FLAIR: VLM with Fine-grained Language-informed Image Representations

Supplementary Material

In this supplementary file, we illustrate more qualitative results in Sec. A, describe the datasets in Sec. B, and present an extensive analysis of the impact of the negative pairs on the FLAIR performance in Sec. C. We further present additional ablation experiments in Sec. D, and the implementation details in Sec. E.

A. Qualitative Results

Attention Maps Visualization. We provide a comprehensive visualization of attention maps of $f_{\text{AttnPool}}(.)$ in Fig. 5 and Fig. 6. We follow DINO [3] to aggregate attention maps from multiple heads. We empirically found that heads 1,4,6,8 mainly focus on foreground objects and aggregate these attention maps to form the visualization. In Fig. 5, we show that the attention maps focus on different parts of an image w.r.t. the local captions. Interestingly, in the "fireplace" example (second row), the attention correctly localizes the "white candle" (second row, second column), which is exactly what the caption describes, although "fireplace" also appears in the sentence. This demonstrates that FLAIR is able to locate an object based on the main semantics of a prompt, instead of simply matching "a bag of words".

In Fig. 6, we visualize the attention maps w.r.t. long captions. When multiple objects appear in a long caption, FLAIR is able to locate them at the same time. Notably, in the "room" example (second row), FLAIR ignores descriptions like "adding a touch of nature to the room" and solely focus on the main semantics: "black shelf", "books" and "lamp". This might reveal one possible future application of FLAIR, understanding the main semantics in complex prompts and grounding the main objects in the image.

Token-to-Text Similarity. We also visualize the similarity between local image tokens and text prompts in Fig. 1 of the main paper. This similarity between the local image tokens and the text prompts could reflect the model's localization capability, which is closely related to the segmentation task. We provide extra visualizations in Fig. 7. We use FLAIR pre-trained on CC3M-recap to compare with DreamLIP [60] trained on Merged30M and OpenCLIP trained on DataComp-XL [16]. As illustrated, compared to OpenCLIP [6] that tends to make over-predictions, FLAIR is able to accurately localize the tokens w.r.t. the text prompts, especially on fine-grained details such as "flower on the cake" and "bird on the branch". This further validates that the fine-grained representations learned by FLAIR are indeed sensitive to the text semantics.

Retrieval Visualization. For the fine-grained image-text retrieval task on the DOCCI [37] benchmark, we visualize the top-5 retrieved captions for a given image, highlighting incorrect captions in red. We compare FLAIR with OpenCLIP [6] trained on 2B samples in Fig. 8. From top to bottom, the similarity scores decrease. Interestingly, compared to OpenCLIP [6], FLAIR tends to retrieve "local" captions first. For example, the top-1 retrieved caption for FLAIR is only describing the "spotlight", while OpenCLIP retrieves "a nighttime view of an artificial waterfall", which can be considered a global description for this image. The incorrectly retrieved captions of OpenCLIP contain relevant keywords like "waterfall", while FLAIR retrieves the captions

Figure 5. Visualization of the attention maps w.r.t. fine-grained captions. In the images, regions with high attention scores are marked in red; in the captions, objects representing the main semantics of the sentences are marked in red, while objects with less semantic significance are underlined.

