Retrieval-Enhanced Contrastive Vision-Text Models

Ahmet Iscen **Mathilde Caron** **Alireza Fathi** **Cordelia Schmid**
Google Research

Abstract

Contrastive image-text models such as CLIP form the building blocks of many state-of-the-art systems. While they excel at recognizing common generic concepts, they still struggle on fine-grained entities which are rare, or even absent from the pre-training dataset. Hence, a key ingredient to their success has been the use of large-scale curated pre-training data aiming at expanding the set of concepts that they can memorize during the pre-training stage. In this work, we explore an alternative to encoding fine-grained knowledge directly into the model's parameters: we instead train the model to retrieve this knowledge from an external memory. Specifically, we propose to equip existing vision-text models with the ability to refine their embedding with cross-modal retrieved information from a memory at inference time, which greatly improves their zero-shot predictions. Remarkably, we show that this can be done with a light-weight, single-layer, fusion transformer on top of a frozen CLIP. Our experiments validate that our **r**etrieval-**e**nhanced **co**ntrastive (RECO) training improves CLIP performance substantially on several challenging fine-grained tasks: for example +10.9 on Stanford Cars, +10.2 on CUB-2011 and +7.3 on the recent OVEN benchmark.

1 Introduction

In the recent years, we have witnessed a surge in the development of vision-language models that are highly adaptable to a broad spectrum of downstream tasks [8, 27, 50, 55, 65]. These models typically work by pre-training two parallel encoders using contrastive learning [46] on large-scale, carefully curated, image-text data [50]. These two-tower models learn to encode image and text into an aligned latent space which enables new appealing capabilities, such as zero-shot transfer to different downstream applications, *e.g.* image classification [50], image-text retrieval [48] or open-world recognition [35, 44]. Zero-shot transfer is a particularly effective approach to adapting to new tasks since the model is capable of handling new tasks without the need for fine-tuning on task-specific data or use of other domain adaptation protocols [63].

Although these models have achieved state-of-the-art results across various generic vision-language benchmarks, our observation is that they tend to struggle on tasks requiring a more fine-grained understanding of visual or textual entities. Our hypothesis is that this disparity largely stems from the fact that during the pre-training phase, the contrastive loss compels the image representation to align with the representation of a short, sometimes noisy, text that neither exclusively nor comprehensively describes it. For example, we hypothesize that current vision-language models are good at associating images of cars with generic concepts such as "car", "mechanics" or "road trip", because these are common words paired with car images, but less at finegrained, instance-level, associations such as the specific brand, series or year of that car. This might therefore produce poor accuracy for zero-shot fine-grained car classification, where the task is to distinguish between different cars.

Consequently, the current path taken by the research community has been to ever scale and curate the pre-training dataset in the hope of covering more and more, and cleaner image-text associations [1, 8, 50, 53]. However, one can wonder if this is a viable direction. An orthogonal effort has focused instead on *memory* or *knowledge*-based approaches [19, 22, 24, 26, 37, 38, 54]. These methods, instead

Preprint. Under review.

of statically ingesting and memorizing all the world knowledge into model parameters, propose to rely on the access to an external source of knowledge. In this work, we follow their path and explore how existing vision-language models can leverage external knowledge for improved zero-shot fine-grained predictions. Also related to this goal, K-Lite [54] proposes to improve vision-text models by enhancing the text captions with more comprehensive text definitions retrieved from an external dictionary, *i.e*. WordNet [41] or Wiktionary [43]. Some caveats of this approach are (i) the retrieval is specific for the text-tower only, (ii) initial captions are augmented within their modality only, hence limiting the potential added-value of retrieval and (iii) the external knowledge base is limited, and may not contain all of the fine-grained entities of interest.

In this work, we explore **r**etrieval-**e**nhanced **co**ntrastive (RECO) vision-text models, *i.e*. a method to equip existing vision-text models with the ability to use relevant external knowledge in a cross-modal manner for improved zero-shot predictions. In particular, our method learns to refine original CLIP [50] text and visual embeddings with cross-modal items retrieved from a memory, namely a web-scale corpus of image-text pairs. For retrieval, we have observed that the image representation can be effectively utilized to identify relevant images closely resembling the query image, or the text representation can be used to identify relevant texts closely resembling the query text. By contrast, when crossing modalities, these representations are less successful in identifying suitable matches. We hence utilize the inherent strength of learned image and text representations within their respective modalities to retrieve relevant items and aid the alignment across modalities.

Overall, as conceptually illustrated in Figure 1, we use an image representation as a query to identify the top-k most similar images and incorporate their associated text to create a multi-modal representation. In a parallel manner, given a text representation as a query, we find the top-k most similar texts and integrate their associated images to create a multi-modal representation. Through this process, we successfully transform the image and text representations into knowledge-enhanced multi-modal versions, which significantly enriches their representation power and simplifies their alignment in fine-grained zero-shot applications. This approach does not presuppose any downstream knowledge and produces a generic single model that can be used effectively across different downstream tasks. We validate our method through thorough design choice analyses. We show that our method improves over CLIP on 11 challenging fine-grained downstream tasks.

2 Related Work

Vision-text pre-training. While early works have shown the promise of representation learning from image-text paired data [11, 17, 28, 67], recent popular papers such as CLIP [50] and ALIGN [27] have truly unleashed the potential of contrastive image-text pre-training. This paradigm simply works with two parallel uni-modal encoders that learn to distinguish between aligned and non-aligned image-text pairs through a cross-modal contrastive objective [42, 46]. Appealing properties of these models are simplicity, scalability and great zero-shot performance [63]. As a result, vision-text contrastive models now form the basic building blocks of more powerful foundational models, such as CoCa [65], Flamingo [1], FLAVA [55], and PaLI [8] for example. In our work, we enhance the capabilities of the CLIP model [50], by adding a light-weight retrieval module. Nevertheless, our method is not specific to CLIP and can be applied to any vision-text model.

Knowledge-based vision-text models. Several works have focused on ways of improving upon different aspects of the contrastive vision-text models, such as their training objectives [12, 15, 66] or through scaling [9, 47]. Yet, only little exploration has been done on their combination with memory or knowledge-based techniques [2, 14, 37, 54]. REACT [37] retrieves image-text pairs from an external memory in order to build a training dataset specialized for a specific downstream task. Unlike REACT [37], our work does not require any pre-knowledge about the nature of the downstream task, and is hence applicable in a full zero-shot transfer. Another key difference is that our model can leverage items from the memory at inference time, while REACT uses retrieved items to automatically generate a training set to finetune their model. Closer to our work, K-LITE [54] learns vision-text models by leveraging external sources of knowledge (*i.e*. WordNet [41] or Wiktionary [43]) to complete captions with more descriptive content. Unlike our approach, the retrieved knowledge is uni-modal (e.g. they complement text with more text) and the external memory is not used for the image tower. Also using a knowledge-based approach but for image-only representation learning, NNCLR [14] finds the visual nearest-neighbor of each training image from a memory for contrastive learning. LGSimCLR [2] uses the language guidance to find most similar visual nearest-neighbor.

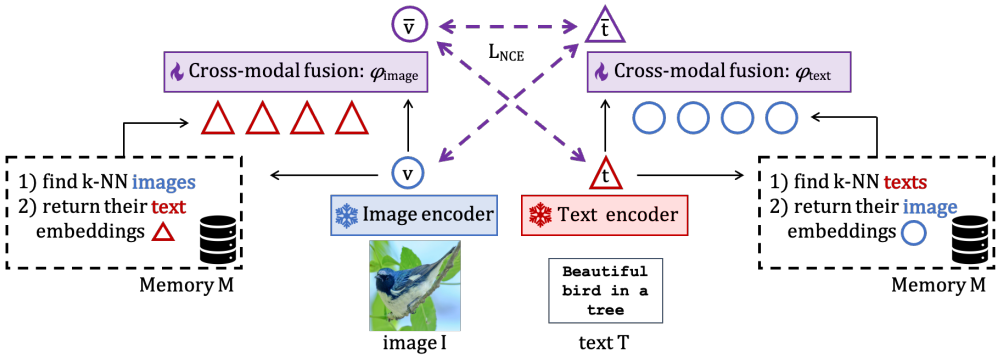

Figure 1: **RECO** works by complementing the frozen representations of pre-trained image-text encoders (such as CLIP) with knowledge retrieved from an external memory. We use an image representation as a query to identify the k most similar images and integrate their associated text embeddings to create a multi-modal representation. Likewise, given a text representation as a query, we find the top-k most similar texts and incorporate their associated images. The fusion of original and retrieved embeddings is done by learning a shallow fusion model to produce improved, multi-modal and knowledge-enhanced versions of the original embeddings. We train for alignment between the refined embeddings, as well as between the refined and original embeddings.

Unlike our work, NNCLR and LGSimCLR only learn visual representations and use retrieval to enhance their supervision during training but not at inference.

Retrieval-based methods. The main argument of the retrieval-based methods is that not all the world knowledge can be compiled into a model's parameters. Thus, the model should also learn to rely on items retrieved from an external memory at inference. Retrieval-based methods have shown their promise in various NLP tasks [4, 21, 30, 34, 60, 62]. More recently, there is an increasing interest in the computer vision for retrieval-based methods as well [19, 22, 24, 26, 37, 38, 54]. Chen *et al*. [7] and Blattmann *et al*. [3] retrieve from an external memory for generative vision modeling. Long *et al*. [38] and Iscen *et al*. [25] propose retrieval-based methods for long-tailed classification. Hu *et al*. [24] learn to retrieve from multiple external knowledge bases for visual question answering. Unlike these methods, our retrieval module does not need to be trained for a specific downstream task, and is capable of aggregating information for unseen tasks in a zero-shot manner.

