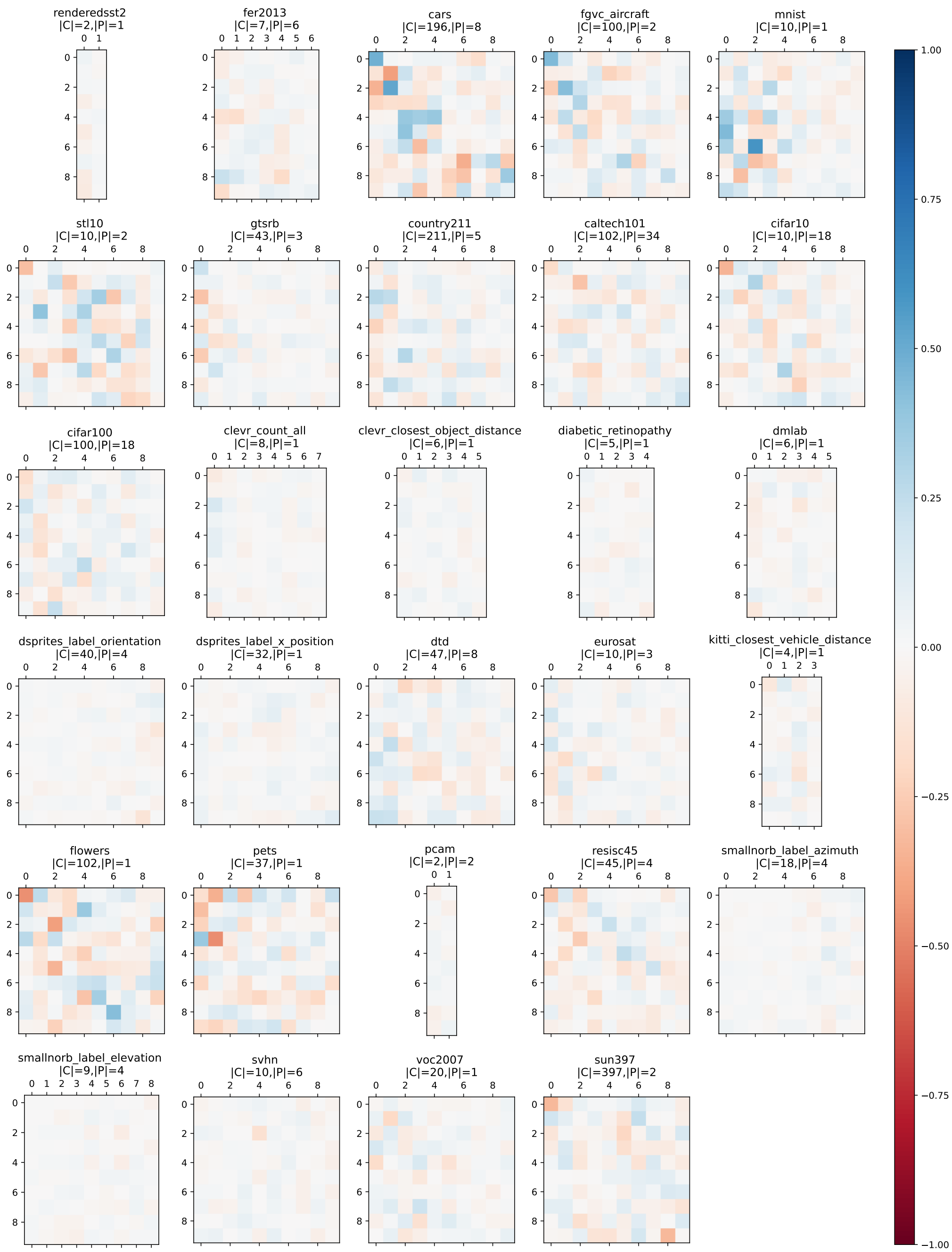

Figure 9: Alignment of the principal components between normalized the text and image encodings with an RN50 model. The full training set for the image encodings is used. Alignment is shown as the cosine similarity between the ith text PC (rows) and $j$the text PC (columns). Text encodings are first normalized, averaged over prompts, and normalized again. The top 10 PC's, ordered by explained variance, are shown per dataset unless there fewer than 10 samples for that modality, in which case all PC's are shown, as occurs when the number of classes is <10. In the case of perfectly aligned, but modality shifted distributions, the identity matrix is expected. High off-diagonal values indicate that while the PC is aligned, they do not explain the same relative amount of variance, and there are spurious directions of variance in at least one modality.

