Improving fine-grained understanding in image-text pre-training

Ioana Bica [1]   **Anastasija Ilić** [1]   **Matthias Bauer** [1]   **Goker Erdogan** [1]   **Matko Bošnjak** [1]   **Christos Kaplanis** [1]
Alexey A. Gritsenko [2]   **Matthias Minderer** [2]   **Charles Blundell** [1]   **Razvan Pașcanu** [1]   **Jovana Mitrović** [1]

Abstract

We introduce SPARse fine-grained Contrastive alignment (SPARC), a simple method for pre-training more fine-grained multimodal representations from image-text pairs. Given that multiple image patches often correspond to single words, we propose to learn a grouping of image patches for every token in the caption. To achieve this, we use a sparse similarity metric between image patches and language tokens and compute for each text token a *language-grouped vision embedding* as the weighted average of patches. The token and language-grouped vision embeddings are then contrasted through a fine-grained sequence-wise loss that only depends on individual samples and does not require other batch samples as negatives, i.e., more detailed information is encoded in a computationally inexpensive way. SPARC combines this fine-grained loss with a contrastive loss between global image and text embeddings to learn representations that simultaneously encode global and local information. We thoroughly evaluate SPARC and show improved performance over competing approaches both on image-level tasks relying on coarse-grained information, e.g. classification, as well as region-level tasks relying on fine-grained information, e.g., retrieval, object detection, segmentation while also improving model faithfulness and captioning in foundational vision-language models.

1. Introduction

Contrastive pre-training from large-scale, noisy image-text datasets (Radford et al., 2021; Jia et al., 2021) has become a widely used paradigm for learning general vision repre-

---
[*]Equal contribution [1]Google DeepMind, London, UK [2]Google DeepMind, Zurich, Switzerland. Correspondence to: Ioana Bica <ioanab@google.com>, Jovana Mitrović <mitrovic@google.com>.
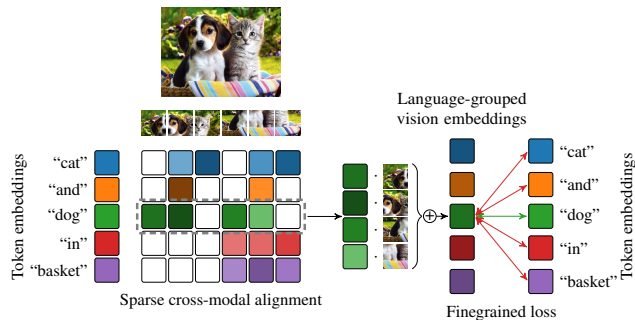
Figure 1: SPARC learns a *language-grouped vision embedding* for every token as the alignment-weighted sum of patches that are most similar to that token. We compute a sparse similarity metric between patches and tokens of individual image-text pairs (left) and compute the resulting alignment weights (middle). We contrast the *language-grouped vision embeddings* with token embeddings in a fine-grained contrastive sequence-wise loss (right).

sentations for a wide range of downstream tasks as well as for learning vision encoders in multimodal foundation models (Alayrac et al., 2022; Chen et al., 2022; Li et al., 2022a). In particular, these models achieve impressive performance on image-level tasks like classification (Radford et al., 2021), coarse-grained retrieval and visual question answering (Alayrac et al., 2022; Chen et al., 2022). On the other hand, these models have been shown to discard fine-grained visual information (Krojer et al., 2022) and work poorly on tasks involving localization (Zhong et al., 2022; Ranasinghe et al., 2022), counting (Paiss et al., 2023) and understanding spatial relationships between objects (Parcalabescu et al., 2021) or object attributes (Yuksekgonul et al., 2022). A recent line of work explores incorporating losses between image patch and text token embeddings (Yao et al., 2021; Mukhoti et al., 2023; Huang et al., 2021; Wang et al., 2022) to learn representations encoding more fine-grained details. Specifically, these local losses learn soft correspondences between image patches and text tokens from image-text pairs by aligning patches corresponding to individual objects in the image to tokens corresponding to the words describing these objects.

While these models have achieved improved performance on some fine-grained tasks, they are computationally and mem-

ory expensive, unstable during training (Yao et al., 2021) and/or rely on pretrained models to kickstart learning (Wang et al., 2022; Huang et al., 2021; Mukhoti et al., 2023).

In this work, we propose *SPARse fine-grained Contrastive alignment (SPARC)*, a novel objective for multimodal pre-training which learns representations that encode both coarse-grained/global and fine-grained/local information. We propose to build *language-grouped vision embeddings* by learning to aggregate (in an unsupervised way) image patches corresponding to individual words in the caption; this is motivated by the observation that usually multiple image patches correspond to one word in the caption. As a first step, SPARC computes the similarity between the patch and token embeddings of an individual image-text pair and enforces sparsity in the resulting similarity matrix. This sparsification enables only the most relevant image patches to be attributed to individual tokens. Next, as illustrated in Figure 1, for every token, we compute the corresponding language-grouped vision embedding as the alignment-weighted sum of the patch embeddings, with the alignment weights computed from the sparsified similarity matrix. The resulting language-grouped vision embeddings are contrasted with the token embeddings from the same image-text pair by optimizing for the similarity between individual token embeddings and the corresponding language-grouped vision embeddings and dissimilarity to all other language-grouped vision embeddings. SPARC combines the resulting fine-grained/local contrastive loss with a global contrastive loss between image and text embeddings which enables it to simultaneously encode global and local information in the learned representations.

Through its design choices, SPARC addresses several short-comings of existing methods for learning image representations with more fine-grained information. Firstly, several of these methods (Yao et al., 2021; Mukhoti et al., 2023; Huang et al., 2021) learn representations with fine-grained losses that compute similarities between all image patch embeddings and all text token embeddings in a batch. This approach is both computationally and memory intensive and does not scale to large batch sizes (which are needed for obtaining good performance for contrastive methods (Radford et al., 2021)). On the other hand, SPARC contrasts patch and token embeddings at the level of individual image-text pairs and does not use other examples from the batch as negatives which enables it to more easily scales to large batch sizes. Secondly, for learning soft correspondences between image patches and text tokens, prior work (Mukhoti et al., 2023; Huang et al., 2021; Wang et al., 2022) usually relies on building cross-modal weighted representations with weights computed as a softmax over patch and token embedding similarities. The winner-takes-all dynamics of softmax (Peterson & Söderberg, 1989; Elfadel & Wyatt Jr, 1993) strongly bias learning towards one-to-one mappings

between individual tokens and patches which often does not correspond to real-world data, e.g. in an image of a dog, the token embedding for "dog" should be matched with *all* patch embeddings that correspond to the dog in the image and not just one. On the flip side, SPARC does not use softmax for calculating the alignment weights which allows it to learn a flexible one-to-many matching between individual tokens and the corresponding patches and to avoid the winner-take-all dynamics of softmax. Thirdly, several of these approaches start from contrastively pre-trained vision-language models (Mukhoti et al., 2023) or from pre-trained language models (Huang et al., 2021; Wang et al., 2022). Moreover, existing fine-grained objectives have been developed in different communities (i.e. medical (Huang et al., 2021; Wang et al., 2022) vs. general vision (Yao et al., 2021; Mukhoti et al., 2023)) leveraging different types and sizes of datasets, architectures and pretraining setups. This makes it difficult to compare different approaches and assess the benefits of using individual fine-grained objectives.

To summarize, our main contributions are as follows:

- We propose SPARC, a novel method for pre-training multimodal models on large-scale noisy image-text data which learns both coarse-grained and fine-grained information.

- Through an extensive experimental evaluation, we show that SPARC significantly improves performance on both fine-grained and coarse-grained downstream tasks over competing methods.

- We perform a thorough like-for-like comparison on the benefits of different fine-grained objectives for large-scale pretraining of multimodal models.

2. Sparse Fine-grained Contrastive Alignment

Let $\mathcal{B} = \{(\boldsymbol{x}_1^v, \boldsymbol{x}_1^t), (\boldsymbol{x}_2^v, \boldsymbol{x}_2^t), \ldots, (\boldsymbol{x}_B^v, \boldsymbol{x}_B^t)\}$ be a mini-batch of image-text pairs. Let $f_v(\cdot)$ be the image encoder, $f_t(\cdot)$ the text encoder and $g_v(\cdot)$ and $g_t(\cdot)$ linear adaptors. For an image $\boldsymbol{x}_i^v$, we denote the corresponding patches as $(\boldsymbol{x}_{i,1}^v, \boldsymbol{x}_{i,2}^v, \ldots, \boldsymbol{x}_{i,P}^v)$ and the patch embeddings as $(\boldsymbol{v}_{i,1}, \boldsymbol{v}_{i,2}, \ldots, \boldsymbol{v}_{i,P})$ with $\boldsymbol{v}_{i,p} = g_v(f_v(\boldsymbol{x}_{i,p}^v)) \in \mathbb{R}^d$ and P the number of patches.

We calculate the global vision embedding as $\overline{\boldsymbol{v}}_i = g_v(h_v(\text{avg\_pool}(\{f_v(\boldsymbol{x}_{i,p}^v)\}_{p=1}^P)))$ with h_v being a single non-linear layer that facilitates the encoding of different granularities of information. For the corresponding text $\boldsymbol{x}_i^t$, we denote the tokens as $(\boldsymbol{x}_{i,1}^t, \boldsymbol{x}_{i,2}^t, \ldots, \boldsymbol{x}_{i,L_i}^t)$ with L_i the number of tokens for sample i. The token embeddings $(\boldsymbol{t}_{i,1}, \boldsymbol{t}_{i,2}, \ldots, \boldsymbol{t}_{i,L_i})$ are computed as $\boldsymbol{t}_{i,l} = g_t(f_t(\boldsymbol{x}_{i,l}^t))$ and the global text embedding $\overline{\boldsymbol{t}}_i$ is computed as $\overline{\boldsymbol{t}}_i = g_t(\text{avg\_pool}(\{f_v(\boldsymbol{x}_{i,l}^t)\}_{l=1}^{L_i}))$.
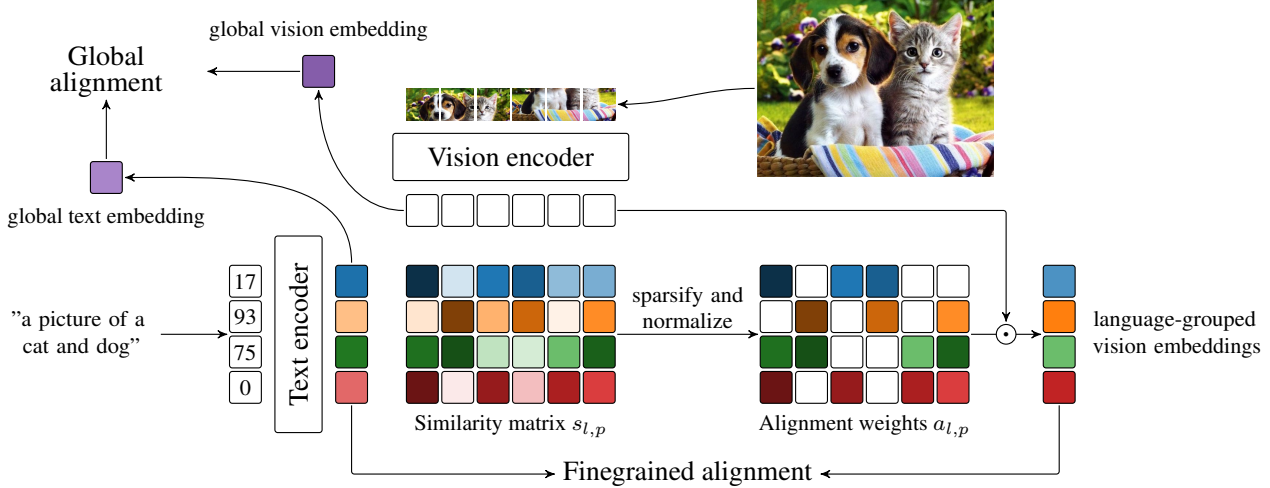
Figure 2: Overall architecture for SPARC. The global alignment loss maximizes the similarity between the global vision and global text embeddings, while minimizing the similarity with the other global embeddings in the batch. To obtain the finegrained alignment, we compute the similarity between the patch embeddings and the token embeddings and then sparsify and normalize the resulting similarity matrix to obtain alignment weights. These alignment weights are then used to group the patch embeddings. The resulting language-grouped vision embeddings are then contrasted to the token emebddings in a sequence-wise finegrained alignment loss.