3 Method

Our goal is to equip powerful pre-trained vision-language models (such as CLIP) with the ability to complement their representations with cross-modal knowledge retrieved from an external memory. We aim to do this without requiring such models to be retrained from scratch, but by simply learning a light-weight retrieval fusion module on top of them. We emphasize that this work does not propose a new model or loss but rather a new way of adapting pre-trained models to use relevant retrieved knowledge at inference time. An overview of our approach, RECO, is shown in Fig. 1.

Preliminaries. We are given a pre-trained frozen dual-encoder vision-text model f, where $\mathbf{v} = f_{\text{image}}(I)$ is the embedding of image I, and $\mathbf{t} = f_{\text{text}}(T)$ is the embedding of text T. We say that these embeddings are *uni-modal* since they are obtained purely from a single modality, either image or text. We assume that image and text embedding spaces are already *aligned*, meaning that they have been trained to produce similar representations for matching image-text pairs and dissimilar representations for non-matching pairs [27, 46, 50, 66]. This alignment is usually obtained by minimizing the InfoNCE loss (or contrastive loss) [46] between embeddings of different modalities:

$$\mathcal{L}_{\text{NCE}}(\mathbf{V}, \mathbf{T}) = -\sum_{i=1}^{n} \left[\log \frac{e^{\mathbf{v}_i^\top \mathbf{t}_i / \tau}}{\sum_j e^{\mathbf{v}_i^\top \mathbf{t}_j / \tau}} + \log \frac{e^{\mathbf{v}_i^\top \mathbf{t}_i / \tau}}{\sum_j e^{\mathbf{v}_j^\top \mathbf{t}_i / \tau}} \right], \tag{1}$$

where $\mathbf{V}$ (resp. $\mathbf{T}$) is the matrix composed of the n visual (resp. text) embeddings in the minibatch and τ is the temperature parameter. We propose to augment the text and visual embeddings, i.e. $\mathbf{t}$ and $\mathbf{v}$, with external cross-modal knowledge in order to enhance both their expressiveness and their cross-modality alignment. In the following of this section, we first detail how we retrieve relevant cross-modal knowledge based on within-modality search. Second, we present how we learn to fuse the retrieved information into the original embeddings.

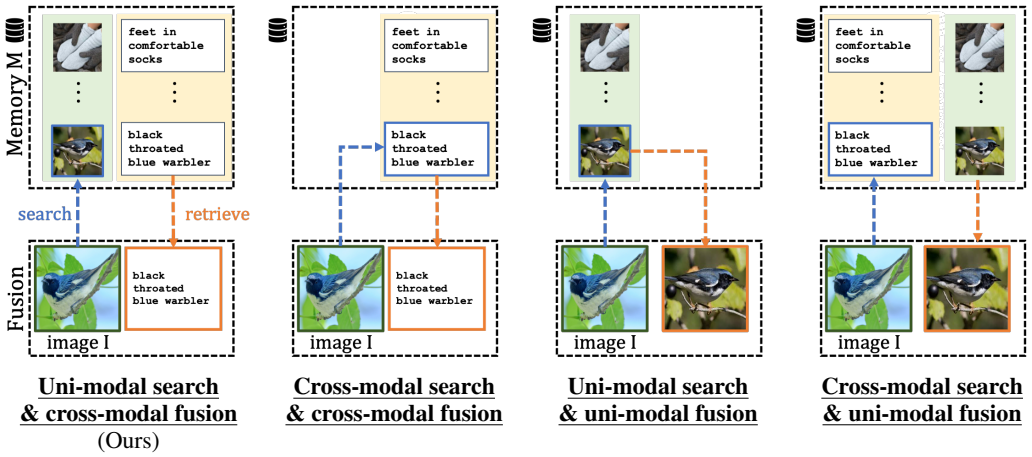

Figure 2: **Conceptual comparison of uni-/cross- modal search and uni-/cross- fusion.** We illustrate the different scenarios for an input image I while the scenarios for text input T are shown in Appendix.

3.1 Retrieving cross-modal external knowledge

Memory. We define the external source of knowledge by a memory $\mathcal{M} = \{(I_i, T_i)\}_{i=1}^{M}$ of M image-text pairs. We assume that $\mathcal{M}$ is very large and covers a broad coverage of concepts. In practice, only a small-subset of $\mathcal{M}$ is relevant for a given input query. Thus, we only consider the k most relevant items from $\mathcal{M}$ for each input obtained by the nearest neighbour search. We denote by $\text{KNN}(\mathbf{v}, \mathcal{M})$ and $\text{KNN}(\mathbf{t}, \mathcal{M})$ the sets formed by the embeddings of the k most relevant items to the queries $\mathbf{v}$ and $\mathbf{t}$ from the memory, where KNN refers to the nearest-neighbour retrieval module.

Cross-modal fusion. Our goal is to augment the text and visual original embeddings with cross-modal knowledge, not necessarily learned during the pre-training stage. For example, given the class name *Yellow bellied flycatcher* in a fine-grained bird classification problem such as CUB [59], we first look for captions in the memory that are semantically similar to the given class name. We then augment the class name representation with the visual representations of the retrieved similar captions, *i.e.* with what an *Yellow bellied flycatcher* looks like. Likewise, given a visual representation of a bird, we look for similar images in $\mathcal{M}$ and use their corresponding captions in the hope that some of them might contain useful information for our problem such as the species of that bird. Specifically, for a given text or image input, the retrieval module $\text{KNN}(., \mathcal{M})$ returns items with the opposite modality than that of the input. We use the subscripts v or t to specify the modality of the retrieved embeddings. That is, $\text{KNN}_t(\mathbf{v}, \mathcal{M})$ returns text embeddings from an image input and $\text{KNN}_v(\mathbf{t}, \mathcal{M})$ returns image embeddings for text input.

Note that we also evaluate *uni-modal* fusion in our experiments, i.e. complementing visual representations with the retrieved visual knowledge and text representation with the retrieved captions. However, we find in practice that this variant leads to poorer performance than cross-modal fusion, as shown in Tab. 3. Intuitively, we hypothesize that this is because the signal brought by cross-modal fusion is richer due to the complementarity of the different modalities [25].

Uni-modal search. We choose to search relevant items in the memory $\mathcal{M}$ based on within-modality similarities, which we refer to as "uni-modal search" as opposed to "cross-modal search". Specifically, we use text-to-text similarity ($t \rightarrow t$) to identify suitable content from a text embedding $\mathbf{t}$ and image-to-image similarity ($v \rightarrow v$) to retrieve relevant matches from a visual embedding $\mathbf{v}$. Formally, let us denote by $\mathbf{V}^{\mathcal{M}}$ and $\mathbf{T}^{\mathcal{M}}$ all the image and text embeddings from $\mathcal{M}$ given by our pretrained vision-text model f, *i.e.* we have $\mathbf{V}^{\mathcal{M}} = [f_{\text{image}}(I_1), \ldots, f_{\text{image}}(I_M)]$ and $\mathbf{T}^{\mathcal{M}} = [f_{\text{text}}(T_1), \ldots, f_{\text{text}}(T_M)]$. The retrieval module is hence finally denoted as $\text{KNN}_t^{v \rightarrow v}(\mathbf{v}, \mathcal{M}) = \mathbf{T}^{\mathcal{M}}{}_{\text{NN}(\mathbf{v}; \mathbf{V}^{\mathcal{M}})}$, *i.e.* for an input image embedding $\mathbf{v}$, the k-NN search is done between $\mathbf{v}$ and $\mathbf{V}^{\mathcal{M}}$, but the corresponding k-NN indices from the text embeddings $\mathbf{T}^{\mathcal{M}}$ are selected. Similarly, we denote the retrieval process as $\text{KNN}_v^{t \rightarrow t}(\mathbf{t}, \mathcal{M}) = \mathbf{V}^{\mathcal{M}}{}_{\text{NN}(\mathbf{t}; \mathbf{T}^{\mathcal{M}})}$ for an input text embedding $\mathbf{t}$.

We also evaluate cross-modal search but find that this leads to much poorer performance, especially in fine-grained problems, as shown in Tab. 3. A possible explanation is that the uni-modal search is an easier task, hence the retrieved elements are more relevant (because more similar) to the input. On

the other hand, cross-modal search suffers from the pre-trained CLIP model's lack of fine-grained alignment between the different modalities, resulting in noisier retrieval. Note that another advantage of uni- *versus* cross- modal search is that the latter requires the pre-trained image and text encoders to be already aligned while we can potentially let go of this hypothesis with uni-modal search. Finally, we illustrate the different scenarios for uni-modal *versus* cross-modal retrieval and fusion in Fig. 2 and quantitatively benchmark these scenarios in Tab. 3.

3.2 Learning how to fuse the retrieved knowledge

Our goal is to refine the original image and text embeddings $\mathbf{v}$ and $\mathbf{t}$ with the cross-modal knowledge gathered from $\mathcal{M}$. We denote these refined image and text embeddings by $\overline{\mathbf{v}}$ and $\overline{\mathbf{t}}$, defined as $\overline{\mathbf{v}} = \phi_{\text{image}}(\mathbf{v}, \text{KNN}_t^{v\to v}(\mathbf{v}, \mathcal{M}))$ and $\overline{\mathbf{t}} = \phi_{\text{text}}(\mathbf{t}, \text{KNN}_v^{t\to t}(\mathbf{t}, \mathcal{M}))$, where ϕ is the *fusion model*.