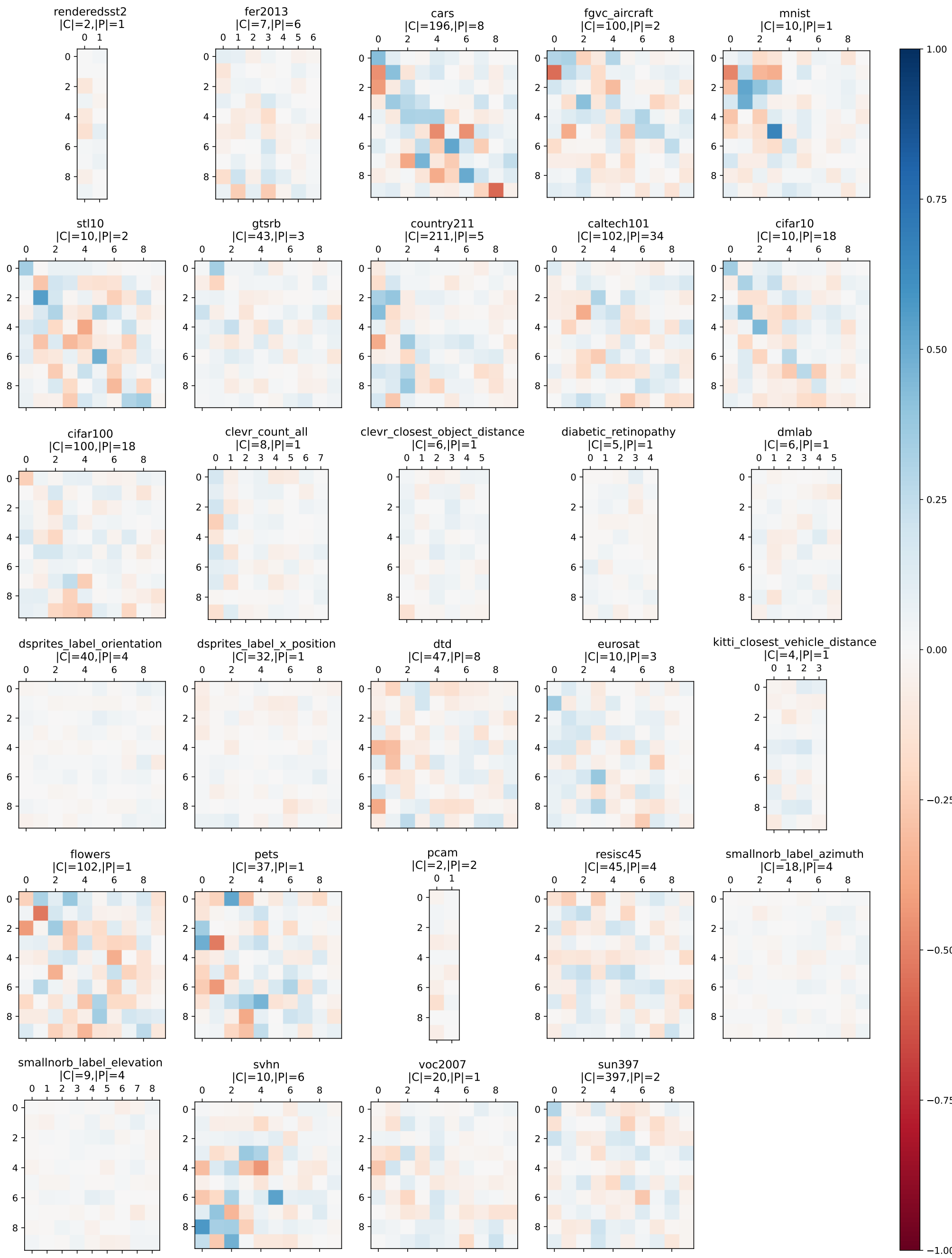

Figure 10: Alignment of the principal components between normalized the text and image encodings with an ViT-L-14 (224) openai model.

Table 8: Baseline zero-shot accuracy (%) for various OpenAI pretrained models. Predictions are made based on the closest label embedding (averaged over prompt templates), using maximum cosine similarity.

Dataset	RN50	ViT-B-16	ViT-B-32	ViT-L-14
Caltech101	82.68	85.95	82.68	89.87
Cars	59.27	71.73	63.60	80.09
Cifar10	78.92	94.12	91.72	96.68
Cifar100	50.46	70.44	66.16	78.06
Clevr Closest Object	27.84	27.13	29.14	30.24
Clevr Count All	33.19	35.11	34.10	36.69
Country211	18.77	24.16	20.47	30.17
Diabetic Retinopathy	61.79	63.12	61.45	63.47
Dmlab	30.21	33.05	30.83	37.02
Dsprites Label Orientation	82.22	72.13	67.53	81.88
Dsprites Label X Position	48.23	48.12	50.69	38.60
Dtd	61.91	66.54	61.54	70.59
Eurosat	85.85	90.70	89.20	94.00
Fer2013	59.88	63.51	60.53	63.37
Fgvc Aircraft	29.73	42.18	32.88	49.56
Flowers	81.67	89.71	83.63	97.45
Gtsrb	63.56	72.28	69.61	84.71
Kitti Closest Vehicle	47.04	54.37	56.97	50.59
Mnist	95.77	96.80	96.44	98.10
Pcam	71.33	73.22	73.01	76.60
Pets	74.32	82.88	78.26	90.08