Global alignment. In order to learn global information, SPARC uses the global contrastive loss (Radford et al., 2021) which operates at the level of global image ($\overline{v}$) and global text embeddings ($\overline{t}$). Specifically, we learn image and text embeddings by maximizing the similarity to the corresponding text and image embeddings, respectively, while minimizing the similarity to other text and image embeddings in the batch, i.e. we optimize:

$$L_g = -\frac{1}{2B} \sum_{i=1}^{B} \left(\log \frac{\exp(\phi(\overline{v}_i, \overline{t}_i)/\tau)}{\sum_{j=1}^{B} \exp(\phi(\overline{v}_i, \overline{t}_j)/\tau)} + \log \frac{\exp(\phi(\overline{t}_i, \overline{v}_i)/\tau)}{\sum_{j=1}^{B} \exp(\phi(\overline{t}_i, \overline{v}_j)/\tau)} \right), \quad (1)$$

with $\phi(\overline{v}_i, \overline{t}_j) = \frac{\overline{v}_i}{\|\overline{v}_i\|_2} \cdot \frac{\overline{t}_j}{\|\overline{t}_j\|_2}$ and τ as temperature.

Finegrained alignment. As usually multiple image patches correspond to one word in the caption, we propose to learn groupings of patches that correspond to individual text tokens. For every token embedding we learn a corresponding *language-grouped vision embedding* as an alignment-weighted sum of patches that encode that token in the visual domain. We propose to compute the alignment weights based on the similarity between token and patch embeddings of the corresponding image-text pair. To facilitate the grouping of appropriate patch embeddings given a text token we sparsify and min-max normalize the similarity matrix to compute the alignment weights. We propose a fine-grained local loss that optimizes for the alignment between individual token embeddings and their cor-

responding language-grouped vision embeddings within a given image-text pair. Specifically, we propose a sequence-wise contrastive loss to optimize this fine-grained alignment within SPARC. Optimizing this loss (in addition to the global contrastive loss above) biases the learned representation to preserve detailed information about the image (as described by the caption) instead of just the global information sufficient to minimize the global contrastive loss. For an image-text pair, let $s_{i,lp}$ represent the similarity between text token embedding $t_{i,l}$ and image patch embedding $v_{i,p}$, i.e. $s_{i,lp} = t_{i,l} \cdot v_{i,p}$, where $s_{i,lp} \in \mathbb{R}$ and $\cdot$ is the inner product. Going forward we drop the example index i for simplicity. To obtain alignment weights, for each token j, we first normalize s_{lp} to $[0, 1]$ using min-max normalization across columns (i.e. patches):

$$\hat{s}_{lp} = \frac{s_{lp} - \min_k s_{lk}}{\max_k s_{lk} - \min_k s_{lk}}. \quad (2)$$

We sparsify the similarity matrix $S = (\hat{s}_{jk})_{1 \le j \le L, 1 \le k \le P}$ to facilitate learning and to encourage each token to be aligned to a few of the patches, i.e.

$$\tilde{s}_{jk} = \begin{cases} \hat{s}_{jk} & \text{if } \hat{s}_{jk} \ge \sigma \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

with σ the sparsity threshold. We compute the weights as

$$a_{jk} = \frac{\tilde{s}_{jk}}{\sum_{r=1}^{R} \tilde{s}_{jr}} \quad (4)$$

where a_{jk} is the weight of patch k for computing the language-grouped vision embedding corresponding to token

j and R is the number of patches with non-zero alignment weight. Note that this approach enables a flexible mapping between a token and arbitrarily many patch embeddings encoding that token in the visual domain, e.g. all image patches corresponding to "dog" can be matched to the token encoding "dog". For every token t_l we compute the corresponding language-grouped vision embedding $\boldsymbol{c}_l$ as

$$\boldsymbol{c}_l = \sum_{r=1}^{R} a_{lr} \boldsymbol{v}_r, \tag{5}$$

i.e. the alignment-weighted combination of corresponding patch embeddings. To learn fine-grained information we propose to optimize the alignment between token embeddings and their corresponding language-grouped vision embeddings. Specifically we propose a fine-grained contrastive loss that operates over sequences of tokens and patches at the level of each image-text pair and does not require negatives from other image-text pairs. This considerably reduced computation and memory costs over previous methods (Yao et al., 2021; Huang et al., 2021) that require samples from the whole batch in order to compute their fine-grained losses. SPARC optimizes the following fine-grained alignment contrastive loss:

$$L_f = -\frac{1}{2B} \sum_{i=1}^{B} \left[\frac{1}{L_i} \sum_{j=1}^{L_i} \left(\log \frac{\exp(\phi(\boldsymbol{c}_{ij}, \boldsymbol{t}_{ij})/\tau)}{\sum_{k=1}^{L_i} \exp(\phi(\boldsymbol{c}_{ij}, \boldsymbol{t}_{ik})/\tau)} + \right. \right.$$
$$\left. \left. \log \frac{\exp(\phi(\boldsymbol{t}_{ij}, \boldsymbol{c}_{ij})/\tau)}{\sum_{k=1}^{L_i} \exp(\phi(\boldsymbol{t}_{ij}, \boldsymbol{c}_{ik})/\tau)} \right) \right], \tag{6}$$

which maximizes the similarity of every token embedding with its corresponding language-grouped vision embedding and minimizes the similarity to other language-grouped vision embeddings in the sequence and vice versa.

Overall objective. The overall SPARC objective is a weighted sum of the global contrastive loss and the fine-grained alignment constrastive loss:

$$L_{\text{SPARC}} = \lambda_g L_g + \lambda_f L_f \tag{7}$$

where λ_g and λ_f are hyperparameters. We provide the pseudo-code for SPARC in Appendix C.

Sparsity threshold. We choose the sparsity threshold σ to be equal to $1/P$ with P the number of image patches. This choice is motivated by the consideration that every text token should attend to at least one image patch. Since we use the min-max normalization and the number of patches is constant, the smallest similarity of $1/P$ is achieved when all patches are equally similar. Note that this threshold naturally allows for the number of patches corresponding to one token to considerably vary between tokens within an image as well as across images; this enables the same class of objects (e.g. "dogs") to be appropriately represented irrespective of the difference in sizes, scales and shapes across different instances within and across images. Note that the threshold allows for the decoupling of similarities of individual patches to different tokens as it allows for different number of zero entries in different rows of the similarity matrix; thus, whether and how much a patch is similar to a token, has no bearing to how similar it is to a different token which is useful e.g. in situations when we have more detailed captions (e.g. "large brown dog") and/or when a single word is represented by multiple tokens.

3. Related Work

Contrastive image-text pre-training CLIP (Radford et al., 2021) and ALIGN (Jia et al., 2021) popularized learning general visual representations by leveraging textual supervision from noisy large-scale data scrapped from the internet. These methods learn representations through a contrastive objective that maximises the similarity between representations of the whole image and the full text of matched image-text pairs and minimizes the similarity between the remaining image-text pairs within the batch. However, learning visual representations through matching the global image and text embeddings can result in a coarse visual representation that discards many fine-grained details (i.e all details that are not needed for differentiating the matching global text embedding from the other text embeddings in the batch). To address this problem, FILIP (Yao et al., 2021) proposes a *cross-modal late interaction mechanism*, which optimizes the token-wise maximal similarity between image and text tokens through a contrastive objective. While this approach achieves a finer-grained alignment between image patches and words in the caption, computing the token-wise similarity between all image and text tokens in the batch becomes memory inefficient for large batch sizes. Moreover, FILIP (Yao et al., 2021) suffers from training instabilities as noted in the original paper. A related approach PACL (Mukhoti et al., 2023) starts from frozen CLIP-pretrained vision and text encoders and trains on top an adapter to obtain better fine-grained understanding.

In a parallel stream of work, several methods from the medical literature learn visual representation using medical image - radiology report pairs from small scale datasets (consisting of up to approximately 200k data points) (Huang et al., 2021; Wang et al., 2022; Dawidowicz et al., 2023). GLoRIA (Huang et al., 2021) builds localized visual representations by contrasting attention-weighted patch embeddings with text tokens, where the attention weights are computed through softmax on the similarity matrix between the patch and token embeddings. Similarly to FILIP, GLoRIA requires computing the similarity between all patch and to-

ken embeddings within the batch which is computationally intensive and does not scale to large batch sizes. Alternatively, MGCA (Wang et al., 2022) considers a token-wise fine-grained loss that employs a bidirectional multi-head attention strategy to learn the matching between patch and token embedding. While this is more efficient to compute, learning these matchings through a bidirectional multi-head cross-attention adds more parameters to the dual encoders, involves tuning several additional hyperparameters and suffers from the same problems of using softmax for computing the attention weights. MGCA also uses a domain-specific disease-level alignment loss that enforces a cluster assignment consistency to leverage inter-subject semantic correspondences. More recent methods (Dawidowicz et al., 2023) incorporate into the pre-training objective not only fine-grained losses similar to the ones used in GLoRIA and MGCA, but also domain-specific features and image views. Note that these methods from the medical literature start from a text encoder pre-trained with medical texts (Alsentzer et al., 2019), while we consider the case of pre-training the image and text encoders jointly from scratch.

Fine-grained understanding in vision-language models. Alternative approaches for improving fine-grained capabilities of vision-language models require pre-trained modules, specialised networks and human annotations. One line of work, proposes matching image regions to textual descriptions through contrastive losses, where the image regions - text description pairs are obtained from human annotations (Li et al., 2022b) or by using region proposal networks (Ren et al., 2015) or various text matching approaches (Zhong et al., 2022; Varma et al., 2023). A separate line of work adds a cross-modal encoder (thus adding a significant number of additional parameters) on top of the image and text encoders and uses captioning (Yu et al., 2022; Li et al., 2022a), masked language modelling (Li et al., 2021; Yang et al., 2022; Ji et al., 2023a; Park & Han, 2023), masked image modelling (Ji et al., 2023a), image-text matching (Zeng et al., 2021; Li et al., 2021; Yang et al., 2022; Park & Han, 2023) and bounding box prediction losses (Zeng et al., 2021) (with bounding boxes obtained from human-annotations (Krishna et al., 2017; Kuznetsova et al., 2020; Shao et al., 2019)).

