Language-only Efficient Training of Zero-shot Composed Image Retrieval – Appendix –

Geonmo Gu*, 1 Sanghyuk Chun*, 2 Wonjae Kim[2] Yoohoon Kang[1] Sangdoo Yun[2]

[1]NAVER Vision [2]NAVER AI Lab * Equal contribution

A. Dataset Details

A.1. CIR datasets

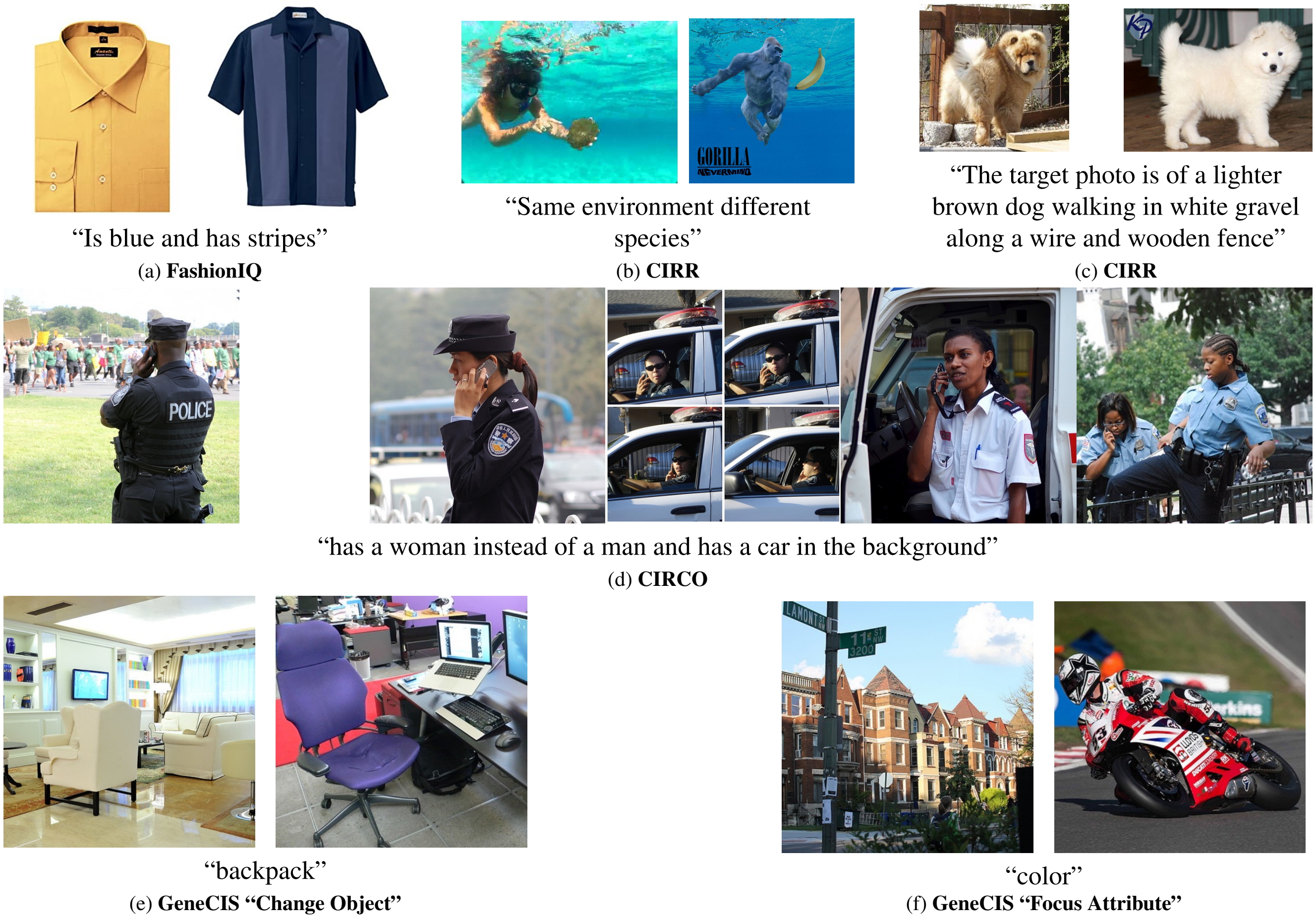
FasionIQ [22] is a collection of fashion-related images in three categories: Shirt, Dress, and Toptee. FashionIQ contains 30,134 triplets from 77,684 images. During the dataset collection period, FashionIQ first collects the attributes of all images and then lets human annotators write a proper caption for highly relative images in terms of attributes. FashionIQ targets a realistic online shopping chat window. Therefore, FashionIQ collects captions using a visual chat-based interface, resulting in containing more realistic user text queries. The images are split into 6:2:2 for training, validation, and evaluation. As we aim to zero-shot CIR, we did not use the training split. Following the previous practice, we report the validation recalls because the labels of the evaluation split have not been publicly released. Examples of FashionIQ triplets are shown in Fig. A.1a.