Renderedsst2	61.72	62.22	59.53	67.00
Resisc45	83.10	90.56	87.25	93.13
Smallnorb Label Azimuth	14.02	13.38	13.03	12.26
Smallnorb Label Elevation	28.75	26.95	28.85	25.14
Stl10	96.14	98.90	98.08	99.44
Sun397	100.00	100.00	100.00	100.00
Svhn	34.62	42.94	31.56	50.75
Voc2007	70.37	77.96	75.28	81.94
Mean	59.77	64.49	61.86	67.84
Median	61.79	70.44	63.60	76.60
Std. Dev.	24.69	25.26	25.17	26.37

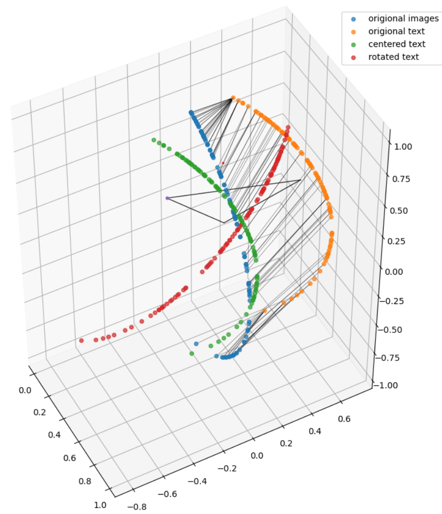

Figure 11: Visualization of why naive mean shifting, and rotation to reduce the modality gap fails on 3D on dummy data. Here, we set the intrinsic dimentionality of the modalities to 1, and normalize. 'Text' is shown as orange points and 'images' as blue points. For easy of visualization we link each point to it's closet point across the modality gap. Modality means and links to the origin is also show (heavy black lines). Orthogonal Procrustes computed on the mean points (red) fails due to the undercontrained nature of the rotation (2 points in 3d space), and rotation over additional axis occurs which destroys the relative distance between points. Mean centered shifting (green) does better, but is distorting as the translation occurs in l2.