Transformer fusion. We model ϕ_{image} and ϕ_{text} as one-layer multi-head self-attention transformer encoders [13, 58]. Intuitively, this choice allows the original embedding to attend to all the retrieved elements in the fusion process. Note that while the fusion models for text and image encoders have identical architectures, they do not share parameters. In practice, the fusion module has a total of 3.16M parameters, which corresponds to only 2% of the total parameter count when using CLIP-B/32 as the backbone f. We have experimented with bigger fusion modules (see Appendix) but find that this light-weight solution works well in practice. We have also tried mean fusion of retrieved and original elements by simply averaging their embeddings but have found in practice that it performs poorly (see Tab. 3). Intuitively, the model needs to learn how to incorporate this new information, by, for example, learning how to omit or enhance some of the retrieved elements.

Learning. We train the fusion model ϕ on a dataset $\mathcal{D} = \{(I_i, T_i)\}_{i=1}^N$ by performing retrieval at training time from the memory $\mathcal{M}$. The pre-trained encoder f is kept frozen. We minimize the alignment loss between the refined embeddings which formally amounts to minimizing the InfoNCE loss of Eq. (1) with the refined embeddings instead of original embeddings, *i.e.* minimizing $\mathcal{L}_{\text{NCE}}(\overline{\mathbf{V}}, \overline{\mathbf{T}})$. We find that it is also sometimes beneficial to perform retrieval for only one of the branches (text or image) at inference time depending on the nature of the downstream task (see Tab. 4). Therefore, we also align the original and refined embeddings by minimizing the following "cross" loss terms: $\mathcal{L}_{\text{NCE}}(\mathbf{V}, \overline{\mathbf{T}})$ and $\mathcal{L}_{\text{NCE}}(\overline{\mathbf{V}}, \mathbf{T})$. This allows to disable one of branches at inference time, since refined and original embeddings are now also aligned. Overall, we minimize:

$$\mathcal{L} = \mathcal{L}_{\text{NCE}}(\overline{\mathbf{V}}, \overline{\mathbf{T}}) + \mathcal{L}_{\text{NCE}}(\overline{\mathbf{V}}, \mathbf{T}) + \mathcal{L}_{\text{NCE}}(\mathbf{V}, \overline{\mathbf{T}}). \tag{2}$$

4 Experiments

In this section, we first detail our experimental protocol, then demonstrate the performance of RECO on several zero-shot benchmarks, and finally propose a thorough analysis of several design choices.

4.1 Experimental setup

Training details. We train the fusion model on top of a frozen CLIP [50]. We use the CLIP-B/32 or the CLIP-L/14 versions. We train on Conceptual Captions 12M ("$\text{CC}_{12\text{M}}$") [6], an image-text dataset containing about 10M pairs. We use a batch size of 4096, learning rate of $1e^{-3}$ decayed with a cosine schedule and weight decay of $1e^{-5}$. The temperature parameter is learned [50]. Training is done for 10 epochs, which lasts about 10 hours on a $4x4$ TPUv2 pod. For the memory, we use the subset of WebLI [8] containing 1B image-text pairs. We remove the near-duplicates of the test images from the memory. We have also explored using smaller but publicly available memory such as LAION-400M dataset [53] and show the results in Appendix.

Evaluation datasets. We consider the following six image classification datasets: Stanford Cars ("Cars") [32], CUB-200-2011 ("CUB") [59], Oxford Flowers ("Flowers") [45], ImageNet-1k ("Im1k") [52], Places365 ("Pl365") [68] and Stanford Dogs ("Dogs") [31]. We also report performance on text-to-image ("T→I") and image-to-text ("I→T") retrieval on Flickr30k ("Flickr") [48] and MS COCO ("COCO") [36]. Finally, we consider the recent Open-domain visual entity recognition (OVEN) benchmark [23], containing 729K test images possibly belonging to 6M entity candidates. More details about these datasets can be found in Appendix or in their corresponding publication.

Evaluation protocol. We evaluate in the zero-shot setting for all the considered benchmarks, meaning that no adaptation is done to the downstream task. As common in the literature [27, 50, 55, 66], we

Table 1: **Zero-shot transfer to image classification and retrieval.** We report top-1 accuracy for classification and recall@1 for retrieval. We show the improvements obtained with RECO on top of CLIP-B/32 and CLIP-L/14: *absolute* performance gains are between brackets. For reference, we also include the performance of K-Lite [54] (another retrieval-augmented method) and other standard image-text foundation models (Flava [55], Align-base [27], LiT-L/16 [66] or PaLI-17B [8]). We also report the total parameter count ("# par.") of the different models (in Million).

		Image classification						T→I		I→T	
Method	# par.	Cars	CUB	Flowers	Im1k	Pl365	Dogs	COCO	Flickr	COCO	Flickr
CLIP-B/32	151	57.2	52.8	62.1	63.5	40.6	58.6	30.2	61.1	51.2	80.9
+ RECO	154	68.1 (+10.9)	63.0 (+10.2)	67.9 (+5.8)	64.6 (+1.1)	42.2 (+1.6)	59.7 (+1.1)	33.6 (+3.4)	65.7 (+4.6)	52.2 (+1.1)	81.8 (+0.9)
CLIP-L/14	428	75.6	61.7	75.6	75.5	42.0	72.7	35.2	68.6	57.2	87.5
+ RECO	435	**82.8** (+7.2)	**73.4** (+11.7)	**79.5** (+3.9)	**76.1** (+0.6)	43.6 (+1.6)	**73.9** (+1.2)	38.7 (+3.5)	**72.6** (+4.0)	**58.0** (+0.8)	**88.5** (+1.0)
Other approaches											
K-Lite [54]	151	10.0	–	78.6	52.3	–	–	–	–	–	–
Flava [55]	172	–	–	–	–	–	–	38.4	65.2	42.7	67.7
Align [27]	247	78.7	38.2	64.9	67.6	**44.0**	56.3	**40.2**	**72.6**	55.1	86.7
LiT-L [66]	638	24.8	61.7	74.1	75.8	42.6	70.8	31.1	57.6	48.5	77.7
PaLI [8]	17,000	–	–	–	72.1	–	–	–	–	–	–

Table 2: **Zero-shot performance on OVEN.** We report top-1 accuracy on seen and unseen categories and their harmonic mean. We also indicate the total number of parameters of each model ("# params").

Method	# params (M)	Seen	Unseen	Harmonic mean
Zero-shot				
PaLI-17B [8]	17,000	4.4	1.2	1.9
CLIP-L/14 [50]	428	5.6	4.9	5.3
CLIP-L/14 [50] + RECO (Ours)	435	**12.4** (+6.8)	**12.7** (+7.8)	**12.6** (+7.3)
Fine-tuning on the OVEN Seen categories				
CLIP-L/14 Fusion [23]	880	33.6	4.8	8.4
PaLI-3B [8]	3,000	19.1	6.0	9.3
CLIP-L/14 CLIP2CLIP [23]	860	12.6	10.5	11.5
PaLI-17B [8]	17,000	28.3	11.2	16.1

add prompts to the text of the downstream tasks, following [66]. We report the top-1 accuracy for all the tasks. We detail all evaluation protocols in Appendix.

4.2 Zero-shot transfer

Image classification and image/text retrieval. In Tab. 1, we observe that RECO allows to boost the zero-shot performance of CLIP on zero-shot image classification and retrieval, with large improvements especially on the fine-grained datasets. For example, we improve the original CLIP-B/32 accuracy by $+10.9$ on Cars, $+10.2$ on CUB and $+5.8$ on Flowers. The performance is also improved on less fine-grained benchmarks such as ImageNet or Places, though by more moderate margins (i.e. respectively $+1.1$ and $+1.6$). Secondly, we see in Tab. 1 that the performance gains are consistent across the two considered vision-text backbones (CLIP-B/32 and CLIP-L/14). For reference, we also report in Tab. 1 the numbers from other popular vision-text approaches [8, 27, 55, 66]. Note that our work is orthogonal to the choice of vision-text model and we expect that combining it with other powerful backbones, *e.g.* ALIGN [27], OpenCLIP [9] or CoCa [65], has the potential to bring further performance gains, especially in the fine-grained settings where their performance might be sub-optimal. Overall, the experiment in Tab. 1 confirms our initial motivation that retrieval from an external memory improves zero-shot recognition tasks, especially in fine-grained settings.

Open-domain visual entity recognition (OVEN). In Tab. 2, we show the zero-shot performance of RECO on the OVEN benchmark. We see that our method improves greatly over CLIP-L/14 on this challenging task, with an impressive relative improvement of $+138\%$. Note that we do not train or fine-tune our model on the OVEN training set. Remarkably, we observe in Tab. 2 that RECO also significantly outperforms much bigger models which are directly *fine-tuned for this task*, for example CLIP2CLIP [23] or PaLI-3B [8] while using respectively $2\times$ and $7\times$ less parameters. It even comes close to the performance of PaLI-17B while being $39\times$ smaller and not using any fine-tuning.