For more related works, including a discussion on the differences between SPARC and sparse attention (Child et al., 2019; Zaheer et al., 2020) see Appendix B.

4. Experiments

The use of custom datasets (Yao et al., 2021) and pretrained language and/or vision models (Huang et al., 2021; Wang et al., 2022; Mukhoti et al., 2023) makes it difficult to discern the benefit of individual fine-grained losses on learn-

ing more detailed representations. In this work we provide a like-for-like comparison to understand the impact of SPARC and competing methods on downstream performance. We reimplement the following competing baselines: CLIP (Radford et al., 2021), FILIP (Yao et al., 2021), PACL (Mukhoti et al., 2023), MGCA (Wang et al., 2022) and GLoRIA (Huang et al., 2021), and use the same pre-training datasets, architecture, training setup and random initialization of the networks for all objectives. We thoroughly evaluate the methods across a broad range of tasks and datasets, from coarse-grained image-level tasks like classification and retrieval to fine-grained tasks like object detection and semantic segmentation. Unlike some competing methods that improve fine-grained understanding at the cost of decreasing coarse-grained task performance, SPARC simultaneously boosts performance over both coarse- and fine-grained tasks across a number of different benchmarks.

4.1. Experimental Setup

Model architectures. We use Vision Transformers (ViTs) (Dosovitskiy et al., 2020) as image encoders and Transformers (Vaswani et al., 2017) as text encoders. We experiment with ViT-B/32, ViT-B/16 and ViT-L/14 and pair them with corresponding language models.

Datasets. We train on large-scale datasets ALIGN (Jia et al., 2021), JFT (Sun et al., 2017; Zhai et al., 2022) and LTIP (Long Text & Image Pairs) (Alayrac et al., 2022). ALIGN has 1.8 billion images-noisy alt-text pairs, JFT has 4 billion images semi-automatically annotated with a class-hierarchy of 30k labels, while LTIP has 312 million higher-quality images - text pairs with richer image captions.

Pre-training details. We resize images to 224×224 and tokenize the text with a 32k vocabulary sentencepiece tokenizer (Kudo & Richardson, 2018) while keeping a maximum number of 55 tokens for each caption. We train all models using the AdamW (Loshchilov & Hutter, 2017) optimizer, a cosine learning rate schedule with linear warm-up and weight decay regularization. We use a batch size of 16348 and train ViT-B models for 200k steps ($\approx$ 3.2 billion data points) and ViT-L models for 250k steps ($\approx$ 4.1 billion data points). See Appendix D for more details.

4.2. Zero-shot Image Classification

We first evaluate SPARC on the coarse-grained task of zero-shot image classification on ImageNet (Russakovsky et al., 2015), ImageNetV2 (Recht et al., 2019), ImageNet-R (Hendrycks et al., 2021), ImageNet-C (Hendrycks & Dietterich, 2019), ImageNet-A (Hendrycks et al., 2019) and ImageNet-Sketch (Wang et al., 2019) which test specific capabilities like robustness to perturbations and various distribution shifts. We follow a similar evaluation proto-

	Objective	IN	IN-V2 Th	IN-V2 MF	IN-V2 TI	IN-R	IN-C	IN-A	IN-Sketch	Flickr30k		MSCOCO	
										i2t	t2i	i2t	t2i
ViT-B/32	CLIP	69.0	68.8	60.4	73.4	62.4	44.6	15.8	52.4	79.2	66.5	53.5	38.4
	FILIP	56.8	54.8	48.4	60.0	44.6	30.8	7.8	39.6	62.6	50.5	35.6	26.2
	PACL	61.2	59.5	51.9	65.2	52.9	36.4	9.3	45.2	65.5	49.8	37.6	26.5
	GloRIA	65.9	64.8	57.0	69.6	57.4	40.7	11.7	48.7	74.6	61.5	46.9	34.5
	MGCA	68.6	67.4	59.2	72.6	61.0	43.5	14.1	50.9	81.5	64.6	54.5	37.7
	SPARC (ours)	**70.4**	**69.6**	**62.1**	**74.5**	**63.2**	**46.5**	**17.3**	**52.7**	**82.5**	**67.7**	**55.0**	**39.7**
ViT-B/16	CLIP	73.9	73.6	66.1	77.1	68.8	50.4	32.5	57.3	84.0	71.6	56.2	42.4
	FILIP	61.4	61.0	53.8	65.6	53.2	35.9	14.2	45.1	69.0	55.8	40.2	29.5
	PACL	63.3	61.7	54.4	66.8	54.1	37.3	12.9	45.4	69.6	54.9	41.8	29.1
	GloRIA	70.4	70.0	62.8	74.7	65.7	46.4	25.0	54.8	78.0	68.4	49.7	38.9
	MGCA	72.7	72.7	65.3	76.3	67.6	48.4	29.8	55.5	82.2	67.7	**57.6**	39.8
	SPARC (ours)	**74.7**	**74.0**	**67.1**	**77.8**	**71.1**	**51.31**	**34.2**	**57.9**	**84.4**	**72.0**	**57.6**	**43.0**
ViT-L/4	CLIP	79.2	78.5	71.8	81.6	78.5	**61.3**	51.5	65.1	84.7	73.7	58.6	44.8
	MGCA	78.0	77.4	70.5	80.6	75.2	57.9	45.5	63.1	85.9	73.2	**59.7**	44.3
	SPARC (ours)	**79.7**	**78.9**	**72.6**	**81.9**	**79.8**	**61.3**	**53.4**	**65.9**	**86.9**	**74.4**	58.9	**45.6**

Table 1: Top-1 accuracy (in %) of zero-shot classification using prompt ensembling on ImageNet (IN) and its variants ImageNet-V2 Threshold (IN-V2 Th), ImageNet-V2 Matched Frequency (In-V2 MF), ImageNet-V2 Top Images (IN-V2 TI), ImageNet-R (IN-R), ImageNet-C (IN-C), ImageNet-Sketch (IN-Sketch) and image-to-text (i2t) and text-to-image (t2i) retrieval on Flickr30k and MSCOCO as measured by Recall at 1.

col to Radford et al. (2021) and use prompt ensembling; see Appendix D for details. From Table 1 we see that SPARC outperforms or matches competing methods in all settings and across different ViT architectures. Specifically, SPARC shows very effective information encoding from larger patches as exhibited by the significant improvements over baselines for ViT B/32. We also evaluate SPARC and competing methods when using only one prompt (instead of prompt ensembling) and observe that while performance for all methods goes slightly down (as in line with the literature) the performance of SPARC decreases less than that of competing methods; see Table 8 in Appendix D for details.

From Table 1, we see that in the pretraining setting PACL (Mukhoti et al., 2023) and GLoRIA (Huang et al., 2021) underperform CLIP, whereas MGCA (Wang et al., 2022) shows more competitive performance to CLIP. While all three methods were developed with the use of pretrained models in mind (and here they are tested in a pretraining from scratch setting), PACL and GLoRIA (as per their original papers) more heavily rely on pretrained components than MGCA which is also reflected in their performance in Table 1; PACL relies most on pretrained components (uses frozen pretrained CLIP vision and text encoders and trains additional networks on top to get better fine-grained understanding) shows the biggest performance degradation compared to CLIP when training from scratch. On the other hand, FILIP (Yao et al., 2021), which was developed as a fine-grained objective for pretraining from scratch, has

proven highly unstable to train across a wide range of learning rates and weight decay parameters which lead to poor performance. This training difficulty has also been noted in the original paper (Yao et al., 2021) (cf. in the Appendix A.3. *"...training is extremely unstable and the Nan loss easily happens."*). Moreover, FILIP uses a number of additional tricks not present in a standard pretraining setup like image augmentations, backtranslation of captions and custom prompt ensembling as per (Yao et al., 2021). Alternatively, empirically, for SPARC, we did not observe any training instabilities caused by thresholding the similarity matrix during training across many different hyperparameter configurations.

4.3. Image-text Retrieval

Next we evaluate SPARC on zero-shot cross-modal retrieval tasks on Flickr30k (Plummer et al., 2015) and MSCOCO (Lin et al., 2014). From Table 1, we see that SPARC outperforms all competing baselines. Similar to classification performance, fine-grained losses PACL and GLoRIA significantly underperform CLIP, while MGCA shows competitive performance to CLIP in the pretraining setting. Unfortunately, FILIP (Yao et al., 2021) again underperforms CLIP. In an attempt to stabilize FILIP we combined it with CLIP and observed improvement on image-to-text Flikr30k on ViT B/32 while being competitive on other benchmarks to CLIP; see Appendix D (Tables 9, 10 and 11) for these results and Recall at 5 and 10 for retrieval.

4.4. Evaluating Faithfulness

Next we test fine-grained performance of SPARC through *faithfulness*—how consistent the model's highest scoring caption is with the ground truth caption(s) (Ji et al., 2023b). This is different from top-1 retrieval (R@1) which measures exact match retrieval and does not evaluate the model's ability to faithfully describe the elements in the image. Faithfulness has been used in the LLM literature to assess the propensity of the model to hallucinate (Adlakha et al., 2023; Razumovskaia et al., 2023) as models with higher faithfulness more accurately capture the details of the ground truth while not inserting additional information (possible hallucinations). The lexical overlap metric of $\mathcal{K}$-Precision measuring the proportion of tokens in the top chosen caption that appear in the ground truth tokens has been shown to correlate well with human judgement (Adlakha et al., 2023). In Table 2 we report the $\mathcal{K}$-Precision on MSCOCO for all tokens ($\mathcal{K}$-P), and $\mathcal{K}$-Precision restricted to nouns and adjectives only ($\mathcal{K}$-P_{na}), as these better encode objects present in the image. We see that SPARC reduced hallucinations of objects (higher $\mathcal{K}$-P_{na}) and shows competitive performance to related methods when taking all tokens into account.

Method	ViT-B/32		ViT-B/16	
	$\mathcal{K}$-P_{na}	$\mathcal{K}$-P	$\mathcal{K}$-P_{na}	$\mathcal{K}$-P
CLIP	76.03	77.82	77.56	78.99
FILIP	63.3	66.83	66.05	70.09
PACL	3.36	26.26	4.09	27.31
GLoRIA	71.63	73.54	73.85	75.3
MGCA	75.79	77.98	77.66	**80.03**
SPARC (ours)	**76.46**	**78.44**	**78.72**	79.77

Table 2: All-token $\mathcal{K}$-Precision ($\mathcal{K}$-P) and the $\mathcal{K}$-Precision on nouns and adjectives ($\mathcal{K}$-P_{na}) (in %) on MSCOCO.