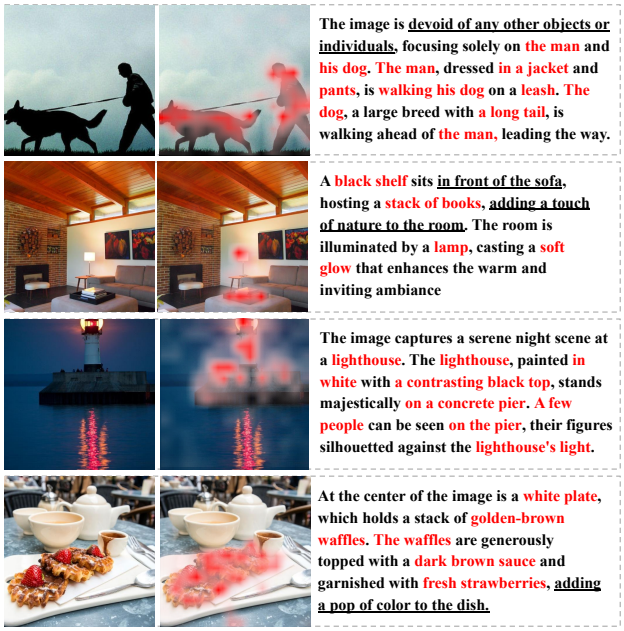

Figure 6. Visualization of the attention maps w.r.t. fine-grained long captions. In the images, regions with high attention scores are marked in red; in the cap tions, objects representing the main semantics of sentences are marked in red, while objects with less semantic significance are underlined.

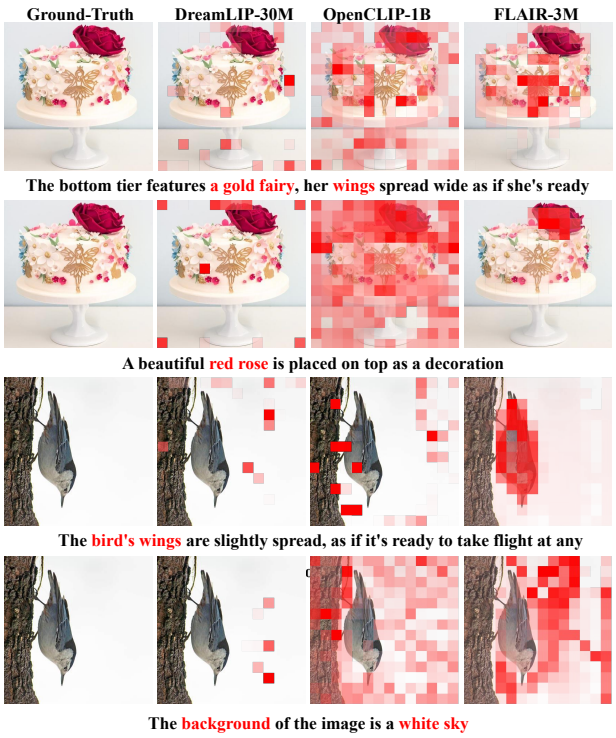

Figure 7. Visualization of the similarity scores between local image tokens and different text queries. While previous works [6, 60] lack fine-grained alignment, FLAIR matches text and image semantics at the token level.

correctly based on a more detailed understanding of the image semantics.

B. Dataset Details

Pre-training Data. FLAIR is pre-trained on CC3M-recap, CC12M-recap, YFCC15M-recap and Merged-30M [60], where each image is equipped with long synthetic captions generated by various MLLMs. Fig. 9 shows an example of the original long captions produced by DreamLIP [60] together with our diverse sampled captions. We take the whole paragraph of the long synthetic caption and split it into sentences. Our K diverse captions are sampled from these sentences, and each caption can contain $s \in \{1, ..., S\}$ merged sentences. In our experiments, we set $S = 3$ and $K = 8$. We detail this choice in Sec. D.4 and Sec. D.3.

Fine-grained Retrieval Data. In order to create the new fine-grained retrieval task, we split the original long captions from DOCCI [37] and IIW [19] into separate sentences. Each sentence can either describe the image globally or describe the fine-grained details of an image. These captions, together with the original images, form our DOCCI-FG and IIW-FG retrieval benchmarks. We provide a visualization of DOCCI-FG containing two images with all the corresponding paired captions in Fig. 10. As illustrated in Fig. 10, the split captions are likely to describe one local part of an image, such as "The wings and chest of the hawk are dark brown, and the left side of it is lit up by white light". We provide detailed statistics on the number of images, captions, and the average number of tokens per caption for standard, fine-grained, and long retrieval benchmarks in Tab. 6. DOCCI-FG and IIW-FG have an average of 7.1 and 10.1 captions per image, respectively, while each caption comprising approximately 18.76 and 22.56 tokens.