CIRR [12] contains 21,552 real-life images sampled from NLVR^2 [18]. CIRR also has training, validation, and test splits, where the test split is separately evaluated via the remote evaluation server. Therefore, we use the validation split of CIRR as the model selection criteria. While FashionIQ is limited to fashion-related domains, CIRR images are in more vast domains and have complex descriptions. During the dataset collection, CIRR collects visually similar images using ResNet-152 [8] trained on ImageNet [15] and repeats the caption collection stage of FashionIQ. However, despite the realistic image domain, CIRR has two significant issues. First, while FashionIQ carefully collects the image pairs with human annotators, the pairs collected by CIRR are automatically collected with ImageNet-trained ResNet without careful human verification. It makes the pairs not actually visually similar and the paired images often significantly differ from each other. Second, while FashionIQ collects captions by letting the annotators mimic realistic online customers, CIRR lets the annotators write captions describing the differences between the images. Due to this reason, CIRR captions are unrealistic but contain unnecessary information or ambiguous descriptions, such as "same environment different species" (Fig. A.1b) or "The target photo is of a lighter brown dog walking in white gravel along a wire and wooden fence" (Fig. A.1c). For these reasons, CIRR is not realistic compared to FashionIQ. To resolve the issue, CIRR employs a retrieval task on a small subset, *e.g.*, five items, but Baldrati et al. [1] observed that the subset retrieval task can be noisy because information of the target image is often not related to the reference image, but only related to the text condition. Furthermore, as Baldrati et al. [1], Saito et al. [16] and Fig. B.1 observed, CIRR has a lot of false negatives (FNs) that can lead to wrong retrieval evaluation, as also shown in image-text cross-modal retrieval [3, 4].

Notably, both Fashion IQ and CIRR suffer from the FN problem; although the ground truth positive is one for each query, there could be multiple ground truths in the database. Furthermore, while FashionIQ collects triplets with careful verification, CIRR is constructed with noisy annotations. To tackle the issue, FashionIQ is evaluated by Recall@K with larger K (*e.g.*, 10 or 50) and CIRR employs a small subset retrieval task. However, these approaches cannot resolve the problem fundamentally.

CIRCO [1] is based on COCO images [11] and contains multiple ground truths. CIRCO has 4.53 average ground truth images per query, enabling a more reliable and robust mAP metric [4, 13]. As CIRCO is designed for evaluating ZS-CIR methods, CIRCO has no training split. Instead, CIRO has a validation split (220 queries) and a test split (800 queries), where the test split is evaluated by the remote evaluation server. Example triplets are shown in Fig. A.1d.

GeneCIS [20] consists of four conditional retrieval tasks: (1) focus on an attribute, (2) change an attribute, (3) focus on an object, and (4) change an object. GeneCIS focuses on defining the similarity in various notations, *e.g.*, the similarity can be defined in the object "with the same **car**" or the attribute "the same **color** as the car". The attribute tasks are built upon



"Is blue and has stripes"
(a) **FashionIQ**

"Same environment different species"
(b) **CIRR**

"The target photo is of a lighter brown dog walking in white gravel along a wire and wooden fence"
(c) **CIRR**

"has a woman instead of a man and has a car in the background"
(d) **CIRCO**

"backpack"
(e) **GeneCIS "Change Object"**

"color"
(f) **GeneCIS "Focus Attribute"**

Figure A.1. **CIR Dataset examples.** In all examples, the first image is the reference, and the right image is the target image with the given caption. For CIRCO, the left image is the query image, and the other four images are all ground truth images with the given text query.

VisualGenome [9] with in-the-wild visual attributes from VAW [14]. The object tasks are based on COCO [11] and its object classes. Each task has about 2,000 queries and the gallery size is only 15 ("Focus on an attribute" task has 10) to avoid the FN problem, but to ensure only one positive for each query, similar to the CIRR subset strategy. Here, the text query is given by the name of the attribute or the object, *e.g.*, "backpack" or "color". Example triplets are in Fig. A.1e and Fig. A.1f.

For all datasets, we use the same prompt for zero-shot composed image retrieval, namely, a photo of `[$]` that `[cond]` where `[cond]` is the given text condition. Example text conditions are shown in Fig. A.1.