4.5. Fine-grained Localization

Open-vocabulary object detection. To evaluate whether the improved fine-grained understanding learned with SPARC translates to tasks requiring fine-grained localization, we use SPARC as a backbone for object detection. We use the OWL-ViT open-vocabulary object detector (Minderer et al., 2022) with a ViT-B/16 backbone. After SPARC pre-training, detection heads are added to the backbone and fine-tuned on Objects365 (Shao et al., 2019) and Visual Genome (Krishna et al., 2017) datasets following the approach in Minderer et al. (2022). We evaluate the resulting model on the large-vocabulary dataset LVIS (Gupta et al., 2019) which is well-suited for testing the transfer of knowledge from image-level pretraining. LVIS contains 1203 categories of objects, of which 307 "rare" categories are excluded from the training data to measure zero-shot

transfer from pretraining. Moreover, we also evaluate detection on the 80 MSCOCO classes. We run detection training three times and report mean and standard deviation in Table 3. SPARC improves over CLIP +0.9% on LVIS and MSCOCO as measured by mean average precision and +3.1% on LVIS "rare" classes. Since LVIS "rare" classes are never seen during detection training, the model has to rely on information transfer from the pretrained representations for these classes. The large improvement of SPARC over the baseline on LVIS AP_{rare} suggests that SPARC has learned more informative fine-grained representations.

Method	LVIS		MSCOCO
	AP_{all}	AP_{rare}	AP_{all}
CLIP	26.9 ± 0.12	22.0 ± 0.79	38.5 ± 0.19
SPARC (ours)	$\mathbf{27.9 \pm 0.11}$	$\mathbf{25.1 \pm 0.95}$	$\mathbf{39.4 \pm 0.13}$

Table 3: Mean Average precision (as mean $\pm$ std. deviation) on all and rare LVIS classes and on all MSCOCO classes.

Semantic Segmentation. Following Mukhoti et al. (2023), we also perform zero-shot segmentation *given* a text label, i.e., we compute image patch embeddings and calculate the cosine similarity of the patch embedding with the text embeddings of all the ground-truth classes. We assign a class for each patch as the text that corresponds to the maximum cosine similarity of that patch. We then upsample the patches to match the resolution of the ground-truth segmentation and calculate for each class the Intersection over Union (IoU) between the predicted and ground-truth segmentations; we report the mean IoU scores over classes present in the ground-truth image. More details about this evaluation can found in Appendix D.

Method	CLIP	FILIP	PACL	GLoRIA	MGCA	SPARC
VOC	23.02	19.32	1.23	22.64	21.91	**27.36**
Context	20.45	9.31	1.61	15.26	11.50	**21.65**

Table 4: Semantic Segmentation: mIoU of predicted and ground-truth segmentation on Pascal VOC and PASCAL Context datasets.

From Table 4, we see that SPARC strongly improves over other baselines, significantly surpassing the next best model by +4.34 mIoU on PASCAL VOC (Everingham et al., 2015) and by +1.2 mIoU on PASCAL Context (Mottaghi et al., 2014). We visualize the predicted segmentation masks on the PASCAL VOC dataset in Figure 8. Whereas CLIP predicts the object in many different parts of the image, SPARC achieves better object localization and predicts object shapes more accurately.
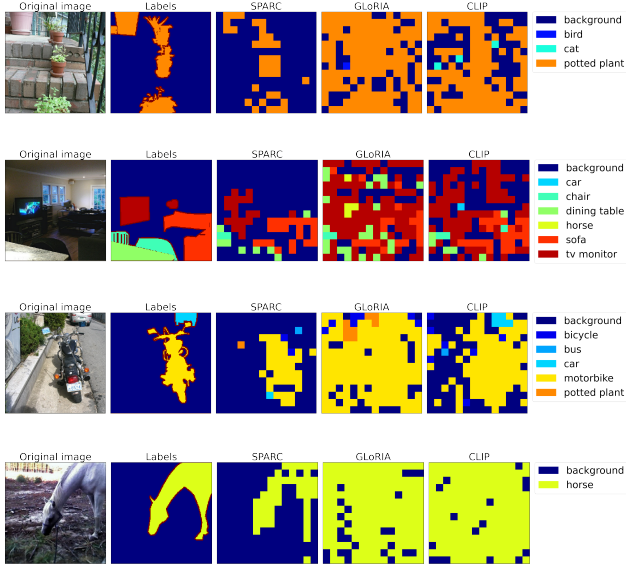
Figure 3: Qualitative results for zero-shot segmentation on Pascal VOC dataset. We illustrate the original image, pixel-level ground-truth labels and the the patch-level segmentation masks obtained from SPARC, GLoRIA and CLIP.

4.6. SPARC Backbones in Vision Language Models

Vision backbones trained contrastively from image-text paired data are often used in foundational vision-language models (VLMs) such as Flamingo (Alayrac et al., 2022) as frozen encoders. We perform experiments where we compare using a CLIP backbone vs. a SPARC backbone in a Flamingo-style architecture. For this, we freeze the ViT-B/16 vision models trained with CLIP and SPARC and pair them with a frozen 400M parameter (pre-trained) language model. On top of the frozen vision and language backbones, we train Perceiver Resampler cross-attention layers (Alayrac et al., 2022) to produce free-form text as output. More details about the training set-up can be found in Appendix D. We evaluate the models on captioning tasks on MSCOCO and Flickr30k. From Table 5 we see that SPARC outperforms CLIP on both datasets.

Method	MSCOCO	Flickr30k
CLIP	24.3	12.9
SPARC (ours)	**25.3**	**13.6**

Table 5: CIDEr score evaluating captioning performance of different vision backbones in a Flamingo-style (Alayrac et al., 2022) model.

4.7. Analysis

Ablations. To assess the benefits of the different components of SPARC, we perform the following two ablations: 1) remove the sparsity on the similarity matrix (set $\sigma = 0$ in Eq. 3) and 2) replace the simple normalization when

computing the alignment weights in Eq. 4 by a softmax. From the results in Table 6 on both fine-grained (MSCOCO retrieval) and coarse-grained (ImageNet zero-shot classification) tasks we notice that both components play a significant role in the model's performance with using softmax yielding the highest decrease in performance. See Appendix A for a detailed discussion of the problems with using softmax to compute the alignment weights.

	MSCOCO (i2t)		MSCOCO (t2i)		ImageNet
	R@1	R@5	R@1	R@5	Top-1 acc.
SPARC	**57.6**	**81.2**	**43.0**	**68.6**	**72.6**
- no sparsity	56.1	80.7	42.4	68.2	72.1
- softmax	55.2	79.8	41.6	67.5	70.6

Table 6: Ablations for the ViT-B/16 SPARC model on the MSCOCO image-to-text (i2t) and text-to-image (t2i) retrieval and zero-shot classification on ImageNet.

Many-to-one mapping. To empirically verify that SPARC resolves the one-to-one mapping caused by softmax-based alignment scores, we compare statistics of the attention weights from SPARC and GLoRIA in the ViT-B/16 models. GLoRIA computes attention weights by applying softmax on the similarity matrix between the patch and token embeddings, while SPARC computes these attention weights by sparsifying the similarity matrix and normalizing it. Both methods use the attention weights to compute language-aware vision embeddings needed for the local losses. On the retrieval evaluation datasets (Flickr30k and MSCOCO) we first pass the image-text pairs through the unimodal encoders and compute the corresponding attention weights for SPARC and GLoRIA. Then, we compute the following statistic for the attention weights for each text token in the dataset: $(h_{j1} - h_{jk})/h_{j1} \cdot 100$, where h_{jk} is the k-th highest attention weight for token j. This represents the relative difference between the highest and k-th highest attention weight (in %) for a given text token. If the method tends to a one-to-one mapping between text and image tokens this relative difference should be high, while for methods that induce a many-to-one mapping this relative difference should be lower as the values of the top highest attention weights should be much closer to each other than in a one-to-one-mapping setting.

	Flickr30k			MSCOCO		
	$k=2$	$k=3$	$k=4$	$k=2$	$k=3$	$k=3$
GLoRIA	26.4%	39.2%	48.5%	26.0%	39.2%	47.8%
SPARC	7.3%	16.4%	28.3%	7.2%	15.7%	26.8%

Table 7: Average relative difference between the highest and k-th highest attention weight for each text token.

As can be seen from the Table 7, for GLoRIA (attention

weights computed through softmax) the second highest attention weight is markedly smaller than the highest weight (sharp decrease of 26.0% on MSCOCO and 26.4% on Flickr30k (average across text tokens)) which indicates a "peakier" distribution compared to the attention weight distribution in SPARC where the relative difference between the highest and second highest is just 7.2% on MSCOCO and 7.3% on Flickr30k indicating a less "peaky" distribution corresponding to more of a many-to-one relationship.

Compute and memory consumption. To understand the computational and memory requirements of different methods, we measure the compute and peak memory usage for one update step for different batch size when training on 256 TPUs v3. In Figure 4 we plot compute (in TFLOPS) for different methods when varying the batch size (B) from 2048 to 16384; for peak memory we observe the same relative ordering of methods with GLoRIA (Huang et al., 2021) being as memory intensive at batch size 4096 as the other methods (e.g. CLIP and SPARC) at batch size 16384. For FILIP, compute increases by more than 200% between B=8196 and B=16384, as opposed to the 100% increase for CLIP, SPARC and MGCA, while GLoRIA (Huang et al., 2021) is over 5 times as compute intensive as CLIP, SPARC and MGCA even at batch size 4096 (thus preventing us from training at higher batch sizes). On the other hand, note that CLIP, SPARC and MGCA use the same order of magnitude of compute and memory with SPARC adding only $+0.55\%$ additional compute and peak memory at $B = 16384$ over CLIP while learning better fine-grained representations (compared to MGCA which adds $+1\%$ additional compute and peak memory). See Appendix D.6 for detailed numbers of compute and peak memory.
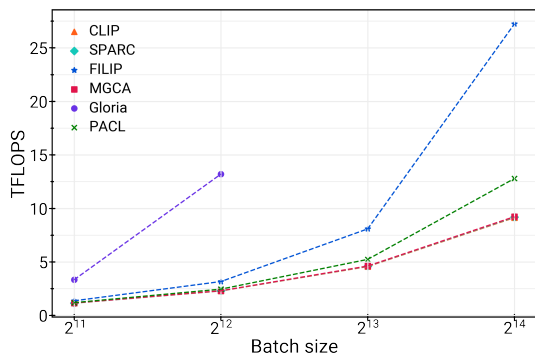

Figure 4: Compute (in TFLOPS) used by all methods.

5. Conclusion

In this work we proposed a novel method SPARse Fine-grained Contrastive Alignment (SPARC) for fine-grained vision-language pretraining. SPARC simultaneously learns information at different levels of granularity by contrasting

both global and localized embeddings. SPARC learns to group patches based on similarity to tokens and contrast the resulting language-grounded vision embeddings with token embeddings. Unlike previous work this comparison is done within individual image-text pairs and does not require the computationally and memory expensive comparison of all patches and tokens within the full batch. Through extensive experimental evaluation we show that SPARC improves performance both on image-level tasks like classification and retrieval and more fine-grained tasks like object detection and segmentation which require localization. Moreover, SPARC improves model faithfulness and captioning in foundation vision-language models. Exploring different approaches to sparsification and learning patch groupings, using more descriptive captions or additional signals like bounding boxes and segmentation masks represents a promising line of future work as it should lead to even more informative representations.