Table 3: **Uni-modal search for cross-modal fusion.** We report top-1 accuracy for zero-shot image classification. We evaluate the impact of uni-modal versus cross-modal search and uni-modal versus cross-modal fusion. These different mechanisms are conceptually illustrated in Fig. 2. We report *absolute* improvement between brackets and the average *relative* improvement over not using retrieval (i.e. CLIP performance) in the last row ("Avg. rel. Δ").

	Search	Fusion	Cars	CUB	Flowers	Im1k	Pl365	Avg. rel. Δ
	–	–	57.2	52.8	62.1	63.5	40.6	–
	ϕ = *Transformer fusion*							
1	Uni-modal	Cross-modal	**68.1** (+10.9)	**63.0** (+10.2)	**67.9** (+5.8)	**64.6** (+1.1)	**42.5** (+1.9)	**+ 9.0 %**
2	Cross-modal	Cross-modal	56.6 (-0.6)	53.8 (+1.0)	64.3 (+2.2)	64.3 (+0.8)	42.4 (+1.8)	+ 1.7 %
3	Uni-modal	Uni-modal	57.3 (+0.1)	51.2 (-1.6)	62.2 (+0.1)	62.1 (-1.4)	41.7 (+1.1)	− 0.4 %
4	Cross-modal	Uni-modal	54.0 (-3.2)	50.7 (-2.1)	61.4 (-0.7)	62.3 (-1.2)	41.2 (+0.6)	− 1.9 %
	ϕ = *Mean fusion*							
5	Uni-modal	Cross-modal	46.9 (-10.3)	44.9 (-7.9)	50.5 (-11.6)	40.1 (-23.4)	23.7 (-16.9)	− 21.7 %
6	Cross-modal	Cross-modal	43.7 (-13.5)	45.3 (-7.5)	58.7 (-3.4)	55.2 (-8.3)	32.7 (-7.9)	− 11.0 %
7	Uni-modal	Uni-modal	44.0 (-13.2)	47.2 (-5.6)	61.3 (-0.8)	55.1 (-8.4)	36.2 (-4.4)	− 9.8 %
8	Cross-modal	Uni-modal	33.4 (-23.8)	30.2 (-22.6)	38.9 (-23.2)	40.0 (-23.5)	24.7 (-15.9)	− 33.0 %

4.3 Design choice analyses

In this section, we validate several components of our model, namely the uni-modal search and cross-modal fusion, training of the fusion module and the number of retrieved elements from the memory. We also propose some qualitative examples to help understanding why RECO improves over CLIP performance. We use ViT-CLIP-B/32 throughout this section.

Uni-modal search and cross-modal fusion. In Tab. 3, we evaluate different alternatives for our method, namely (i) performing cross-modal search in the memory instead of uni-modal search and (ii) fusing uni-modal items (i.e. combining text with text and image with image) instead of cross-modal fusion. These different scenarios (uni- *versus* cross- modal search and fusion) are detailed in Section 3.1 and conceptually illustrated in Fig. 2. Firstly, we observe in Tab. 3 that uni-modal search (row 1) leads to a better performance compared to cross-modal search (row 2), with $+9.0$ *versus* $+1.7$ average relative improvement over CLIP. We remark that the gap is especially important for fine-grained datasets such as Cars, CUB and Flowers. This agrees with our hypothesis that cross-modal search suffers from the pre-trained CLIP model's lack of fine-grained alignment between different modalities. By contrast, using the inherent strength of image and text representations within their respective modalities allows to retrieve relevant matches, as qualitatively observed in Fig. 4.

Secondly, we observe in Tab. 3 that uni-modal fusion (rows 3 and 4) works substantially worse than cross-modal fusion (rows 1 and 2). Indeed, we see that augmenting text embeddings with other text embeddings and image embeddings with other image embeddings does not bring any significant improvement over the baseline, and even tends to hurt the performance. Intuitively, a possible explanation is that cross-modal fusion allows us to inject complementary signal into the original embeddings [25]. By contrast, uni-modal provides signal that is already similar to the input, hence not as much additional information. Finally, we see in Tab. 3 that all the variants (rows 5, 6, 7 and 8) fail when simply averaging retrieved and original embeddings instead of learning the fusion with a transformer. This highlights the importance of *learning* to incorporate the retrieved items to the original embeddings before deploying the model at inference.

Image and text retrieval fusion modules. In Tab. 4, we compare models trained to fuse only text original embeddings (row 1), only image original embeddings (row 2) or both (row 3). We observe that while models trained to fuse only image or text perform reasonably well on some benchmarks, they typically lag behind on other benchmarks. For example, the model trained for only image fusion (row 2) is strong on zero-shot Dogs benchmark but behind on CUB and COCO. Secondly, as shown in Tab. 4, unlike the vision-only or text-only variants, our model can be used in different modes at inference time in a flexible manner. Indeed, because we have trained it to align the refined embeddings with the original ones (see the cross terms in the loss 2), we can choose to disable the retrieval for one of the branches at inference time, depending on the task. Therefore, we compare in Tab. 4 different options at inference time: using retrieval only for the image input, only for the text input or for both of them, denoted respectively by $\overline{\mathbf{v}}$, $\overline{\mathbf{t}}$ and $\overline{\mathbf{v}}\&\overline{\mathbf{t}}$. We observe in Tab. 4 that depending on the nature of the task, one of these options might be preferable over the others. For example, for image

Table 4: **Image and text retrieval fusion modules.** We report zero-shot top-1 accuracy for image classification and recall@1 for image retrieval. We compare models trained only for text fusion (row 1), image fusion (row 2) or both (row 3). Our model can be used in different modes at inference: retrieval only for image ($\overline{\mathbf{v}}$), retrieval only for text ($\overline{\mathbf{t}}$) or retrieval for both image and text ($\overline{\mathbf{v}}$&$\overline{\mathbf{t}}$).

				CUB				Dogs				COCO T→I			
	ϕ_{image}	ϕ_{text}	fusion training loss	$\overline{\mathbf{v}}$	$\overline{\mathbf{t}}$	$\overline{\mathbf{v}}$&$\overline{\mathbf{t}}$	Best	$\overline{\mathbf{v}}$	$\overline{\mathbf{t}}$	$\overline{\mathbf{v}}$&$\overline{\mathbf{t}}$	Best	$\overline{\mathbf{v}}$	$\overline{\mathbf{t}}$	$\overline{\mathbf{v}}$&$\overline{\mathbf{t}}$	Best
1		✓	$\mathcal{L}_{\text{NCE}}(\mathbf{V}, \overline{\mathbf{T}})$	✗	59.3	✗	59.3	✗	59.6	✗	59.6	✗	33.3	✗	33.3
2	✓		$\mathcal{L}_{\text{NCE}}(\overline{\mathbf{V}}, \mathbf{T})$	59.7	✗	✗	59.7	59.2	✗	✗	59.2	31.3	✗	✗	31.3
3	✓	✓	$\mathcal{L}_{\text{NCE}}(\overline{\mathbf{V}}, \overline{\mathbf{T}})+\mathcal{L}_{\text{NCE}}(\mathbf{V}, \overline{\mathbf{T}})+\mathcal{L}_{\text{NCE}}(\overline{\mathbf{V}}, \mathbf{T})$	60.0	58.4	63.0	**63.0**	59.7	59.7	59.4	**59.7**	31.9	33.6	31.7	**33.6**

Method	Train data	CUB
CLIP-B/32	–	52.8
+ CLIP-style	$\text{CC}_{12\text{M}}$	44.8
+ CLIP-style	$\text{CC}_{12\text{M}}$ + $\mathcal{R}_{\text{Webli}}$	46.8
+ RECO	$\text{CC}_{12\text{M}}$ + $\mathcal{R}_{\text{Webli}}$	**63.0**

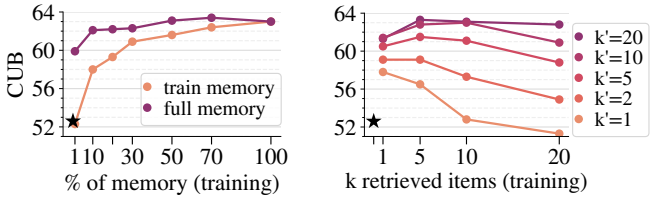

Figure 3: **(left) Disantangling the effect of additional training and RECO. (middle) Effect of updating the memory after training. (right) Effect of the number k of retrieved elements.** We report zero-shot top-1 accuracy on CUB. The CLIP baseline is shown with symbol ★.

classification we see that augmenting the image embeddings with retrieved text has more positive impact than augmenting the text embeddings, though the best of performance is obtained with both. On the other hand, text and image retrievals seem to benefit more from augmenting the text rather than the image side. This intuitively can be explained by the fact that text descriptions in retrieval benchmarks are typically highly specific compared to the class names in image classification and so augmenting with visual examples of what they refer to greatly helps the alignment. We demonstrate qualitative examples of this hypothesis in Appendix. Overall, at inference time, one can choose the best inference mode for a particular downstream task by validation on a held-out set.

Is the performance boost merely due to additional training? We validate the hypothesis that the performance gains are indeed due to our method, and not due to training an additional layer on top of CLIP. We replace RECO with an MLP layer of the same capacity initialized from scratch. We train it in a CLIP-style manner on the subset of Webli that we use when training RECO. We denote this subset by $\mathcal{R}_{\text{Webli}}$: it contains the $k = 10$ nearest-neighbors for each CC12M datapoint retrieved from the Webli dataset, and contains 61M unique examples. Results are presented in Fig. 3 (left). We observe that training an extra layer on top of CLIP does not bring any performance gains and even deteriorates its performance. This is somehow expected since CLIP was extensively trained on a large and rich dataset [50] and additional training on a relatively small dataset deteriorates the general-purpose property of its representations. Overall, this experiment validates our retrieval-augmented approach.