C. Extended Analysis of Negatives

As discussed in the methodology section of the main paper, FLAIR produces a unique image representation for each image-text pair using the text-conditioned attention pooling. Specifically, the text-conditioned embedding $\mathbf{v}^{\text{tc}}$ is jointly conditioned by the local image tokens $\mathbf{v}^{\text{loc}}$ and global text tokens $\mathbf{t}^{\text{g}}$:

$$\mathbf{v}^{\text{tc}} = f_{\text{AttnPool}}(\mathbf{v}^{\text{loc}}, \mathbf{t}^{\text{g}})$$

When considering the global text token $\mathbf{t}^{\text{g}}$, which forms both positive and negative pairs in $\mathcal{L}^{\text{tcs}}$, one positive pair ($\langle \mathbf{v}^{\text{tc}}_{i,i_k}, \mathbf{t}^{\text{g}}_{i_k} \rangle$) and five types of negative pairs can be identified. As visually depicted in Fig. 11, these negatives are:

$$\langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{j_k} \rangle, \langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{i_k} \rangle, \langle \mathbf{v}^{\text{tc}}_{i,i_k}, \mathbf{t}^{\text{g}}_{j_k} \rangle, \langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{l_k} \rangle, \langle \mathbf{v}^{\text{tc}}_{i,i_k}, \mathbf{t}^{\text{g}}_{i_m} \rangle$$

Figure 8. Visualization of image-to-text retrieval samples on the DOCCI-FG [37] benchmark, comparing FLAIR with OpenCLIP [6]. For each image, the top-5 retrieved captions are displayed. The incorrect retrieved captions are marked in red. The top-1 retrieved captions are **bold**.

Dataset	#Images	#Captions	#Captions per Image	#Tokens per Caption
Standard Text-image Retrieval Dataset				
MSCOCO [30]	5,000	25,000	5.0	11.77
Flickr30K [40]	1,000	5,000	5.0	14.03
Fine-grained Text-image Retrieval Dataset				
DOCCI-FG [37]	5,000	35,533	7.1	18.76
IIW-FG [19]	612	6204	10.1	22.56
Long Text-image Retrieval Dataset				
DCI [49]	7,805	7,805	1.0	172.73
IIW [19]	612	612	1.0	239.73
SV-1k [5]	1,000	1,000	1.0	173.24
SV-10k [5]	10,000	10,000	1.0	173.66

Table 6. Dataset details of the standard, fine-grained and long image-text retrieval task. SV-1K and SV-10K denote the 1K and 10K subset from the ShareGPT4V [5] dataset. Values of long text-image retrieval are directly obtained from [51], since we follow their evaluation setting.

The notation $\{i, j, l\}$ indicates that this pair is constructed from the {Image, Text Condition, Text}, which stems from the $\{i$-th, j-th, l-th$\}$ image separately, while k represents the k-th caption for image i. The pair $\langle \mathbf{v}^{\text{tc}}_{i,i_k}, \mathbf{t}^{\text{g}}_{i_m} \rangle$ is unique, as it arises from the k-th and m-th captions of the same image.

Empirical Comparison. By introducing text-conditioned attention pooling for multi-caption settings, FLAIR considers one positive and up to five distinct negative pairings. Modeling all five negatives simultaneously causes significant computational overhead. Thus, we investigate the importance of each negative type. To study their effects, we conducted a comprehensive ablation experiment (Tab. 7). For each setup, we trained FLAIR with one positive and only one negative pairing at a time, using a batch size of 1,024. All models were trained on the CC3M-recap [60] dataset for 10 epochs. To evaluate training dynamics, we analyzed the training loss ($\mathcal{L}_{\text{train}}$) and validation performance using the MSCOCO retrieval task. Key findings include: 1. The negative $\langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{l_k} \rangle$ suffers from high $\mathcal{L}_{\text{train}}$ and poor validation performance. As this negative spans across three different source images, it likely introduces noise rather than aiding learning. 2. The negatives $\langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{i_k} \rangle$ and $\langle \mathbf{v}^{\text{tc}}_{i,i_k}, \mathbf{t}^{\text{g}}_{i_m} \rangle$ converge quickly during training, but their $\mathcal{L}_{\text{train}}$ swiftly drops to nearly zero. Their evaluation on MSCOCO reveals poor performance, suggesting the existence of shortcuts. For $\langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{i_k} \rangle$, the model likely ignores image information and relies solely on text conditioning, thus failing in evaluation, when image information is vital. 3. The negative $\langle \mathbf{v}^{\text{tc}}_{i,i_k}, \mathbf{t}^{\text{g}}_{j_k} \rangle$ converges to a reasonable $\mathcal{L}_{\text{train}}$, but its performance (2.4% R@1 on T2I) indicates limited learning benefit. 4. The negative $\langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{j_k} \rangle$, currently used in FLAIR, reaches the best retrieval results, demonstrating its effectiveness.