Acknowledgements

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Adlakha, V., BehnamGhader, P., Lu, X. H., Meade, N., and Reddy, S. Evaluating correctness and faithfulness of instruction-following models for question answering. *arXiv preprint arXiv:2307.16877*, 2023.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

Alsentzer, E., Murphy, J. R., Boag, W., Weng, W.-H., Jin, D., Naumann, T., and McDermott, M. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*, 2019.

Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.

Child, R., Gray, S., Radford, A., and Sutskever, I. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Dawidowicz, G., Hirsch, E., and Tal, A. Limitr: Leveraging local information for medical image-text representation. *ArXiv*, abs/2303.11755, 2023. URL https://api.semanticscholar.org/CorpusID:257636659.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Elfadel, I. M. and Wyatt Jr, J. L. The" softmax" nonlinearity: Derivation using statistical mechanics and useful properties as a multiterminal analog circuit element. *Advances in neural information processing systems*, 6, 1993.

Everingham, M., Eslami, S. M. A., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, January 2015.

Geng, S., Yuan, J., Tian, Y., Chen, Y., and Zhang, Y. Hiclip: Contrastive language-image pretraining with hierarchy-aware attention. *arXiv preprint arXiv:2303.02995*, 2023.

Gupta, A., Dollar, P., and Girshick, R. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5356–5364, 2019.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Hendrycks, D., Zhao, K., Basart, S., Steinhardt, J., and Song, D. Natural adversarial examples.(2019). *arXiv preprint cs.LG/1907.07174*, 5(6), 2019.

Hendrycks, D., Basart, S., Mu, N., Kadavath, S., Wang, F., Dorundo, E., Desai, R., Zhu, T., Parajuli, S., Guo, M., et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8340–8349, 2021.

Hoffmann, D. T., Schrodi, S., Behrmann, N., Fischer, V., and Brox, T. Eureka-moments in transformers: Multi-step tasks reveal softmax induced optimization problems. *arXiv preprint arXiv:2310.12956*, 2023.

Huang, S.-C., Shen, L., Lungren, M. P., and Yeung, S. Gloria: A multimodal global-local representation learning framework for label-efficient medical image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3942–3951, 2021.

Ji, Y., Tu, R., Jiang, J., Kong, W., Cai, C., Zhao, W., Wang, H., Yang, Y., and Liu, W. Seeing what you miss: Vision-language pre-training with semantic completion learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6789–6798, 2023a.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023b.

Jia, C., Yang, Y., Xia, Y., Chen, Y.-T., Parekh, Z., Pham, H., Le, Q., Sung, Y.-H., Li, Z., and Duerig, T. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pp. 4904–4916. PMLR, 2021.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123:32–73, 2017.

Krojer, B., Adlakha, V., Vineet, V., Goyal, Y., Ponti, E., and Reddy, S. Image retrieval from contextual descriptions. *arXiv preprint arXiv:2203.15867*, 2022.

Kudo, T. and Richardson, J. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*, 2018.

Kuznetsova, A., Rom, H., Alldrin, N., Uijlings, J., Krasin, I., Pont-Tuset, J., Kamali, S., Popov, S., Malloci, M., Kolesnikov, A., et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, 2020.

Li, J., Selvaraju, R., Gotmare, A., Joty, S., Xiong, C., and Hoi, S. C. H. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34: 9694–9705, 2021.

Li, J., Li, D., Xiong, C., and Hoi, S. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pp. 12888–12900. PMLR, 2022a.

Li, L. H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.-N., et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10965–10975, 2022b.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Minderer, M., Gritsenko, A., Stone, A., Neumann, M., Weissenborn, D., Dosovitskiy, A., Mahendran, A., Arnab, A., Dehghani, M., Shen, Z., et al. Simple open-vocabulary object detection. In *European Conference on Computer Vision*, pp. 728–755. Springer, 2022.

Mottaghi, R., Chen, X., Liu, X., Cho, N.-G., Lee, S.-W., Fidler, S., Urtasun, R., and Yuille, A. The role of context for object detection and semantic segmentation in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

Mukhoti, J., Lin, T.-Y., Poursaeed, O., Wang, R., Shah, A., Torr, P. H., and Lim, S.-N. Open vocabulary semantic segmentation with patch aligned contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19413–19423, 2023.

Paiss, R., Ephrat, A., Tov, O., Zada, S., Mosseri, I., Irani, M., and Dekel, T. Teaching clip to count to ten. *arXiv preprint arXiv:2302.12066*, 2023.

Parcalabescu, L., Cafagna, M., Muradjan, L., Frank, A., Calixto, I., and Gatt, A. Valse: A task-independent benchmark for vision and language models centered on linguistic phenomena. *arXiv preprint arXiv:2112.07566*, 2021.

Park, J. and Han, B. Multi-modal representation learning with text-driven soft masks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2798–2807, 2023.

Peterson, C. and Söderberg, B. A new method for mapping optimization problems onto neural networks. *International Journal of Neural Systems*, 01(01):3–22, 1989.

Plummer, B. A., Wang, L., Cervantes, C. M., Caicedo, J. C., Hockenmaier, J., and Lazebnik, S. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE international conference on computer vision*, pp. 2641–2649, 2015.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Ranasinghe, K., McKinzie, B., Ravi, S., Yang, Y., Toshev, A., and Shlens, J. Perceptual grouping in vision-language models. *arXiv preprint arXiv:2210.09996*, 2022.

Ranasinghe, K., McKinzie, B., Ravi, S., Yang, Y., Toshev, A., and Shlens, J. Perceptual grouping in contrastive vision-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5571–5584, 2023.

Razumovskaia, E., Vulić, I., Marković, P., Cichy, T., Zheng, Q., Wen, T.-H., and Budzianowski, P. *Dial BeInfo for Faithfulness*: Improving factuality of information-seeking dialogue via behavioural fine-tuning, 2023.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do imagenet classifiers generalize to imagenet? In *International conference on machine learning*, pp. 5389–5400. PMLR, 2019.

Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115: 211–252, 2015.

Shao, S., Li, Z., Zhang, T., Peng, C., Yu, G., Zhang, X., Li, J., and Sun, J. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8430–8439, 2019.

Shen, K., Guo, J., Tan, X., Tang, S., Wang, R., and Bian, J. A study on relu and softmax in transformer. *arXiv preprint arXiv:2302.06461*, 2023.

Sun, C., Shrivastava, A., Singh, S., and Gupta, A. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pp. 843–852, 2017.

Varma, M., Delbrouck, J.-B., Hooper, S., Chaudhari, A., and Langlotz, C. Villa: Fine-grained vision-language representation learning from real-world data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 22225–22235, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Wang, F., Zhou, Y., Wang, S., Vardhanabhuti, V., and Yu, L. Multi-granularity cross-modal alignment for generalized medical visual representation learning. *Advances in Neural Information Processing Systems*, 35:33536–33549, 2022.

Wang, H., Ge, S., Lipton, Z., and Xing, E. P. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pp. 10506–10518, 2019.

Xu, J., De Mello, S., Liu, S., Byeon, W., Breuel, T., Kautz, J., and Wang, X. Groupvit: Semantic segmentation emerges from text supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18134–18144, 2022.

Xu, J., Hou, J., Zhang, Y., Feng, R., Wang, Y., Qiao, Y., and Xie, W. Learning open-vocabulary semantic segmentation models from natural language supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2935–2944, 2023.

Yang, J., Duan, J., Tran, S., Xu, Y., Chanda, S., Chen, L., Zeng, B., Chilimbi, T., and Huang, J. Vision-language pre-training with triple contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15671–15680, 2022.

Yao, L., Huang, R., Hou, L., Lu, G., Niu, M., Xu, H., Liang, X., Li, Z., Jiang, X., and Xu, C. Filip: Fine-grained interactive language-image pre-training. *arXiv preprint arXiv:2111.07783*, 2021.

Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., and Wu, Y. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

Yuksekgonul, M., Bianchi, F., Kalluri, P., Jurafsky, D., and Zou, J. When and why vision-language models behave like bag-of-words models, and what to do about it? *arXiv preprint arXiv:2210.01936*, 2022.

Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297, 2020.

Zeng, Y., Zhang, X., and Li, H. Multi-grained vision language pre-training: Aligning texts with visual concepts. *arXiv preprint arXiv:2111.08276*, 2021.

Zhai, S., Likhomanenko, T., Littwin, E., Busbridge, D., Ramapuram, J., Zhang, Y., Gu, J., and Susskind, J. Stabilizing transformer training by preventing attention entropy collapse. *ICML*, 2023.

Zhai, X., Kolesnikov, A., Houlsby, N., and Beyer, L. Scaling vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12104–12113, 2022.

Zhong, Y., Yang, J., Zhang, P., Li, C., Codella, N., Li, L. H., Zhou, L., Dai, X., Yuan, L., Li, Y., et al. Regionclip: Region-based language-image pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16793–16803, 2022.

A. Problems with using softmax for obtaining alignment weights

$Softmax$ is ubiquitously used to normalise activations that should or could be interpreted as probabilities, as it is for example the case of attention/alignmnet weights. One potential reason behind this choice is the dominating practice of using $softmax$ as the output activation function for classification tasks, being the canonical link function for multinomial outputs. Another appealing property is that it acts as a differentiable max-operator, allowing for a natural interpretation of *selecting* one class out of multiple.

However, $softmax$ can be problematic from a gradient flow perspective (Shen et al., 2023; Zhai et al., 2023; Hoffmann et al., 2023), and in this section we will expand this observation and the implications it might have on our specific use case. Also, intuitively from its role as a soften max operator, softmax prefers to converge to peaky uni-modal distribution, selecting one out of k, and is less likely to represent multi-modal distributions. This is due to how gradients flow through the activation, leading to winner-takes-all dynamics (Elfadel & Wyatt Jr, 1993; Peterson & Söderberg, 1989) that ensures the peakyness and unimodality of the distribution represented.

If we assume $a(\mathbf{h}) = softmax(\mathbf{h})$[1], for some $\mathbf{h} \in \mathcal{R}^k$, then we can write the derivative as

$$\frac{\partial \mathbf{a}_i}{\partial \mathbf{h}_j} = \begin{cases} \mathbf{a}_i - \mathbf{a}_i^2 & \text{iff } i = j \\ -\mathbf{a}_i \mathbf{a}_j & \text{otherwise} \end{cases} \tag{8}$$

Assume we have some loss L which is a function of $\sum_i \mathbf{a}_i \mathbf{V}_i$, i.e. some values $\mathbf{V}_i \in \mathcal{R}^n$ that have been summarised using attention weights $\mathbf{a}_i$.