Updating the memory after training. A clear advantage of the retrieval-based models is that the external memory can be updated with additional, and more contemporary information. We evaluate the effectiveness of RECO when using a larger memory that is not observed during the training. We first create various random subsets of Webli by randomly removing a percentage of data. Then, we train separate RECO models with each Webli subset as its memory. At inference, we evaluate each RECO model either with the subset of memory that it was trained with, or the full Webli memory. Results are shown in Fig. 3 (center). We observe that training and evaluating RECO with only 1% of Webli as the memory does not show improvements compared to the CLIP baseline. However, we observe a significant improvement when evaluating the same model with full Webli memory at inference. This confirms that RECO is capable of utilizing an updated memory without re-training.

Effect of the number of retrieved elements. In Fig. 3 (right), we study the effect of the number of retrieved elements in the memory. We evaluate different numbers of k-NN during the training and inference time, *i.e.* we train our model with k items from the memory but use k' at inference. We see in Fig. 3 (right) that RECO generally obtains a higher performance when $k' > k$ at inference. Interestingly, the performance saturates after $k = 10$. An explanation is that increasing the number of retrieved elements goes with a reduction of the relevancy of the retrieved items.

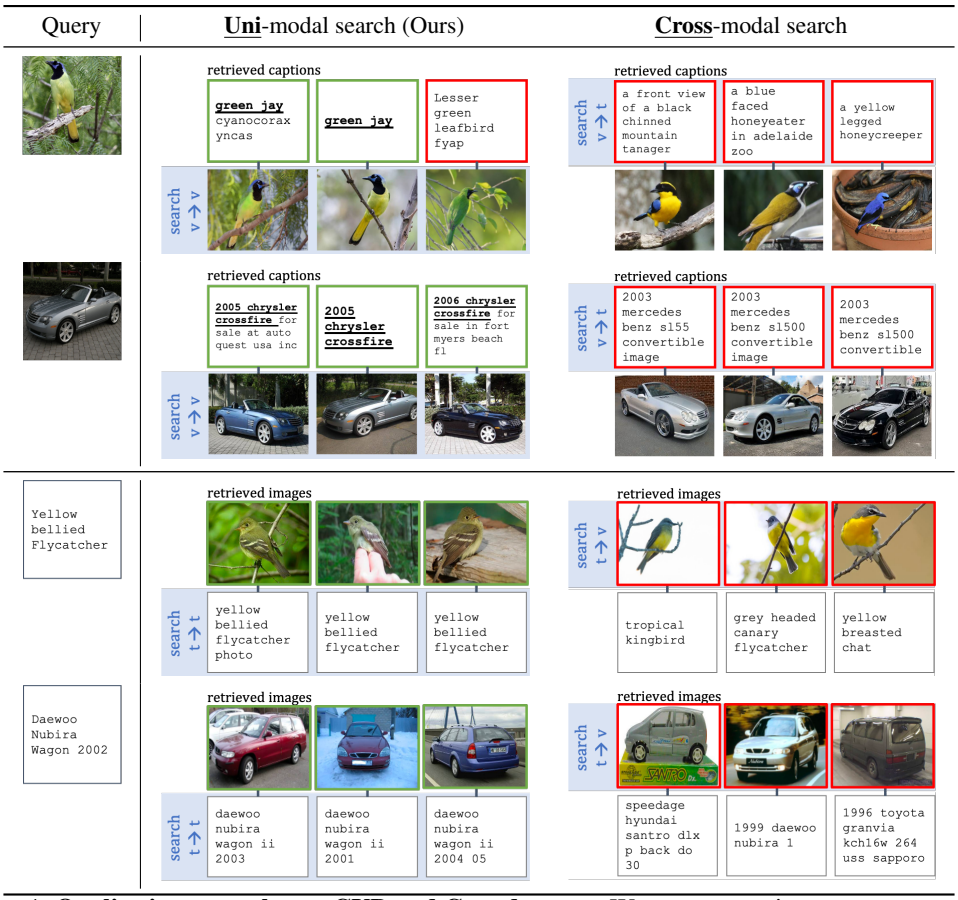

Figure 4: **Qualitative examples on CUB and Cars datasets.** We compare uni- versus cross- modal search for two image queries (top) and two text queries (bottom). Uni-modal search allows to find more suitable matches to the query, which improves the relevancy of the fused elements. We frame in red (resp. green) the unrelevant (resp. relevant) retrieved items to be fused with the query.

Qualitative study. In Fig. 4, we provide illustrative examples of why RECO can be useful for fine-grained image classification on CUB or Cars datasets. We compare our method with a variant using cross-modal search instead of uni-modal search to illustrate the importance of using the inherent strength of image-only and text-only representations. We observe in Fig. 4 that uni-modal search allows to retrieve better matches for the query. This is because image-to-image or text-to-text search retrieves more similar items to the query than crossing modalities. As a result, retrieved items are more accurate, which leads to a higher accuracy for fine-grained tasks.

5 Conclusion

In this paper, we introduce RECO, a method that enhances the fine-grained recognition capabilities of pre-trained vision-text models. Our approach shows the importance of uni-modal retrieval, yet cross-modal fusion for image and text inputs. We show that RECO consistently improves the performance on 11 different zero-shot tasks and that the gains are especially important in challenging fine-grained tasks. We include a broader impact discussion in Appendix.

Limitations. A limitation of this work is that it assumes to have access to a large and rich source of image-text pairs knowledge. While we show in Appendix that public datasets , *e.g.* LAION [53], can serve this purpose, the best of performance is obtained with a large private memory. Alternatively, one could use search engine APIs as the memory. Another limitation is that the performance gains of RECO come at the cost of increased inference time. In practice, we use a highly-optimized approximate k-NN algorithm [20], and querying 1B examples in Webli takes milliseconds. It is also possible to mitigate this issue by reducing the number of retrieved items. As shown in Fig. 3, retrieving a single element already improves over CLIP substantially, while reducing compute requirements.

References

[1] 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)

[2] Banani, M.E., Desai, K., Johnson, J.: Learning visual representations via language-guided sampling. arXiv preprint arXiv:2302.12248 (2023)

[3] Blattmann, A., Rombach, R., Oktay, K., Müller, J., Ommer, B.: Semi-parametric neural image synthesis. arXiv preprint arXiv:2204.11824 (2022)

[4] Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., Van Den Driessche, G.B., Lespiau, J.B., Damoc, B., Clark, A., et al.: Improving language models by retrieving from trillions of tokens. In: Proceedings of the International Conference on Machine Learning (ICML) (2022)

[5] Bossard, L., Guillaumin, M., Van Gool, L.: Food-101–mining discriminative components with random forests. In: Proceedings of the European Conference on Computer Vision (ECCV) (2014)

[6] Changpinyo, S., Sharma, P., Ding, N., Soricut, R.: Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2021)

[7] Chen, W., Hu, H., Saharia, C., Cohen, W.W.: Re-imagen: Retrieval-augmented text-to-image generator. arXiv preprint arXiv:2209.14491 (2022)

[8] 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. International Conference on Learning Representations (ICLR) (2023)

[9] Cherti, M., Beaumont, R., Wightman, R., Wortsman, M., Ilharco, G., Gordon, C., Schuhmann, C., Schmidt, L., Jitsev, J.: Reproducible scaling laws for contrastive language-image learning. arXiv preprint arXiv:2212.07143 (2022)

[10] De Vries, T., Misra, I., Wang, C., Van der Maaten, L.: Does object recognition work for everyone? In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

[11] Desai, K., Johnson, J.: Virtex: Learning visual representations from textual annotations. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2021)

[12] Dong, X., Zheng, Y., Bao, J., Zhang, T., Chen, D., Yang, H., Zeng, M., Zhang, W., Yuan, L., Chen, D., et al.: Maskclip: Masked self-distillation advances contrastive language-image pretraining. arXiv preprint arXiv:2208.12262 (2022)

[13] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (ICLR) (2021)

[14] Dwibedi, D., Aytar, Y., Tompson, J., Sermanet, P., Zisserman, A.: With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2021)

[15] Gao, Y., Liu, J., Xu, Z., Zhang, J., Li, K., Ji, R., Shen, C.: Pyramidclip: Hierarchical feature alignment for vision-language model pretraining. Advances in Neural Information Processing Systems (NeurIPS) (2022)

[16] Gerry: Sports100: 100 sports image classification. `https://www.kaggle.com/datasets/gpiosenka/sports-classification/metadata` (2021), accessed: 2023-05-23

[17] Gomez, L., Patel, Y., Rusinol, M., Karatzas, D., Jawahar, C.: Self-supervised learning of visual features through embedding images into text topic spaces. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

[18] Goyal, Y., Khot, T., Summers-Stay, D., Batra, D., Parikh, D.: Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2017)

[19] Gui, L., Wang, B., Huang, Q., Hauptmann, A., Bisk, Y., Gao, J.: Kat: A knowledge augmented transformer for vision-and-language. arXiv preprint arXiv:2112.08614 (2021)