A.2. Training corpora

We provide examples of training corpora in Tab. 10, including CC3M [17], StableDiffusion Probmpts (SDP)[1], OpenWebText [6], and COYO-700M [2]. For OpenWebText, we only show one example of its shortened version, where the full document contains 1,386 words and 7,681 characters. We observe that the captions of CC3M are too generalized, therefore it has a weakness to describe an image in details. On the other hand, COYO-700M is too detailed, *e.g.*, containing the exact product name, such as IPhone 6S. In practice, we use CC3M and SDP for representing a conceptualized caption for describing an image and detailed instruction for explaining an image by deep generative model, *i.e.*, StableDiffusion.

- **CC3M:** "person, was surprised by the staff", "red and white flag on the mast", "football player celebrates scoring for football team against football team in the final", "concept plug - in hybrid car on display", "a pencil drawing of a zebra and her baby.", "airline – reasons why person leads the way in experience"

[1] https://huggingface.co/datasets/FredZhang7/stable-diffusion-prompts-2.47M

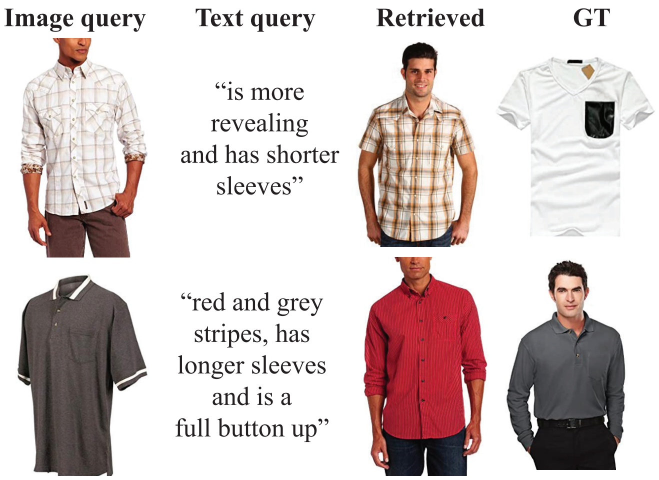


Image query
Text query
Retrieved
GT
“is more revealing and has shorter sleeves”
“red and grey stripes, has longer sleeves and is a full button up”

Figure B.1. **False negatives in CIR datasets.** Examples are drawn from FashionIQ [22], retrieved by LinCIR.

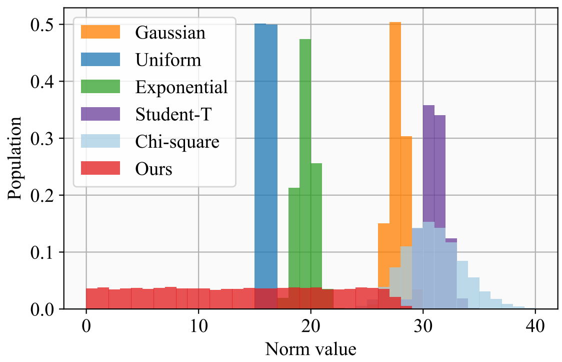


Figure B.2. **Norm distribution of different distributions.** The statistics are measured by 768-dim random vectors (the CLIP ViT-L/14 textual embedding dimension) drawn from six different probabilistic distributions. “Ours” denotes $\text{Unif}(0, 1) \times \mathcal{N}(0, 1)$.

- **SDP:** “a full body character design by artgerm, greg rutkowski and alphonse mucha. sci - fi dagger. white tape and red translucent plastic tape project show attctive showgirl!! sci - fi helmet!! sharp edges. contour light effect!!. ultra detailed, elegant, intricate, octane render.”, “realistic detailed face portrait of a beautiful young otherworldly ethereal alien geisha with blue hair by alphonse mucha, ayami kojima, yoshitaka amano, charlie bowater, karol bak, greg hildebrandt, jean delville, and mark brooks, art nouveau, neogothic, gothic, rich deep moody colors, celestial, surreal majestic winter pine dreamscape, character concept design”, “cute anthropomorphic guinea pig full as an jedi in a spaceship, body portrait, divine lightning, by greg rutkowski, by charlie bowater”, “boxing match between donald trump vs joe biden, stage lighting, award winning photo”, “a chibi anime warrior with long red hair quickly swinging her sword in a full arc, dynamic slashing pose, detailed, anime”, “saul goodman shaking hands with purple thanos at a walmart”, “a fuzzy pokemon:: by beeple and james gilleard and justin gerard :: ornate, dynamic, particulate, intricate, elegant, highly detailed, centered, artstation, smooth, sharp focus, octane render, 3”
- **OpenWebText:** “One family says the ratings-grabbing reality show ”Extreme Makeover: Home Edition” turned their personal tragedy into a practical nightmare, leaving them with virtually nothing but a lawsuit. . . . for more analysis and interviews on the top legal stories each weeknight at 6 p.m. ET on MSNBC TV.” (1,386 words 7,681 characters)
- **COYO-700M:** “Kidney humble nurse mascot design with a syringe”, “Load image into Gallery viewer, Trevor Medium Highland Cow”, “American Girl Doll Grace Thomas Goty Girl Of The Year 2015 + AG Outfit VGC”, “Lajama Luxury Tempered Glass Phone Case For IPhone 6 6S 7 8 Plus Anti-scratch Silicone Protector Glass Back Cover For Iphone X”