Softmax gradients vanish at initialisation.　Assume we have a large number of patches or tokens we want to attend over. In our notation, $k \gg 0$. At initialisation, all preactivation entries $\mathbf{h}_i$ will be small numbers of similar magnitude. The attention weights will be uniformally distributed over the k patches, leading to $\mathbf{a}_i \approx \frac{1}{k} \ll 1, \forall i$. Due to the weights being almost uniformally distributed, different observation will lead to randomly selecting a different patch. Therefore in expectation the gradient through the softmax on a particular token i will be scaled by $\frac{1}{k^2}$ which will vanish very fast to 0 as k grows. Note that in the rare scenario that the system picks the i-th element, the gradient becomes $\frac{1}{k}$ which also vanishes to 0 as k grows. If we consider a very large k, this ensures that we have a plateau at initialization that might be hard to escape (or might take many updates to do so). See also (Hoffmann et al., 2023) for a similar observation.

Softmax exhibits *winner-takes-all* dynamics.　This has been understood and seen as a desirable property early on, see for example (Peterson & Söderberg, 1989) and (Elfadel & Wyatt Jr, 1993). One way to intuitively justify this behaviour is to think of the effect of applying the softmax operation multiple time (i.e. study the dynamics of a system whose transition function is just softmax). As shown in (Peterson & Söderberg, 1989) Fig. 5, the corners of the simplex act as attractors of this dynamical system, where from any initial condition, the system very quickly converges to one of the corners. This is caused by the dynamics of the gradients. When a particular weight is pushed up, all other weights are pushed down due to the normalisation. The amount by which the weight is pushed depends on its magnitude. So if a particular weight is larger and correlates positively with the desired behaviour, it will be pushed up proportionally more than other weights that correlate positively. Note that the particular form of the function (including the exponentiation) play a role in the form the gradients take, and removing the exponentiation will change the behaviour. These types of dynamics, have the downside of leading the distribution induced by the softmax to be unimodal. That is, softmax will act, as the name of the activation indicates, as a *max* operator, preferring to learn a behaviour where it picks one out of k, rather than multiple equally relevant candidates.

Softmax saturates proportional to its certainty　Assume $\exists i$ such that $\forall j, j \neq i$ we have $a_i \gg a_j$. This implies that $1 - a_i \to 0$ and $a_j < 1 - a_i$. The gradient for the i-th position, according to equation 8, will be $a_i(1 - a_i)$ and will go to zero as linearly as a_i approaches 1. The gradient for any other position j, will go to 0 at the same rate, as it will be roughly a_j which is bounded from above from $1 - a_i$. Note that a step of size Δ on h, due to the exponentiation and normalization of softmax, will make $a_i \to 1$ exponentially fast for constant change in h.

---

[1]By abuse of notation, we will use $\mathbf{a} \in \mathcal{R}^k$, where $\mathbf{a} = a(\mathbf{h})$ and use $\mathbf{a}_i$ for the i-th dimension of vector $\mathbf{a}$

B. Additional related works

Fine-grained understanding in VLMs.   We further expand here the discussion on achieving fine-grained understanding in vision-language models (VLMs) through additional losses and modules. In addition to the approaches described in Section 3, another line of work involves proposes modifying the underlying vision transformer architecture to build modules that lead to a hierarchical grouping of image regions: e.g. GroupViT (Xu et al., 2022), OVSegmentor (Xu et al., 2023), HiCLIP (Geng et al., 2023). While these methods propose architectural changes, the objective used for training still involves having a global contrastive loss. Conversely, in our work, we use the standard vision transformer architecture and propose instead changes to the training objective to achieve finegrained understanding.

Moreover, note that several of these approaches (Xu et al., 2023) and the other methods who add a cross-modal encoder on top of the dual image-text encoder (Li et al., 2021; Yang et al., 2022) with captioning/masked language modelling losses start training from pre-trained text encoders and/or vision encoder.

Similarly, (Ranasinghe et al., 2023) improve the semantic and spatial information in dual encoders trained contrastively by changing the patch embeddings aggregation methods from average pooling to max pooling and by starting training with both pre-trained vision and language encoders. In our work, we focus specifically on the set-up of training the dual encoders from scratch.

Sparse similarity.   Another thread of related works involve using sparse attention to reduce the compute and memory cost of attention mechanisms in transformers (Child et al., 2019; Zaheer et al., 2020). In particular, with sparse attention we do not compute the similarity scores between all pairs of tokens, but rather compute only a (small) subset of all the similarity scores. This approach is commonly used in language modeling where it brings significant compute and memory savings when processing longer text sequences as it avoids the quadratic complexity of computing the full attention mechanism. Alternatively, with SPARC we compute the full similarity matrix between all text embeddings and all image embeddings for each image-text pair. As a next step, we sparsify the full similarity matrix in order to learn an alignment between text tokens and parts of the image that visually encode that text. We then normalize the sparsified similarity matrix to compute the alignment weights between each text token and all image patches from which we compute the resulting language-grouped vision embeddings. Therefore, the cost of computing the sparse similarity matrix needed for SPARC is small, as also highlighted by the Compute and memory consumption analysis in Section 4.7.

C. SPARC pseudo-code

Listing 1 provides JaX-alike pseudo-code for the SPARC objective detailing the construction of both the global and local losses.

```
1   # Models:
2   #    vision_encoder
3   #    language_encoder
4   # Inputs:
5   #    image - [B, H, W, C]
6   #    text - [B, N]
7   # Hyperparameters:
8   #    similarity_threshold
9   #    global_loss_weight
10  #    local_loss_weight
11  #    inverse_temperature
12
13  def pairwise_contrastive_loss(a, b, labels):
14      labels = eye(a.shape[0])
15      logits_ab = dot(a * b.T) * inverse_temperature
16      return softmax_cross_entropy(logits=logits_ab, labels=labels, reduction='mean')
17
18  def masked_pairwise_contrastive_loss(a, b, mask):
19      batch_size, seq_len, _ = a.shape[0]
20      mask_logits = einshape('bnm->(bn)m', 1.0 - mask, n=seq_len)
21      labels = einshape('ns->(bn)s', eye(a.shape[1]), b=batch_size)
22      logits = einsum('bmd,bnd->bmn', a, b) * inverse_temperature
23      logits = einshape('bnm->(bn)m', logits)
24      loss = softmax_cross_entropy(logits=logits - mask_logits * INF, labels=labels)
25      loss = sum(loss * mask) / sum(mask)
26      return loss
27
28  # ---------- GLOBAL LOSS ----------
29
30  # encoders include adapters
31  v_patch_embed = vision_encoder(image)
32  l_token_embed, language_mask = language_encoder(text)
33
34  v_embed = l2_normalize(mean(v_patch_embed, axis=1), axis=-1)
35  l_embed = l2_normalize(mean(l_token_embed, axis=1), axis=-1)
36
37  loss_vl = pairwise_contrastive_loss(v_embed, l_embed)
38  loss_lv = pairwise_contrastive_loss(l_embed, v_embed)
39
40  global_loss = 0.5 * (loss_vl + loss_lv)                                           # (eq 1)
41
42  # ---------- LOCAL LOSS ----------
43
44  # similarity calculation
45  similarity = einsum('btd,bpd->btp', l_token_embed, v_patch_embed)
46
47  # min-max normalisation
48  similarity = (similarity - min(similarity, axis=-1)) /
49               (max(similarity, axis=-1) - min(similarity, axis=-1))               # (eq 2)
50
51  # thresholding
52  similarity = where(similarity < similarity_threshold, 0.0, similarity)           # (eq 3)
53
54  # alignment-weighting
55  v_align_weights = similarity / sum(similarity, axis=-1)                          # (eq 4)
56  l_grouped_v_patch_embed = einsum('btp,bpd->btd', v_align_weights, v_patch_embed) # (eq 5)
57
58  l_grouped_v_patch_embed = l2_normalize(l_grouped_v_patch_embed, axis=-1)
59  l_token_embed = l2_normalize(l_token_embed, axis=-1)
60
61  loss_vl_local = masked_pairwise_contrastive_loss(l_grouped_v_patch_embed, l_token_embed, language_mask)
62  loss_lv_local = masked_pairwise_contrastive_loss(l_token_embed, l_grouped_v_patch_embed, language_mask)
63
64  local_loss = 0.5 * (loss_vl_local + loss_lv_local)                               # (eq 6)
65
66  # ---------- TOTAL (SPARC) LOSS ----------
67
68  loss = global_loss_weight * global_loss + local_loss_weight * local_loss         # (eq 7)
```

Listing 1: Pseudo-code for SPARC.

D. Experiments details

D.1. Model architectures

For the dual-encoder, we use the standard Vision Transformers (ViTs) (Dosovitskiy et al., 2020) as image encoders and Transformers (Vaswani et al., 2017) as text encoders. We perform experiments with ViT-B models with different patch sizes (ViT-B/32 and ViT-B/16) and a ViT-L model with patch size 14 (ViT-L/14). Thus, for the ViT-B image encoder, we use a model with 12 layers, 768 width and 12 attention heads, while for the ViT-L image encoder we use a model with 24 layers, 1024 width and 16 attention heads. For the language encoder, we use an architecture with 12 layers, 768 width and 12 attention heads. The linear adapters $g_v(\cdot)$ and $g_t(\cdot)$ project the vision and language embeddings respectively to a shared embedding space of dimensionality 512.

D.2. Datasets

As described in Section 4, we use the following datasets for pre-training: ALIGN (Jia et al., 2021), JFT (Sun et al., 2017; Zhai et al., 2022) and LTIP (Long Text & Image Pairs) (Alayrac et al., 2022). Note that for JFT, where the images were semi-automatically annotated with a class-hierarchy of 30k labels, we flatten the hierarchical label structure and use all the assigned labels to describe the image. We use a multi-step training strategy where we alternate sampling batches from each of the 3 large datasets; the gradient updates are then performed by aggregating the gradients from computing the loss on one batch from each of the datasets.

D.3. Baselines

Our implementation of baselines follow the publicly available code (where available[2]) with a few minor differences we outline here.

In the original MGCA implementation, token-wise cross-modal alignment (see Eqn. 5 in the original paper) uses the last-layer attention weight from a visual token to the [CLS] token (averaged across multiple heads) to weight the loss terms for different visual tokens (and vice versa for language tokens). In our implementation, since we do not use the [CLS] token but instead use average pooling to get the global language/vision embeddings, we omit this weighting operation.

In the original GLoRIA implementation, language tokens are aggregated for each word to ensure that contrasted language embeddings refer to complete words (see Section 3.2.1 in the original paper); however, to ensure fair comparison, we do not have this additional aggregation operation, and instead use language tokens directly in local losses. Additionally, in our experiments we found that it is crucial to normalize the pairwise vision-language embedding similarities (see Eqn. 3 in the original paper) by $\sqrt{D}$ where D is the embedding size. Without this normalization, we found training with GLoRIA to be unstable. Moreover, recall that GLoRIA requires computing similarities between all token embeddings and all patch embeddings in the batch. This is memory expensive and it was not possible (due to device memory constraints) for batch sizes of 16348. Consequently, we used a batch size of 4096 for Gloria and trained the models for 800k steps (to match the number of examples seen by the other baseline). See discussion in Section D.6 for detailed computation of FLOPs and memory usage of GLoRIA.