C.1 Dataset selection

The 29 datasets were chosen so as to facilitate comparison to prior work, but also to cover a range of zero-shot and few-shot performance. STL10, for example, was included due to CLIP's 94.8 percent zero-shot accuracy even when using an RN50 based architecture. Additionally, the datasets present not only domain shifts but *task* shifts such as distance estimation (KITTI), counting (Clevr Count All) and orientation estiation (Smallnorb, Dsprites). The datasets contain a varying number of classes from 397 for SUN397 to binary classification in the case of Renderedsst.

In VTAB, we use KITTI v3.3.0 not 3.2.0 due to incompatibility with latest verison of task-adaptation lib. Original CLIPBaselines lib uses 3.2.0. We were unable to evaluate on the SUN397 vtab version due to an undecodabel image. We use the pytorch datasets SUN397 version instead to complete the VTAB evaluation set. All prompt templates and dataset label string representations are obtained from LIAON's `CLIPbenchmark`.

C.2 Model Details and Preprocessing

We adopt the same input input transforms used in pretrain for various models and ensure consistency between train and test, with fixed-constant normalization (per model), bi-cubic interpolation to the model input size, and center cropped.

D Computation

A disadvantage of the retreival approach is that computational complexity of the inference step scales logarithmically with downstream dataset size (although this is somewhat balanced by the fact training is free). In the few-shot case this is almost never a problem unless there are huge numbers of classes. For large datasets, there may be practically a minor slowdown, however we observed for all datasets in this paper, this was minor. In short; we did not find it to be a major concern.

Table 9: Dataset details. Lower datasets are the VTAB (Zhai et al., 2019) versions.

Dataset	Abbreviation	Test size	Number of classes
Stanford Cars	cars	8,041	196
Country211		21,100	211
Facial Emotion Recognition 2013	fer2012	7,178	7
FGVC Aircraft		3,333	100
GTSRB		12,630	43
MNIST		10,000	10
RenderedSST2		1,821	2
STL10		8,000	10
SUN397		108,754	397
Pascal VOC 2007 Classification	voc2007	14,976	20
Caltech-101		6,085	102
CIFAR-10	cifar10	10,000	10
CIFAR-100	cifar100	10,000	100
CLEVR Object Distance		15,000	6
CLEVR Counts		15,000	8
Diabetic Retinopathy		42,670	5
DMLAB		22,735	6
DSPRITES Orientation		73,728	40
DSPRITES Position		73,728	32
Describable Textures	dtd	1,880	47
EuroSAT		5,400	10
Oxford Flowers 102	flowers	6,149	102
KITTI closest vehicle distance		711	4
PatchCamelyon	pcam	32,768	2
Oxford-IIIT Pets	pets	3,669	37
RESISC45		6,300	45
SmallNORB Azimuth		12,150	18
SmallNORB Elevation		12,150	9
SVHN		26,032	10

Table 10: Base model details. GMAC refers to number of Giga Multiply–ACcumulate operations. We use the implementations of Ilharco et al. (2021).

Name	Embedding Dimension	Layers	Heads	Parameters (M)	GMACs
RN50	1024	-	-	102	9.16
ViT-B-32	512	12	8	151	7.40
ViT-B-16	512	12	8	150	20.57
ViT-L-14	768	12	12	428	87.73
ViT-H-14	1024	24	16	986	190.97
ViT-g-14	1024	24	16	1370	290.74

Given we only require forward passes for all experiments, with no finetuning of the base encoder, we were able to perform all experiments on a single G5.16x instance in the AWS cloud. We cache features for both the images and the various label-prompt combinations.