[20] Guo, R., Sun, P., Lindgren, E., Geng, Q., Simcha, D., Chern, F., Kumar, S.: Accelerating large-scale inference with anisotropic vector quantization. In: Proceedings of the International Conference on Machine Learning (ICML) (2020)

[21] Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.: REALM: Retrieval augmented language model pre-training. In: Proceedings of the International Conference on Machine Learning (ICML) (2020)

[22] Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.: Retrieval augmented language model pre-training. In: Proceedings of the International Conference on Machine Learning (ICML) (2020)

[23] Hu, H., Luan, Y., Chen, Y., Khandelwal, U., Joshi, M., Lee, K., Toutanova, K., Chang, M.W.: Open-domain visual entity recognition: Towards recognizing millions of wikipedia entities. arXiv preprint arXiv:2302.11154 (2023)

[24] Hu, Z., Iscen, A., Sun, C., Wang, Z., Chang, K.W., Sun, Y., Schmid, C., Ross, D.A., Fathi, A.: Reveal: Retrieval-augmented visual-language pre-training with multi-source multimodal knowledge memory. arXiv preprint arXiv:2212.05221 (2022)

[25] Iscen, A., Fathi, A., Schmid, C.: Improving image recognition by retrieving from web-scale image-text data. arXiv preprint arXiv:2304.05173 (2023)

[26] Izacard, G., Lewis, P., Lomeli, M., Hosseini, L., Petroni, F., Schick, T., Dwivedi-Yu, J., Joulin, A., Riedel, S., Grave, E.: Few-shot learning with retrieval augmented language models. arXiv preprint arXiv:2208.03299 (2022)

[27] Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: Proceedings of the International Conference on Machine Learning (ICML) (2021)

[28] Joulin, A., Van Der Maaten, L., Jabri, A., Vasilache, N.: Learning visual features from large weakly supervised data. In: Proceedings of the European Conference on Computer Vision (ECCV) (2016)

[29] Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2015)

[30] Khandelwal, U., Levy, O., Jurafsky, D., Zettlemoyer, L., Lewis, M.: Generalization through memorization: Nearest neighbor language models. International Conference on Learning Representations (ICLR) (2020)

[31] Khosla, A., Jayadevaprakash, N., Yao, B., Fei-Fei, L.: Novel dataset for fine-grained image categorization. In: First Workshop on Fine-Grained Visual Categorization, CVPR (2011)

[32] Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: Proceedings of the International Conference on Computer Vision (ICCV) (2013)

[33] 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** (2017)

[34] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., et al.: Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in Neural Information Processing Systems (NeurIPS) (2020)

[35] Liang, F., Wu, B., Dai, X., Li, K., Zhao, Y., Zhang, H., Zhang, P., Vajda, P., Marculescu, D.: Open-vocabulary semantic segmentation with mask-adapted clip. arXiv preprint arXiv:2210.04150 (2022)

[36] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Proceedings of the European Conference on Computer Vision (ECCV) (2014)

[37] Liu, H., Son, K., Yang, J., Liu, C., Gao, J., Lee, Y.J., Li, C.: Learning customized visual models with retrieval-augmented knowledge. arXiv preprint arXiv:2301.07094 (2023)

[38] Long, A., Yin, W., Ajanthan, T., Nguyen, V., Purkait, P., Garg, R., Blair, A., Shen, C., van den Hengel, A.: Retrieval augmented classification for long-tail visual recognition. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[39] Maji, S., Rahtu, E., Kannala, J., Blaschko, M., Vedaldi, A.: Fine-grained visual classification of aircraft. arXiv preprint arXiv:1306.5151 (2013)

[40] Marino, K., Rastegari, M., Farhadi, A., Mottaghi, R.: Ok-vqa: A visual question answering benchmark requiring external knowledge. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

[41] Meyer, C.M., Gurevych, I.: Wiktionary: A new rival for expert-built lexicons? Exploring the possibilities of collaborative lexicography. na (2012)

[42] Miech, A., Alayrac, J.B., Smaira, L., Laptev, I., Sivic, J., Zisserman, A.: End-to-end learning of visual representations from uncurated instructional videos. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[43] Miller, G.A.: WordNet: An electronic lexical database. MIT press (1998)

[44] 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 with vision transformers. arXiv preprint arXiv:2205.06230 (2022)

[45] Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing (2008)

[46] van den Oord, A., Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748 (2018)

[47] Pham, H., Dai, Z., Ghiasi, G., Kawaguchi, K., Liu, H., Yu, A.W., Yu, J., Chen, Y.T., Luong, M.T., Wu, Y., et al.: Combined scaling for open-vocabulary image classification. arXiv preprint arXiv:2111.10050 (2021)

[48] Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In: Proceedings of the International Conference on Computer Vision (ICCV) (2015)

[49] Prabhu, V.U., Birhane, A.: Large image datasets: A pyrrhic win for computer vision? arXiv preprint arXiv:2006.16923 (2020)

[50] 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: Proceedings of the International Conference on Machine Learning (ICML) (2021)

[51] Ridnik, T., Ben-Baruch, E., Noy, A., Zelnik-Manor, L.: Imagenet-21k pretraining for the masses. arXiv preprint arXiv:2104.10972 (2021)

[52] 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 (2015)

[53] Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., Komatsuzaki, A.: Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. arXiv preprint arXiv:2111.02114 (2021)

[54] Shen, S., Li, C., Hu, X., Xie, Y., Yang, J., Zhang, P., Gan, Z., Wang, L., Yuan, L., Liu, C., et al.: K-lite: Learning transferable visual models with external knowledge. Advances in Neural Information Processing Systems (NeurIPS) (2022)

[55] Singh, A., Hu, R., Goswami, V., Couairon, G., Galuba, W., Rohrbach, M., Kiela, D.: Flava: A foundational language and vision alignment model. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[56] Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., Rohrbach, M.: Towards vqa models that can read. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2019)

[57] Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., Belongie, S.: The inaturalist species classification and detection dataset. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2018)

[58] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in Neural Information Processing Systems (NeurIPS) (2017)

[59] Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011)

[60] Wang, S., Xu, Y., Fang, Y., Liu, Y., Sun, S., Xu, R., Zhu, C., Zeng, M.: Training data is more valuable than you think: A simple and effective method by retrieving from training data. arXiv preprint arXiv:2203.08773 (2022)

[61] Weyand, T., Araujo, A., Cao, B., Sim, J.: Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2020)

[62] Wu, Y., Rabe, M.N., Hutchins, D., Szegedy, C.: Memorizing transformers. In: ICLR (2022)

[63] Xian, Y., Lampert, C.H., Schiele, B., Akata, Z.: Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2018)

[64] Xiao, J., Hays, J., Ehinger, K.A., Oliva, A., Torralba, A.: Sun database: Large-scale scene recognition from abbey to zoo. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2010)

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

[66] Zhai, X., Wang, X., Mustafa, B., Steiner, A., Keysers, D., Kolesnikov, A., Beyer, L.: Lit: Zero-shot transfer with locked-image text tuning. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2022)

[67] Zhang, Y., Jiang, H., Miura, Y., Manning, C.D., Langlotz, C.P.: Contrastive learning of medical visual representations from paired images and text. In: Machine Learning for Healthcare Conference (2022)

[68] Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)

[69] Zhu, Y., Groth, O., Bernstein, M., Fei-Fei, L.: Visual7w: Grounded question answering in images. In: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR) (2016)

6 Appendix

A Additional results

A.1 Using LAION-400M as the memory

In Table 5, we show that our method also works when using a public dataset as the memory bank instead of our private source of knowledge. Indeed, we observe that using LAION-400M [53] as the memory bank for RECO gives substantial gains of performance compared to the CLIP baseline across our different zero-shot tasks: for example +6.9 on Cars and +6.1 on CUB. This validates that our method is generic and can work with different choices of external knowledge.

Table 5: **Choice of memory bank.** We report zero-shot top-1 accuracy on different image classification tasks. We evaluate RECO when using two different sources of knowledge: the non publicly available WebLI [8] dataset (our default) and the publicly available LAION-400M [53] dataset.

Memory bank	Publicly available	Cars	CUB	Flowers	Im1k	Pl365
None	–	57.2	52.8	62.1	63.5	40.6
WebLI [8] (default)	✗	68.1 (+10.9)	63.0 (+10.2)	67.9 (+5.8)	64.6 (+1.1)	42.2 (+1.6)
LAION-400M [53]	✓	64.1 (+6.9)	58.9 (+6.1)	63.7 (+1.6)	63.4 (-0.1)	42.3 (+1.7)

A.2 More complex fusion module

In Table 6, we experiment with fusion modules of varying sizes. We observe that using a fusion module of one or two layers works comparatively well. However, using larger fusion modules with more layers, *e.g.* four, six or eight, deteriorates the performance. We hypothesize that this is because increasing the capacity of the fusion creates overfitting. Overall, using a single-layer fusion module brings large gains of performance on top of CLIP while being very light-weight to train.

Table 6: **Size of the fusion module.** For each variant, we report the total number of parameters ("# params") in millions and the percentage of the total parameter count which is part of the fusion modules ("% fusion params"). We report zero-shot top-1 accuracy on three image classification tasks.