For FILIP [50] we follow the original paper and implement token dropping for FILIP which the authors propose in order to reduce the large memory consumption of their method. In the original paper the authors comment on the training difficulty in the original paper (cf. in the Appendix A.3."...training is extremely unstable and the Nan loss easily happens."). We observed similar training instability in our setup across a wide range of learning rates and weight decay parameters. This training instability leads to significant performance degradation compared to CLIP. We hypothesize that the non-standard additional tricks that FILIP uses such as image augmentations, backtranslation of captions and custom prompt ensembling could potentially improve training stability; note that we do not use these tricks in order to ensure a fair comparison across methods. Given FILIP's training instability, we conducted a number of additional experiments combining CLIP and FILIP in order to better understand the training instability. Below in Tables 9 and 10 we present these results – as can be seen combining these two methods leads to some improvements on some benchmarks while some performance degradation on other benchmarks.

For PACL, as no code is publicly available at the time of writing, we closely follow the descriptions provided in the paper for our reimplementation up to one notable detail – we include a learnable temperature parameter in the loss as we found this to

---

[2]GLoRIA: `https://github.com/marshuang80/gloria`, MGCA: `https://github.com/HKU-MedAI/MGCA`

significantly improve performance. Note that the PACL objective was proposed with using frozen CLIP checkpoints and training a vision adaptor on top of the vision encoder while in this work we examine the setting of pretraining from scratch.

Finally, all methods in our paper use learned temperature parameters (instead of fixed temperatures as is done in the original MGCA, GLoRIA and PACL implementations) as our experiments showed that this significantly improved performance for all methods.

	Objective	IN	IN-V2 Th	IN-V2 MF	IN-V2 TI	IN-R	IN-C	IN-A	IN-Sketch
ViT-B/32	CLIP	66.7	66.2	58.9	71.5	63.2	42.6	15.1	51.7
	FILIP	52.7	50.7	44.0	55.8	47.1	28.7	8.4	38.2
	PACL	58.9	56.9	50.0	62.6	54.0	34.9	9.3	44.1
	GLoRIA	62.8	61.5	54.3	66.7	56.7	38.4	11.2	47.5
	MGCA	66.0	64.5	56.4	69.5	62.0	41.1	14.7	51.7
	SPARC (ours)	**68.1**	**67.0**	**59.7**	**72.0**	**64.9**	**44.5**	**16.7**	**53.2**
ViT-B/16	CLIP	71.6	70.9	63.7	74.8	71.1	**48.5**	32.2	56.8
	FILIP	56.6	55.6	48.9	59.7	54.0	33.2	14.4	43.1
	PACL	61.1	59.6	52.6	64.8	56.3	36.1	12.8	45.2
	GLoRIA	67.4	66.9	59.8	71.7	66.6	43.8	24.6	54.2
	MGCA	69.6	69.3	62.2	73.6	68.8	46.1	29.0	55.0
	SPARC (ours)	**72.6**	**71.1**	**64.4**	**75.0**	**72.0**	48.5	**33.8**	**57.3**
ViT-L/4	CLIP	77.3	75.9	69.5	79.1	78.8	59.6	**52.5**	64.5
	MGCA	75.6	73.9	68.0	77.9	77.2	56.0	45.0	63.1
	SPARC (ours)	**78.2**	**76.9**	**70.6**	**80.0**	**79.3**	**59.7**	51.9	**65.4**

Table 8: Top-1 accuracy (in %) of zero-shot classification on ImageNet (IN) and its variants ImageNet-V2 Threshold (IN-V2 Th), ImageNet-V2 Matched Frequency (In-V2 MF), ImageNet-V2 Top Images (IN-V2 TI), ImageNet-R (IN-R), ImageNet-C (IN-C), ImageNet-Sketch (IN-Sketch).

D.4. Hyperparameters details

We train all models using the AdamW (Loshchilov & Hutter, 2017) optimizer, a cosine learning rate schedule with linear warm-up of 2500 steps. For all methods, we sweep over learning rate and weight decay values in the following ranges: learning rate in $[7e-4, 9e-4, 1.1e-4]$ and weight decay in $[0.1, 0.2, 0.3]$. We use a batch size of 16348 (except for GLoRIA for which we use 4096 batch size) and we pre-train the ViT-B models for 200k steps (≈ 3.2 billion data points).

For the other SPARC hyperparameters, we set the global loss weight $\lambda_g = 0.5$ and we sweep the local loss weight in $\lambda_f \in [0.5, 1.0, 5.0, 10.0]$. Moreover, we use a learned temperature parameter τ.

For baseline specific hyperparameters, we follow the publicly available code (where available) and the original papers. For MGCA (Wang et al., 2022), as described in the paper, we set the weighing of the different losses $\lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 1$, the number of attention heads for computing the cross-modal embeddings to 1 with a 128 embedding dimension. For MGCA's crossmodal prototype alignment loss, we use 500 prototypes with $\epsilon = 0.05$ and 3 iterations for the Sinkhorn-Knopp clustering algorithm.

For FILIP, we implemented the token dropping procedure described in the paper and use 20% token dropping in our experiments.

For PACL, we closely follow the original paper in terms of implementation up to one notable detail – we include a learnable temperature parameter in the loss as we found this to significantly improve performance.

	Objective	IN	IN-V2 Th	IN-V2 MF	IN-V2 TI	IN-R	IN-C	IN-A	IN-Sketch
ViT-B/32	CLIP	66.7	66.2	58.9	71.5	63.2	42.6	15.1	51.7
	FILIP	52.7	50.7	44.0	55.8	47.1	28.7	8.4	38.2
	CLIP + FILIP	66.5	65.8	58.2	71.1	63.0	42.3	15.1	51.3
	SPARC (ours)	**68.1**	**67.0**	**59.7**	**72.0**	**64.9**	**44.5**	**16.7**	**53.2**
ViT-B/16	CLIP	71.6	70.9	63.7	74.8	71.1	**48.5**	32.2	56.8
	FILIP	56.6	55.6	48.9	59.7	54.0	33.2	14.4	43.1
	CLIP + FILIP	71.8	70.5	63.4	74.4	70.6	47.8	32.0	56.2
	SPARC (ours)	**72.6**	**71.1**	**64.4**	**75.0**	**72.0**	**48.5**	**33.8**	**57.3**

Table 9: Top-1 accuracy (in %) of zero-shot classification on ImageNet (IN) and its variants ImageNet-V2 Threshold (IN-V2 Th), ImageNet-V2 Matched Frequency (In-V2 MF), ImageNet-V2 Top Images (IN-V2 TI), ImageNet-R (IN-R), ImageNet-C (IN-C), ImageNet-Sketch (IN-Sketch). All methods have been trained on ALIGN, JFT, LTIP for the same number of training steps.

		MSCOCO						Flickr30k					
		image-to-text			text-to-image			image-to-text			text-to-image		
	Objective	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
ViT-B/32	CLIP	53.5	78.2	86.7	38.4	64.8	74.9	79.2	95.1	97.2	66.5	88.0	**93.1**
	FILIP	35.6	61.0	73.1	26.2	51.0	62.4	62.6	86.9	92.9	50.5	77.7	84.9
	CLIP + FILIP	52.0	77.0	85.6	37.8	64.4	74.5	81.2	95.4	97.1	66.8	87.7	92.3
	SPARC (ours)	**55.0**	**79.1**	**87.3**	**39.7**	**65.9**	**75.7**	**82.5**	**96.2**	**97.6**	**67.7**	**88.2**	93.0
ViT-B/16	CLIP	56.2	80.6	88.2	42.4	**68.6**	78.3	84.0	96.1	98.2	71.6	90.3	94.1
	FILIP	40.2	66.0	76.3	29.5	55.3	66.3	69.0	89.8	94.0	55.8	81.5	87.9
	CLIP + FILIP	54.9	79.0	87.4	41.3	67.7	77.5	82.7	97.0	98.4	71.1	90.5	94.7
	SPARC (ours)	**57.6**	**81.2**	**88.5**	**43.0**	**68.6**	**78.5**	**84.4**	**97.6**	**98.7**	**72.0**	**91.2**	**94.9**

Table 10: Results on zero-shot image-to-text and text-to-image retrieval on MSCOCO and Flickr30k datasets. R@i denotes Recall at i. All methods have been trained on ALIGN, JFT, LTIP for the same number of training steps.

D.5. Prompt ensembling for zero-shot classification

Following (Radford et al., 2021) and (Yao et al., 2021) we use prompt templates to augment the label for classification tasks. We use the prompt templates format from (Yao et al., 2021):

$$[\text{prefix}]\{\text{class label}\}, [\text{suffix}] \tag{9}$$

For the [prefix], we use the templates from (Radford et al., 2021). On the other hand, for the [suffix], we use the templates from (Yao et al., 2021), which shows that adding the reference word 'it' at the end of the prompt, e.g. 'I like it', further improves performance.

D.6. Memory consumption and FLOPS for the different methods

We provide detailed numbers for the FLOPS (in TFLOPS) and of the Peak Memory (in MB) in Table 12 and visualize the numbers and relative differences in Figure 5.

D.7. Semantic segmentation

For zero-shot semantic segmentation, we pass the patch embeddings through the extra dense layer and the adapter to compute the cosine similarity with the text embeddings for the ground-truth classes. Similarly to (Mukhoti et al., 2023) we compute the mean Intersection over Union (mIoU) only for the foreground classes.

	Objective	Flickr30k						MSCOCO					
		image-to-text			text-to-image			image-to-text			text-to-image		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
ViT-B/32	CLIP	79.2	95.1	97.2	66.5	88.0	**93.1**	53.5	78.2	86.7	38.4	64.8	74.9
	PACL	65.5	86.8	92.2	49.8	76.5	84.7	37.6	65.1	75.7	26.5	50.6	61.8
	GLoRIA	74.6	92.1	96.2	61.5	85.3	90.7	46.9	73.0	82.7	34.5	61.0	71.7
	MGCA	81.5	93.9	96.8	64.4	86.5	92.0	54.5	78.6	86.8	37.7	63.7	74.0
	FILIP	62.6	86.9	92.9	50.5	77.7	84.9	35.6	61.0	73.1	26.2	51.0	62.4
	SPARC (ours)	**82.5**	**96.2**	**97.6**	**67.7**	**88.2**	93.0	**55.0**	**79.1**	**87.3**	**39.7**	**65.9**	**75.7**
ViT-B/16	CLIP	84.0	96.1	98.2	71.6	90.3	94.1	56.2	80.6	88.2	42.4	**68.6**	78.3
	PACL	69.6	89.7	94.2	54.9	80.7	87.3	41.8	67.8	77.6	29.1	54.3	65.5
	GLoRIA	78.0	95.5	98.0	68.4	88.9	93.2	49.7	75.4	84.6	38.9	65.1	75.2
	MGCA	82.2	96.1	98.1	67.7	88.5	93.2	**57.6**	80.5	87.8	39.8	65.7	75.3
	FILIP	69.0	89.8	94.0	55.8	81.5	87.9	40.2	66.0	76.3	29.5	55.3	66.3
	SPARC (ours)	**84.4**	**97.6**	**98.7**	**72.0**	**91.2**	**94.9**	**57.6**	**81.2**	**88.5**	**43.0**	**68.6**	**78.5**
ViT-L/14	CLIP	84.7	96.9	98.4	73.7	**91.8**	95.4	58.6	82.6	89.1	44.8	70.5	79.5
	MGCA	85.9	96.9	98.1	73.2	91.6	95.3	**59.7**	**83.2**	**89.7**	44.3	69.6	78.8
	SPARC (ours)	**86.9**	**97.3**	**98.6**	**74.4**	91.7	**95.4**	58.9	82.9	**89.7**	**45.6**	**71.1**	**80.1**

Table 11: Results on zero-shot image-to-text and text-to-image retrieval on MSCOCO and Flickr30k datasets. R@i denotes Recall at i.