B. More experiments

B.1. Norm distributions of different probabilistic distributions

Fig. B.2 shows the distribution of the norms of the 768-dimensional random vectors drawn from each probability distribution. Here, we observe that the samples drawn from a Gaussian distribution have almost identical norm sizes. Similarly, other probability distributions, such as uniform, exponential, χ^2, and student-t distributions, suffer from the same problem. In other words, regardless of the actual gap between an image-text pair, a random noise sampled from such distributions always adds additional information to the textual latent embedding with an almost fixed amount regarding its norm. However, in practice, the gap between image-text pairs can be diverse. We presume this less diverse norm size of the added random noise restricts the generalizability against the modality gap. Fig. B.2 also illustrates that our design choice shows a more diverse norm distribution than the other distributions.

B.2. LinCIR with other backbones

Tab. B.1 shows the results of LinCIR with BLIP ViT-B/16 backbone [10]. Similar to CLIP, the BLIP ViT-B/16 backbone uses ViT-B/16 [5] as the image encoder and Transformer [19] as the text encoder. We use the BLIP encoders as the separated feature encoders as the CLIP encoders, instead of using the joint multimodal embedding extraction. The table shows that the BLIP backbone shows a comparable performance to the CLIP backbone, despite of using a smaller backbone.

	CIRCO mAP@5	GeneCIS R@3	FashionIQ R@10	CIRR R@10	Avg
CLIP ViT-L/14	12.59	32.38	26.28	66.68	34.48
BLIP ViT-B/16 [10]	12.75	29.65	25.00	65.35	33.19

Table B.1. **BLIP results.**

B.3. Comparison with more methods

In the main paper, we compare LinCIR with two ZS CIR methods: Pic2Word [16] and SEARLE [1]. In this subsection, we compare LinCIR with more recent CIR methods, such as CompoDiff [7] and CoVR [21]. The most significant drawback of CompoDiff and CoVR is a heavy computation. While LinCIR only needs 30 mins to train a high performing ZS CIR model, CompoDiff and CoVR take more than a day even with a ViT-L backbone.

CompoDiff tackles ZS CIR problem by generating massive 18.8M synthetic triplets and employing image feature editing strategy via a latent diffusion model. Despite its ability to handle negative text conditions and mask conditions, CompoDiff suffers from heavy computations. Tab. B.2 shows the full comparisons of CompoDiff and the other ZS CIR methods, including LinCIR. We can observe that CompoDiff needs a heavy GPU computations (about 10 days with 128 A100 for training) and a relatively slow inference time (0.12 vs. 0.02). Although CompoDiff has better flexibility than LinCIR, we do not directly compare CompoDiff with LinCIR due to its heavy resource computation.

		Training time1 (h)	Training time2 (h)	Total training time (h)	Inference time (s)	Training GPUs
ViT-L	Pic2Word	3.0	-	3.0	0.02	A100 x 8
	SEARLE	1.7	2.5	4.2	0.02	A100 x 8
	LinCIR	0.5	-	0.5	0.02	A100 x 8
	CompoDiff	6 days, 10hours	3 days, 5hours	9 days, 15hours	0.12	A100 x 128
ViT-H	Pic2Word	7.3	-	7.3	0.035	A100 x 8
	SEARLE	3.6	4.5	8.1	0.042	A100 x 8
	LinCIR	0.7	-	0.7	0.042	A100 x 8
ViT-G	Pic2Word	13.4	-	13.4	0.050	A100 x 8
	SEARLE	6.3	8.1	14.4	0.047	A100 x 8
	LinCIR	0.8	-	0.8	0.047	A100 x 8

Table B.2. **Training time and inference time comparisons.** Pic2Word and LinCIR is trained on a single stage, while SEARLE and CompoDiff need a two stage training strategy. The inference time is measured by a single A100 GPU with a single image.

Similarly, we do not directly compare LinCIR with CoVR because CoVR relies on the joint multi-modal encoder of BLIP [10] and a large-scale video dataset (1.6M triplets). We focus on a fair comparison of CIR methods that share the same embedding space (*i.e.*, the CLIP latent embedding space), without using a multimodal cross-attention Transformer taking both image and text inputs to compute a multimodal embedding. CoVR also uses a larger image resolution (384 pixels) than our comparison methods (224 pixels).