# fusion layer	# params (M)	% fusion params	Cars	Flowers	Im1k
0	151.3	0%	57.2	62.1	63.5
1 (default)	154.4	2.0%	**68.1**	67.9	**64.6**
2	157.6	4.0%	67.8	**69.1**	64.3
4	163.9	7.7%	61.9	62.8	59.8
6	170.2	11.1%	60.6	64.5	59.6
8	176.5	14.3%	61.1	64.4	59.6

A.3 End-to-end finetuning versus frozen backbone

In Table 7, we evaluate the behavior of RECO when *finetuning* the original encoders at the same time as the fusion module. We observe in Table 7 that the performance is comparable to keeping the encoders frozen as in our default setting. Freezing the encoders has the advantage of requiring less compute resources. In our implementation and using the same hardware, finetuning CLIP-B/32 along with the fusion module has a training step $1.6\times$ longer than working with frozen CLIP-B/32 and training only the fusion module.

A.4 Qualitative examples

Zero-shot retrieval. In Figure 5, we show some qualitative examples when applying RECO to zero-shot retrieval tasks on COCO dataset. Specifically, we aim to gain an understanding about why the model benefits more from augmenting the text rather than the image side when applied to retrieval downstream tasks (see Table 4 of the main paper). In Figure 5, we look at three different image-text input pairs from the validation set of COCO and display the retrieved captions fused with the query image as well as the retrieved images fused with the query text.

Table 7: **Full finetuning versus frozen encoders.** Both mechanisms produce good performance but freezing the backbone allows for $1.6\times$ faster training. We report zero-shot top-1 accuracy.

CLIP encoders f	Cars	Flowers	Im1k
Frozen (default)	68.1	**67.9**	**64.6**
Finetuned	**68.5**	67.1	64.2

We observe in Figure 5 that the text captions of an image in COCO retrieval task usually focus on one specific aspect of the image (for example the towel in the image of the bathroom or the British flag on the train). We observe that the retrieved images from the input text are likely to also contain this particular aspect of interest and hence match well with the original caption. For example, the retrieved images from the train with a British flag caption (bottom of Figure 5) all contain representations of vehicles with painted British flags, which is more likely to help the alignment with the original input.

On the contrary, the captions retrieved from the original image may focus on another particularity of the image, not mentioned in the original caption. For example, the captions retrieved from the train image contain information about train numbers, train station locations or operators. This information is not useful for this task, because the ground-truth caption focuses on the fact that the train carriage has a British flag on its side. This brings distracting signal instead of helping the alignment. Overall, we think these qualitative examples help us understand why disabling retrieval for the image input and enabling it for the text side results in a better performance in this task.

Zero-shot image classification. In Figure 6, we display some qualitative examples of RECO for zero-shot image classification downstream tasks. Specifically, we consider several query images and their class names and show the corresponding elements (captions for query image and images for query class name) retrieved by our model. We observe in Figure 6 that in majority of cases, given a visual input, searching for similar images and retrieving their corresponding captions effectively returns descriptions containing useful information for fine-grained classification problem. For example, the captions retrieved from the cat image at the top of Figure 6 contain the breed of that cat (siamese). Likewise, given the class name "siamese cat", RECO look for similar captions, for example "picture of a siamese cat", and returns their corresponding images. These all contain visual examples of what a siamese cat looks like. Figure 6 show several successful examples of this mechanism and helps giving intuition about why RECO helps for zero-shot image classification.

Interestingly, we observe some failure cases when retrieving from a query class name which is ambiguous in the sense that it can refer to several things. For example, the retrieved images from the class name "prince of wales feathers" in Flowers dataset returns non useful information such as the emblem of prince of wales or a picture of a feather. This is because "prince of wales feathers" can refer to many things other than a flower species. We observe this behavior for several class names of the Flowers classification benchmark which have a meaning outside of the flower species they refer to; for example "bird of paradise", or "bishop of llandaff" where one of the retrieved image is from the actual person who used to be the Bishop of Llandaff, a community in Wales.

B Evaluation details

B.1 Evaluation datasets

We report the details of each image classification dataset that we use to evaluate our model. Note that we only use test or validation splits of each of these datasets, training sets are disregarded. Stanford Cars ("Cars") [32] contains $8,041$ test images of 196 fine-grained car classes. Each class name consists of make, model and year of a car, *e.g. 2012 Tesla Model S* or *2012 BMW M3 coupe*. CUB-200-2011 ("CUB") [59] consists of $5,794$ test images of 200 bird species. The dataset contains additional annotations, such as part locations, binary attibutes, and bounding boxes, but we do not use any of them. Oxford Flowers ("Flowers") [45] has $6,149$ test images of 102 flower categories commonly found in the United Kingdom. The dataset images contain strong visual variations, such as scale, pose, and light changes. ImageNet-1k ("Im1k") [52], or also referred to as ILSVRC 2012, contains $50,000$ validation images from $1,000$ classes. Class names are obtained from the synsets in WordNet [43], and come from a large variety of generic concepts. Places365 ("Pl365") [68] has $36,500$ validation images from 365 generic scene categories. Some examples of class names include

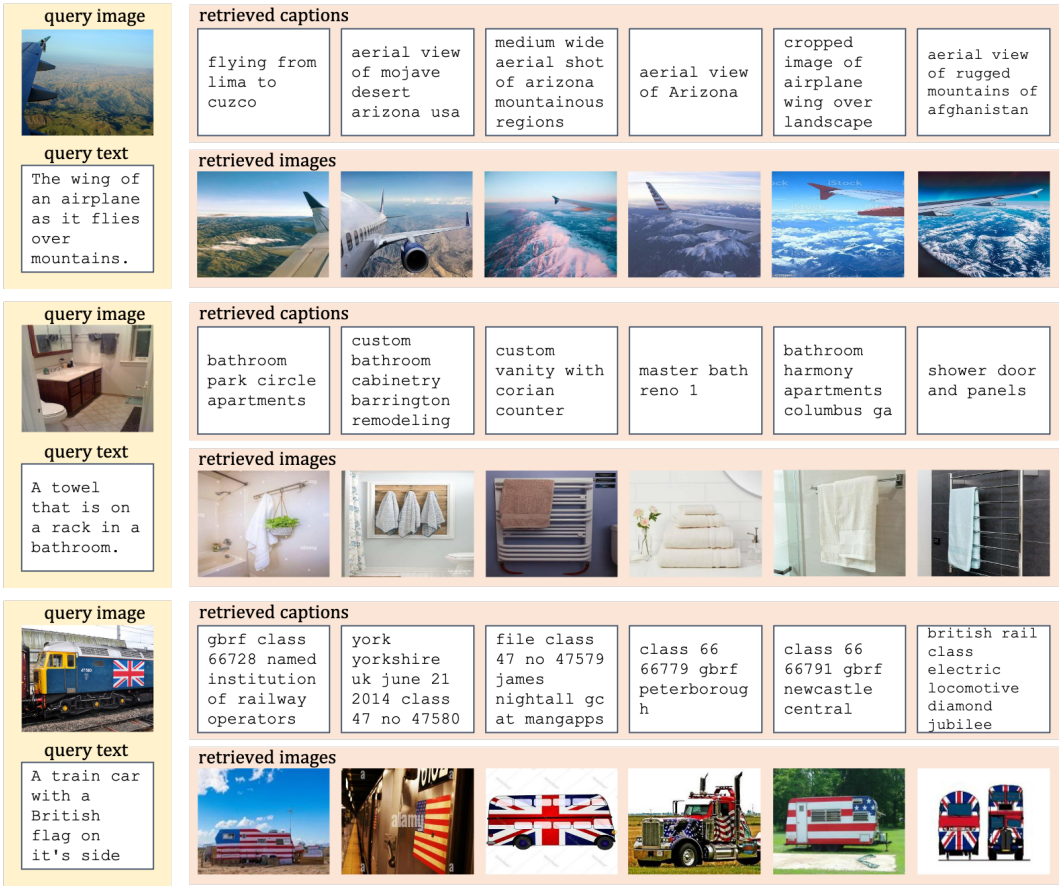

Figure 5: **Qualitative examples of RECO for image and text retrieval.** We display image and text queries on the left panel and retrieved captions and images on the right panel. We observe that retrieved images tend to match better with the input original image than retrieved captions with the input original text. For example, the retrieved captions from the aerial view do not mention a lot "mountains" while this is present in the original text. Instead, they mention many specific locations, for example lima, cuzco, arizona or afghanistan, which are not relevant to the original text description. On the contrary, the retrieved images from the text query are semantically similar to the original image. This qualitatively explains why the best of performance of RECO for zero-shot retrieval is achieved by disabling retrieval on the query image and enabling it on the query text (see Table 4 of the main paper).

living room, cottage, lecture room, pier etc. Finally, Stanford Dogs ("Dogs") [31] consists of $8,580$ test images from 120 dog breeds.

We use two datasets for our retrieval experiments. Flickr30k ("Flickr") [48] contains 1000 image-text pairs. Each image contains 5 sentence-level descriptions, or captions. Similarly, MS COCO ("COCO") [36] test set, as defined by Karpathy and Li [29], contains $5,000$ image-text pairs, where each image contains 5 captions. We report the performance for text-to-image ("T→I") and image-to-text ("I→T") retrieval on both datasets.

B.2 Evaluation protocols

Zero-shot image classification. We follow the standard setup [50] of embedding each class name with the text encoder. We classify an image to the class which has the highest cosine similarity between the image embedding and the corresping class name embedding. We report top-1 accuracy in all the image classification benchmarks. There is no variance at zero-shot evaluation time since the inference for both text and vision encoders are deterministic.