	FLOPS (TFLOPS)				Peak memory (MB)			
Objective	B = 2048	B = 4096	B = 8192	B = 16384	B = 2048	B = 4096	B = 8192	B = 16384
CLIP	1.15	2.29	4.57	9.14	4394	4452	5889	8578
PACL	1.2	2.46	5.24	12.8	4682	6267	9786	14785
GLoRIA	3.34	13.21	−	−	8013	13840	−	−
MGCA	1.16	2.31	4.62	9.23	4412	4462	5936	8681
FILIP	1.37	3.17	8.09	27.25	4394	5230	8657	15463
SPARC (ours)	1.15	2.3	4.6	9.19	4408	4450	5914	8620

Table 12: TFLOPS and peak memory usage for one update step of each method for different batch sizes.

D.8. SPARC backbones in vision language models

We train the Perceiver Resampler part of Flamingo (Alayrac et al., 2022) on the ALIGN (Jia et al., 2021), LTIP (Long Text & Image Pairs) (Alayrac et al., 2022) and VTP (Video & Text Pairs) (Alayrac et al., 2022) datasets. VTP consists of 27 million short videos paired with text descriptions, where each video if 22s on average. We use the AdamW optimizer, a cosine learning rate schedule with peak learning rate of $1e − 4$, linear warmup with 5000 warm-up steps and 250k training steps in total.

D.9. SPARC vs CLIP Faithfulness Examples

To further understand the ability of SPARC and CLIP models to faithfully describe the elements in the image, we provide several qualitative examples. Thus, for MSCOCO, we chose examples where the top-1 retrieved caption for both SPARC and CLIP is not part of the ground truth captions, but where where SPARC has higher all-token $\mathcal{K}$-Precision (Figure 6) and higher $\mathcal{K}$-Precision restricted to nouns and adjectives (Figure 7). From these figure, we notice that captions retrieved using the CLIP representations describe objects that not present in the image (e.g. "several signs for bars" when there are none present) or get the number of objects wrong (e.g. "two motorcycles" when there is only one motorcycle). Alternatively, captions retrieved using the SPARC representations are more faithful to the image, but also provide more descriptive details
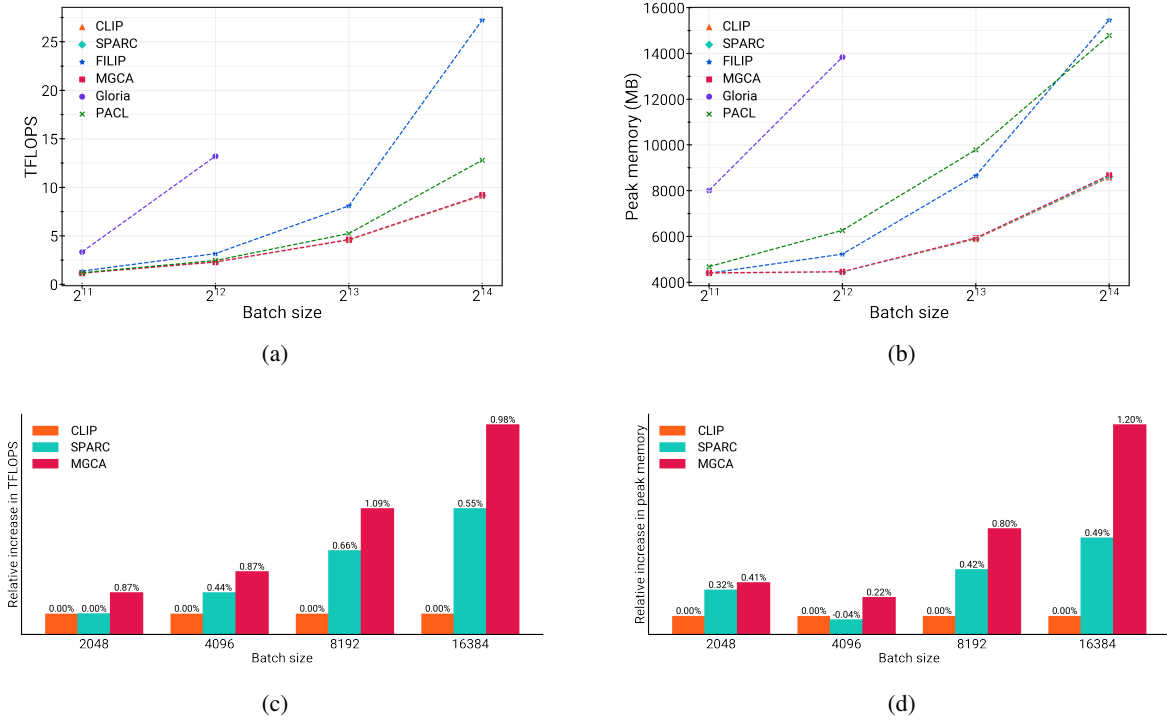
(a)

(b)

(c)

(d)

Figure 5: TFLOPS (a) and Peak Memory (b) used by all methods. Relative increase in TFLOPS (c) and Peak memory (d) when comparing SPARC and MGCA to CLIP.

(e.g. "young boy in white shirt", "dinner table with a place setting").

Image				
SPARC	a fire hydrant on the corner near a building	a man riding on the back of a motorcycle down a street	the store sign is clearly visible on the front of the building	a woman stands under a shade in an asian market next to a woman sitting
CLIP	filtered photo taken of a traffic light on a curb	the two motorcycles are in the middle of traffic	several signs for bars and restaurants are displayed on buildings	motorcycles are good transportation in the local marketplace
Ground Truth	an old rusty fire hydrant standing on a cracked sidewalk a yellow firehydrant on the sidewalk near a building a white fire hydrant a sidewalk and a building there is a old rusty fire hydrant on the street a apartment building with all glass windows around it	a person on a motor bike and a car on the street a man in the city riding a motorcycle in traffic without a helmet a person on a motorcycle with a large back wheel riding on a street a man riding on the back of a motorcycle behind a red car a group of cars driving through an intersection	a white truck parked in front of a store with lots of windows a large white building on the corner of a street a white building storefront has a van and car parked in front of it a white vehicle in front of a building a delivery truck is parked in front of a store	several people riding on a motorcycle with an umbrella open couples riding motor cycles carrying umbrellas and people sitting at tables a group of people riding scooters while holding umbrellas some tables and umbrellas sitting next to a building pedestrians and motorcyclists near an open outdoor market

Figure 6: SPARC vs CLIP vs Ground Truth for examples where SPARC has higher all-token $\mathcal{K}$-Precision ($\mathcal{K}$-P)

Image				
SPARC	a young man looks at his cell phone while at a restaurant	a group of older people sitting next to each other eating cake	a young boy in white shirt doing a trick on skateboard	a dinner table with a place setting and dish of food that has meat and vegetables
CLIP	a woman has her picture taken while in the restaurant	a woman has her picture taken while in the restaurant	a teenager attempting a jump on his skateboard	grilled meat with two sides of mashed potatoes and broccoli
Ground Truth	while in a restaurant a young man is playing with his phone a male fiddles with a smart phone in a restaurant a cluttered restaurant has a boy at a table with a phone a young man using a smart phone in a cafe a young man looking at his cell phone while seated at a restaurant	four people at a table with a birthday cake that has just been lit four people sitting around a table with a cake a group of people sitting at a table a group of diners celebrate a birthday with cake a group of people are sitting at a table with a cake	a guy with a white shirt and jeans riding a skateboard a person on a skateboard jumping it in the air small boy in white shirt and jeans jumping on top of a black board a skate boarder is performing a trick on his skateboard a boy skating on a dry land with his hands up	a plate with various foods sits on top of the place mat a plate of meat rice and vegetables are sitting on a table a plate of food with a knife and fork a drink with some food in a bowl a plate on a mat with steak rice and vegetables a plate of meat rice and vegetables is on the table

Figure 7: SPARC vs CLIP vs Ground Truth for examples where SPARC has higher $\mathcal{K}$-Precision restricted to nouns and adjectives ($\mathcal{K}$-P$_{na}$)
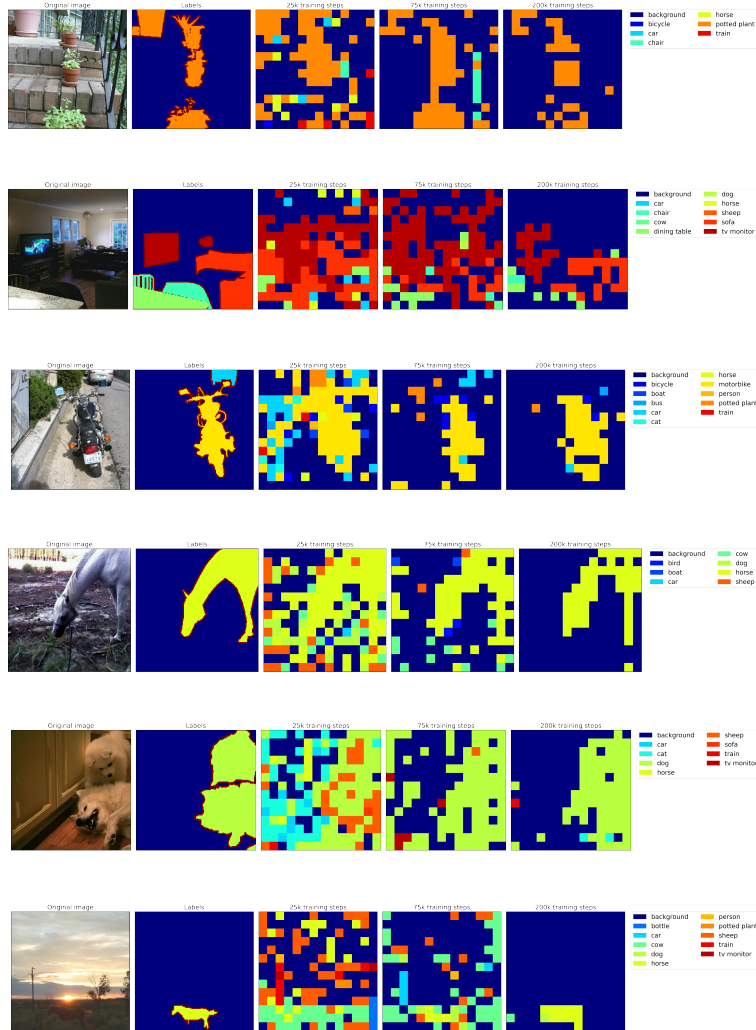
Figure 8: Qualitative results for zero-shot segmentation on Pascal VOC dataset. We illustrate the original image, pixel-level ground-truth labels and the the patch-level segmentation masks obtained from SPARC, GLoRIA and CLIP.