B.4. GeneCIS full results

Tab. B.3 shows the full results of the comparison methods on GeneCIS. Tab. B.4 shows the average scores for “Focus”, “Change”, “Attribute”, and “Object” tasks. In the table, we observe that LinCIR shows significantly better performances than others, especially for “Focus” tasks and “Attribute” tasks. We presume that our keyword masking strategy for SMP improves the ability to understand short keywords of GeneCIS, such as “color” in Fig. A.1f.

B.5. Exploring inference prompt

We also explore the impact of the ZS CIR prompt rather than “a photo of `[$]` that `[cond]`”. Tab. B.5 shows the results. In the paper, we observe that choosing a different prompt can significantly enhance retrieval performance. For example, by

		Focus Attribute			Change Attribute			Focus Object			Change Object			Average		
		R@1	R@2	R@3	R@1	R@2	R@3	R@1	R@2	R@3	R@1	R@2	R@3	R@1	R@2	R@3
ViT-L	Pic2Word	15.65	28.16	38.65	13.87	24.67	33.05	8.42	18.01	25.77	6.68	15.05	24.03	11.16	21.47	30.38
	SEARLE	17.00	29.65	40.70	16.38	25.28	34.14	7.76	16.68	25.31	7.91	16.84	25.05	**12.26**	22.11	31.30
	LinCIR	16.90	29.95	41.45	16.19	27.98	36.84	8.27	17.40	26.22	7.40	15.71	25.00	12.19	**22.76**	**32.38**
ViT-H	Pic2Word	18.60	30.70	42.10	13.16	23.91	33.14	9.23	17.60	27.14	6.58	16.48	25.36	11.89	22.17	31.94
	SEARLE	18.75	31.50	42.25	15.53	26.85	35.89	10.61	18.67	26.53	8.47	17.86	26.22	13.34	23.72	32.72
	LinCIR	19.60	31.50	41.55	16.62	27.60	37.50	9.80	18.83	27.86	9.03	17.55	25.71	**13.76**	**23.87**	**33.16**
ViT-G	Pic2Word	12.45	23.40	33.65	11.74	21.88	30.87	9.90	19.34	27.35	8.57	18.16	26.12	10.67	20.70	29.50
	SEARLE	16.30	29.40	40.70	16.15	27.32	35.46	10.77	18.16	27.91	8.27	15.56	25.77	12.87	22.61	32.46
	LinCIR	19.05	33.00	42.30	17.57	30.16	38.07	10.10	19.08	28.06	7.91	16.33	25.71	**13.66**	**24.64**	**33.54**

Table B.3. **GeneCIS full results.**

		"Focus" Avg			"Change" Avg			"Attribute" Avg			"Object" Avg		
		R@1	R@2	R@3	R@1	R@2	R@3	R@1	R@2	R@3	R@1	R@2	R@3
ViT-L	Pic2Word	14.76	26.42	35.85	7.55	16.53	24.90	12.04	23.09	32.21	10.28	19.86	28.54
	SEARLE	**16.69**	27.47	37.42	**7.84**	**16.76**	25.18	12.38	23.17	33.01	**12.15**	21.06	29.60
	LinCIR	16.55	**28.97**	**39.15**	**7.84**	16.56	**25.61**	**12.59**	**23.68**	**33.84**	11.80	**21.85**	**30.92**
ViT-H	Pic2Word	15.88	27.31	37.62	7.91	17.04	26.25	13.92	24.15	34.62	9.87	20.20	29.25
	SEARLE	17.14	29.18	39.07	**9.54**	**18.27**	26.38	14.68	25.09	34.39	12.00	22.36	31.06
	LinCIR	**18.11**	**29.55**	**39.53**	9.42	18.19	**26.79**	**14.70**	**25.17**	**34.71**	**12.83**	**22.58**	**31.61**
ViT-G	Pic2Word	12.10	22.64	32.26	9.24	**18.75**	26.74	11.18	21.37	30.50	10.16	20.02	28.50
	SEARLE	16.23	28.36	38.08	**9.52**	16.86	26.84	13.54	23.78	34.31	12.21	21.44	30.62
	LinCIR	**18.31**	**31.58**	**40.19**	9.01	17.71	**26.89**	**14.58**	**26.04**	**35.18**	**12.74**	**23.25**	**31.89**

Table B.4. **GeneCIS average results.**

changing "a photo of `[$]` that `[cond]`" to "Observe `[$]` that `[cond]`", LinCIR R@50 is improved by almost 2.0pp (46.48 to 48.41). On the other hand, we observe that Pic2Word is rarely improved by changing the prompt. SEARLE shows a reasonable improvement compared to Pic2Word but still performs worse than LinCIR.