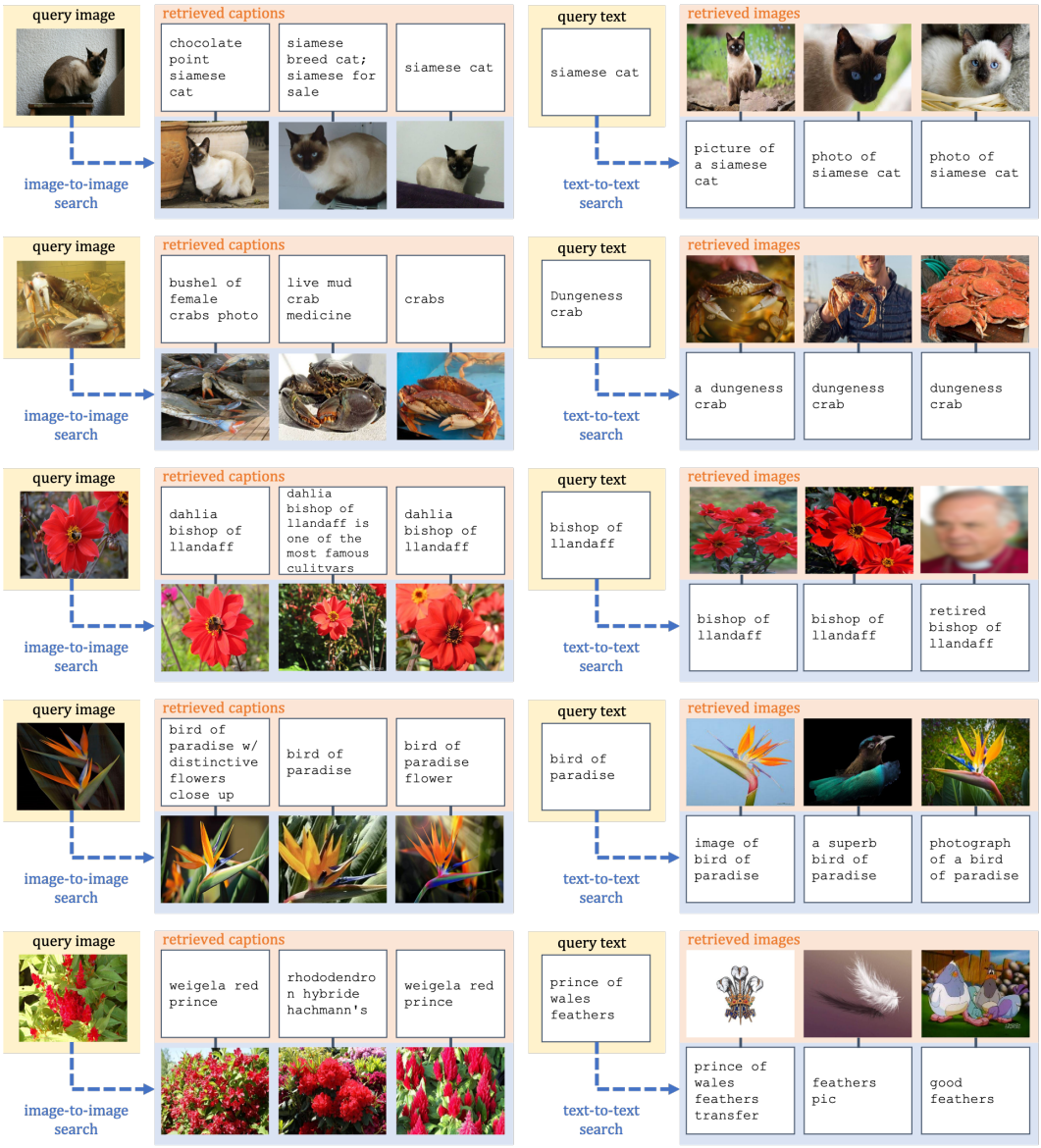

Figure 6: **Qualitative examples of RECO for image classification.** We consider several query images and their class names ("query text") and show the retrieved items to be fused with them. For a given a class name, looking for similar captions and using their corresponding images usually returns relevant visual examples. For example, the images retrieved from the class name "dungeness crab" show examples of what dungeness crabs visually look like. However, when the class name can refer to several things, for example "bird of paradise" which is both a flower and a bird species, then the visual retrieved examples are not always relevant to the finegrained classification problem at hand.

Image and text retrieval. For image-to-text retrieval, given an input image, we rank all the text embeddings according to their similarity to this image embedding. We report the proportion of images that ranks the correct text within the first R positions as the recall@R. The process is the symetric for text-to-image retrieval by switching the role of text and image. We report recall@1 in all the retrieval tasks. There is no variance at zero-shot evaluation time since the inference for both text and vision encoders are deterministic.

Variance and error bars. We report the performance variance on our small CLIP-B/32 setting to make sure that observed gains are significant. We train RECO with CLIP-B/32 backbone 5 times with different random seeds. We perform the evaluation for each model separately and report the

Table 8: **Standard deviation of RECO results.** We run five RECO training, each one with a different random seed. We show zero-shot image classification and retrieval results, and their standard deviation across 5 runs.

	Image classification						T→I		I→T	
Method	Cars	CUB	Flowers	Im1k	Pl365	Dogs	COCO	Flickr	COCO	Flickr
CLIP-B/32	57.2	52.8	62.1	63.5	40.6	58.6	30.2	61.1	51.2	80.9
RECO	68.1 ±0.3	63.0 ±0.3	67.9 ±0.4	64.6 ±0.1	42.2 ±0.1	59.7 ±0.2	33.6 ±0.1	65.7 ±0.3	52.2 ±0.3	81.8 ±0.3

accuracy, averaged over 5 run, with the variance in Table 8. We observe that the standard deviation is small across 5 runs, always below 0.4 across all benchmarks.

OVEN benchmark. OVEN benchmark [23] is created by combining 14 existing datasets (ImageNet21k-P [51, 52], iNaturalist2017 [57], Cars196 [32], SUN397 [64], Food101 [5], Sports100 [16], Aircraft [39], Oxford Flowers [45], Google Landmarks v2 [61], and various VQA (visual question answering) datasets [16, 18, 33, 40, 56, 69]) and grounding their categories to Wikipedia entities. The benchmark consists of two splits. Entity Split measures the image recognition or retrieval capabilities of a model, whereas the Query Split is designed as a VQA task. We focus on the Entity Split in this paper.

The Entity Splits contains training, validation, and test splits. However, since we focus on zero-shot image classification in this paper, we ignore the training and validation splits, and evaluate our model (trained on CC12M as discussed in Sec. 4) directly on the test set. The test set contains $729,259$ examples from $20,549$ entities. Each example belongs to a single entity. Nevertheless, total number of candidate entities during inference is $6,084,494$, *i.e.* there are more than 6M distractor entities at inference.

Note that unlike other image classification datasets, each example is an image-text pair in OVEN. The so-called *intent* text accompanies each image, and clarifies the question at hand, *e.g. what is the model of this vehicle?* Similarly, each entity is also an image-text pair, containing the entity name and entity image. We simply follow the same protocol as other image classification datasets in this paper, and only consider the example image and entity name.

C Illustrative comparison of uni-/cross- modal search and uni-/cross- fusion

We give a conceptual comparison of uni-/cross- modal search and uni-/cross- fusion for an image input I in the paper. We now show in Figure 7 this comparison for a text input T.

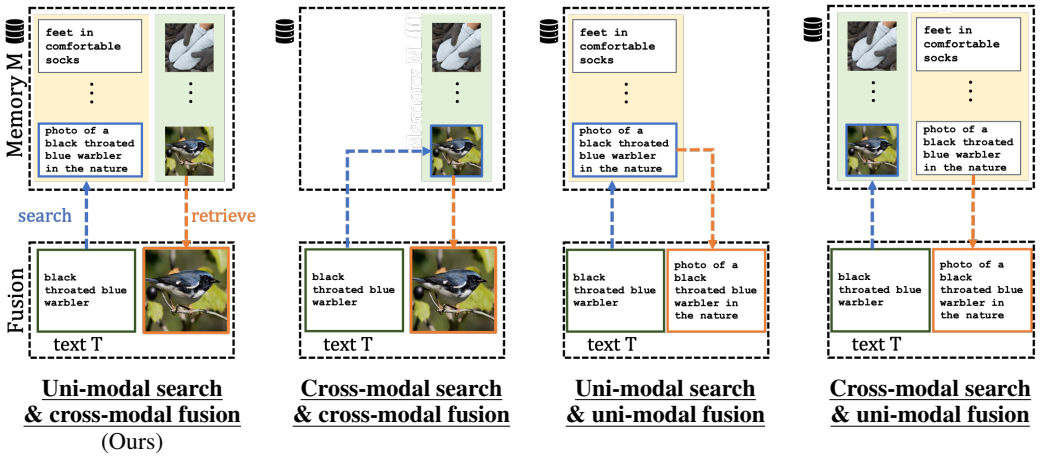

Figure 7: **Conceptual comparison of uni-/cross- modal search and uni-/cross- fusion.** We illustrate the different scenarios for a text input T while the scenarios for image input I are shown in the main paper.

D Broader Impact

We propose a retrieval-based recognition approach, where we search for similar images and text in a large-scale memory. Data retrieved from such uncurated sources may be biased against certain populations across the world [10, 49]. Furthermore, it is important that the privileged user data does not exist in such data collections, in order to avoid using the data without the consent of its owner. We acknowledge these potential misuses, and encourage the community to utilize more fair and responsible data collections.