C. Qualitative results

C.1. Retrieval from LAION

We compare qualitative retrieval results of the comparison methods on a million-scale search database using LAION-2B. Fig. C.1 shows the results using CLIP ViT-L features. We include more examples in `https://github.com/navervision/lincir`, including ViT-H features. In the figure, we observe that Pic2Word cannot handle both image and text conditions. For example, in the first example, Pic2Word ignores the crow visual information but only focuses on the "old man" text query. Similarly, the second example shows that Pic2Word ignores the visual information from the Eiffel Tower and the cat images but only focuses on climbing. SEARLE shows better results than Pic2Word, but as shown in the first example, SEARLE often attends to the visual information rather than the textual query. Interestingly, although our method is not trained on multiple query examples as the second and third examples, it shows reasonable retrieval results.

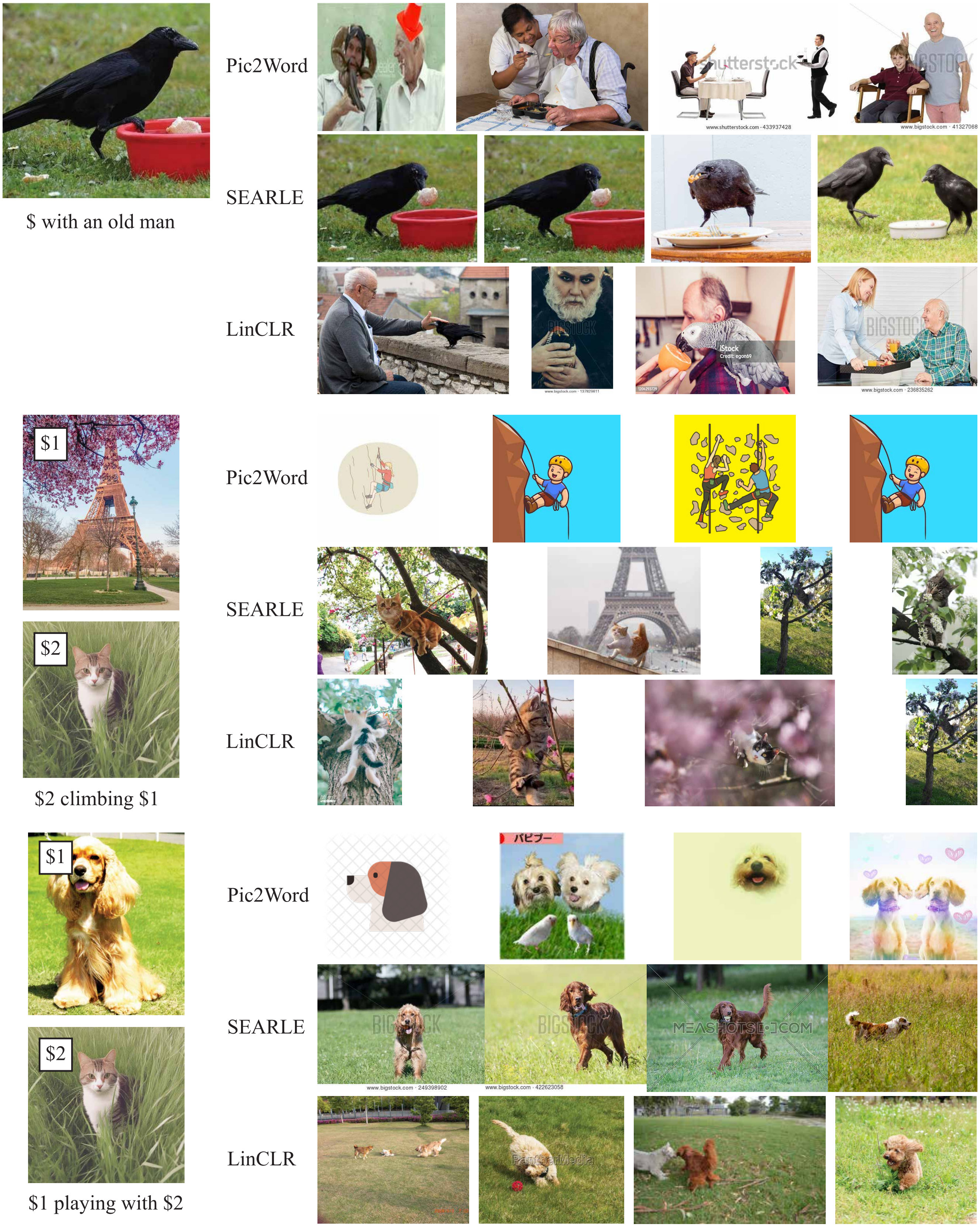


Figure C.1. **Retrieval results from LAION-2B CLIP-L features.** More examples are in `more_examples.pdf`.

	Pic2Word R@10	SEARLE R@10	LinCIR R@10	Pic2Word R@50	SEARLE R@50	LinCIR R@50
a photo of [$] that [cond]	**25.51**	24.64	26.28	44.98	44.41	46.48
[$] that [cond]	24.83	25.05	27.00	45.09	44.90	47.56
[$] with [cond]	24.16	25.08	26.99	43.03	44.77	47.62
[$], [cond]	25.20	24.39	27.08	43.93	44.40	47.87
[$] adapted to [cond]	23.31	24.92	26.36	42.82	43.97	47.63
[$] modified by [cond]	23.56	23.83	26.02	42.37	43.20	46.29
[$] in response to [cond]	23.31	25.10	26.71	41.90	44.82	47.31
[$] transformed by [cond]	24.02	24.10	26.45	42.81	43.69	46.52
[$] influenced by [cond]	21.30	23.52	26.52	39.76	43.16	47.28
Retrieval of [$] using feedback [cond]	21.01	22.13	24.82	38.98	40.99	44.96
[$] guided by [cond]	24.01	24.40	26.56	42.84	44.52	47.15
[$] adjusted to [cond]	24.05	24.63	26.78	43.68	44.55	47.67
[$] in alignment with [cond]	22.49	23.93	26.07	40.43	42.19	46.38
[$] in correspondence to [cond]	22.55	23.60	26.44	41.23	42.39	46.55
[$] refined with [cond]	22.91	23.58	26.66	41.59	43.17	46.89
[$] as directed by [cond]	23.28	25.59	26.96	42.17	45.30	47.70
[$] evolved from [cond]	24.64	23.12	26.77	44.02	42.63	47.26
[$] inspired by [cond]	24.35	24.18	26.26	43.71	43.61	47.05
[$] with adjustments from [cond]	23.76	24.11	26.26	42.11	44.64	46.92
[$] in consideration of [cond]	22.20	23.38	26.78	40.87	42.26	47.11
[$], taking into account [cond]	23.71	24.18	26.44	42.74	43.08	47.23
[$] as influenced by the query [cond]	21.45	22.92	25.86	39.72	43.04	45.72
[$] reshaped by [cond]	24.05	24.31	26.07	43.17	43.68	46.25
[$] curated based on [cond]	25.05	24.86	26.48	**45.36**	44.72	47.11
[$] showcasing [cond]	23.68	24.27	26.50	42.68	44.48	46.80
An instance of [$] where [cond]	23.32	24.50	25.71	42.09	43.24	45.84
[$] highlighting [cond]	24.45	22.34	26.19	42.98	41.58	46.40
A depiction of [$] exhibiting [cond]	22.26	23.42	25.53	41.97	42.79	45.27
[$] as exemplified by [cond]	23.58	24.95	26.35	42.98	44.89	46.78
[$] demonstrating [cond]	24.21	24.59	25.78	43.24	44.32	45.89
An illustration of [$] portraying [cond]	24.71	25.15	26.18	43.95	44.70	46.47
[$] in the context of [cond]	24.08	25.07	27.16	43.56	44.31	47.56
[$] as influenced by [cond]	21.30	24.37	26.61	40.22	43.97	46.89
[$] characterized by [cond]	24.21	23.40	26.64	44.04	42.28	46.49
[$]: An exploration of [cond]	23.51	25.52	26.25	42.66	45.45	46.59
A presentation of [$] underlined by [cond]	22.93	22.48	24.86	40.44	41.09	44.03
A manifestation of [$] reflecting [cond]	22.55	22.71	25.25	40.58	42.04	45.39
[$] in light of [cond]	22.62	25.03	26.41	41.70	44.71	46.99
[$] as a testament to [cond]	22.02	24.25	26.93	41.28	44.19	47.65
[$] intertwined with [cond]	24.45	23.64	25.06	43.92	42.37	45.15
[$] complemented by [cond]	25.23	24.21	26.32	45.04	43.23	47.07
[$] juxtaposed with [cond]	25.40	24.51	26.76	44.73	43.10	47.43
A representation of [$] in relation to [cond]	23.26	23.83	25.56	41.69	43.31	46.31
[$] that [cond]	24.83	25.05	27.00	45.09	44.90	47.56
[$] which [cond]	24.29	24.58	27.07	43.62	44.69	47.74
[$] where it [cond]	24.62	25.20	27.12	44.30	44.65	47.97
Discover [$] that [cond]	25.39	25.12	26.07	45.02	44.82	46.24
Retrieve [$] that [cond]	23.86	24.47	26.61	42.60	43.78	46.64
Search for [$] that [cond]	24.39	24.04	26.83	44.15	43.50	47.17
Identify [$] which [cond]	22.49	23.67	26.46	41.35	43.60	46.70
Highlight [$] that [cond]	24.53	23.45	26.01	43.89	42.72	46.65
Present [$] where it [cond]	24.12	24.40	27.29	43.58	43.66	47.94
Showcase [$] that [cond]	24.63	24.68	26.63	44.00	44.42	47.26
Explore [$] which [cond]	24.46	23.72	26.40	42.88	43.79	46.02
Find [$] that [cond]	24.65	25.03	27.28	44.97	44.44	47.73
Source [$] which [cond]	24.67	24.38	27.16	44.24	44.03	47.63
View [$] where it [cond]	24.06	24.96	27.16	43.56	43.93	47.99
Examine [$] that [cond]	25.10	24.53	26.64	44.89	43.94	47.65
Analyze [$] which [cond]	23.90	23.66	26.05	42.32	42.92	46.27
Observe [$] that [cond]	24.53	**26.50**	**27.42**	44.78	**45.83**	**48.41**
Report [$] which [cond]	23.91	23.69	26.48	42.92	43.11	47.04
See [$] where it [cond]	25.47	25.28	27.24	44.76	45.14	48.24
Document [$] that [cond]	24.49	24.52	26.44	44.31	44.58	47.17
Average performance	23.82	24.27	26.44	42.96	43.76	46.88
Best prompt	25.51	26.50	**27.42**	45.36	45.83	**48.41**

Table B.5. **FashionIQ R@10 and R@50 by varying text prompts.**

References

[1] Alberto Baldrati, Lorenzo Agnolucci, Marco Bertini, and Alberto Del Bimbo. Zero-shot composed image retrieval with textual inversion. In *ICCV*, 2023. 1, 4

[2] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. `https://github.com/kakaobrain/coyo-dataset`, 2022. 2

[3] Sanghyuk Chun. Improved probabilistic image-text representations. *arXiv preprint arXiv:2305.18171*, 2023. 1

[4] Sanghyuk Chun, Wonjae Kim, Song Park, Minsuk Chang Chang, and Seong Joon Oh. Eccv caption: Correcting false negatives by collecting machine-and-human-verified image-caption associations for ms-coco. In *ECCV*, 2022. 1

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3

[6] Aaron Gokaslan, Vanya Cohen, Ellie Pavlick, and Stefanie Tellex. Openwebtext corpus. `http://Skylion007.github.io/OpenWebTextCorpus`, 2019. 2

[7] Geonmo Gu, Sanghyuk Chun, HeeJae Jun, Yoohoon Kang, Wonjae Kim, and Sangdoo Yun. Compodiff: Versatile composed image retrieval with latent diffusion. *arXiv preprint arXiv:2303.11916*, 2023. 4

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1

[9] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123: 32–73, 2017. 2

[10] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, pages 12888–12900. PMLR, 2022. 3, 4

[11] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 1, 2

[12] Zheyuan Liu, Cristian Rodriguez-Opazo, Damien Teney, and Stephen Gould. Image retrieval on real-life images with pre-trained vision-and-language models. In *ICCV*, pages 2125–2134, 2021. 1

[13] Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pages 681–699. Springer, 2020. 1

[14] Khoi Pham, Kushal Kafle, Zhe Lin, Zhihong Ding, Scott Cohen, Quan Tran, and Abhinav Shrivastava. Learning to predict visual attributes in the wild. In *CVPR*, pages 13018–13028, 2021. 2

[15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115:211–252, 2015. 1

[16] Kuniaki Saito, Kihyuk Sohn, Xiang Zhang, Chun-Liang Li, Chen-Yu Lee, Kate Saenko, and Tomas Pfister. Pic2word: Mapping pictures to words for zero-shot composed image retrieval. In *CVPR*, 2023. 1, 4

[17] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 2

[18] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. *arXiv preprint arXiv:1811.00491*, 2018. 1

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3

[20] Sagar Vaze, Nicolas Carion, and Ishan Misra. Genecis: A benchmark for general conditional image similarity. In *CVPR*, 2023. 1

[21] Lucas Ventura, Antoine Yang, Cordelia Schmid, and Gül Varol. CoVR: Learning composed video retrieval from web video captions. *arXiv:2308.14746*, 2023. 4

[22] Hui Wu, Yupeng Gao, Xiaoxiao Guo, Ziad Al-Halah, Steven Rennie, Kristen Grauman, and Rogerio Feris. Fashion iq: A new dataset towards retrieving images by natural language feedback. In *CVPR*, pages 11307–11317, 2021. 1, 3