Neg.	$\mathcal{L}_{\text{train}}$	T2I@1	T2I@5	I2T@1	I2T@5
$\langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{l_k} \rangle$	5.8	0.0	0.1	0.0	0.1
$\langle \mathbf{v}^{\text{tc}}_{i,i_k}, \mathbf{t}^{\text{g}}_{i_m} \rangle$	0.0	0.0	0.1	0.0	0.0
$\langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{i_k} \rangle$	0.0	0.0	0.1	0.0	0.0
$\langle \mathbf{v}^{\text{tc}}_{i,i_k}, \mathbf{t}^{\text{g}}_{j_k} \rangle$	1.53	2.4	7.8	0.3	1.2
$\langle \mathbf{v}^{\text{tc}}_{i,j_k}, \mathbf{t}^{\text{g}}_{j_k} \rangle$	0.68	**24.5**	**49.1**	**36.4**	**62.7**

Table 7. Retrieval performance of FLAIR on the MSCOCO [30] validation set when trained with different negative types on the CC3M-recap [60] dataset for 10 epochs. All models use ViT-B/16 as vision encoder. The best retrieval results are **bold**.

D. Additional Ablation Experiments

Aside from the main ablation study on the components of FLAIR described in the main paper, we conduct additional experiments to validate specific design choices. These include pre-training FLAIR on different data sources (Sec. D.1), comparing the diverse sampling strategy with a fixed merging strategy (Sec. D.2), ablating the maximum number of sampled sentences S (Sec. D.3), and examining how the number of sampled captions K affects the performance (Sec. D.4).

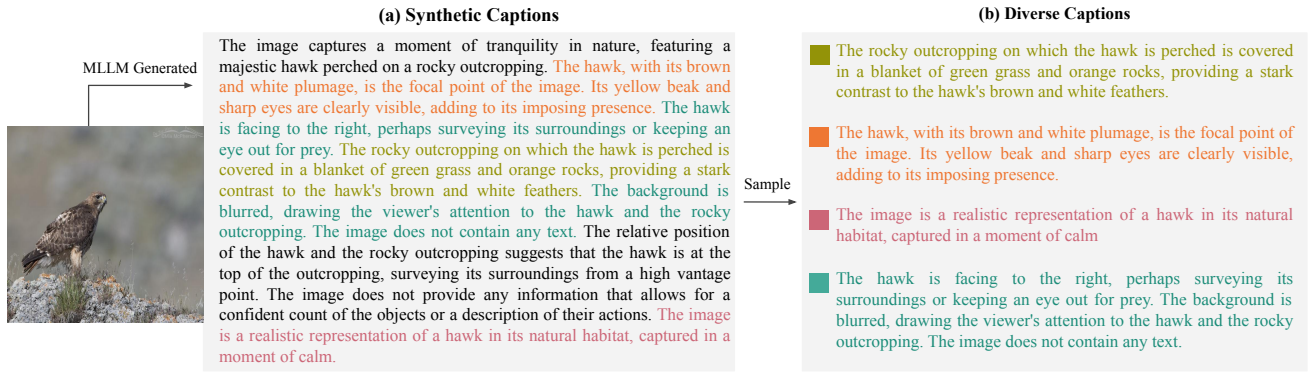

Figure 9. Examples of our diverse captions. Image and captions are taken from CC3M-recap [60]. Given the synthetic long captions generated by an MLLM, we sample $K = 4$ sub-captions where each sub-caption consists of $s \in \{1, 2, 3\}$ sentences. In our main experiments, we use $K = 8$.

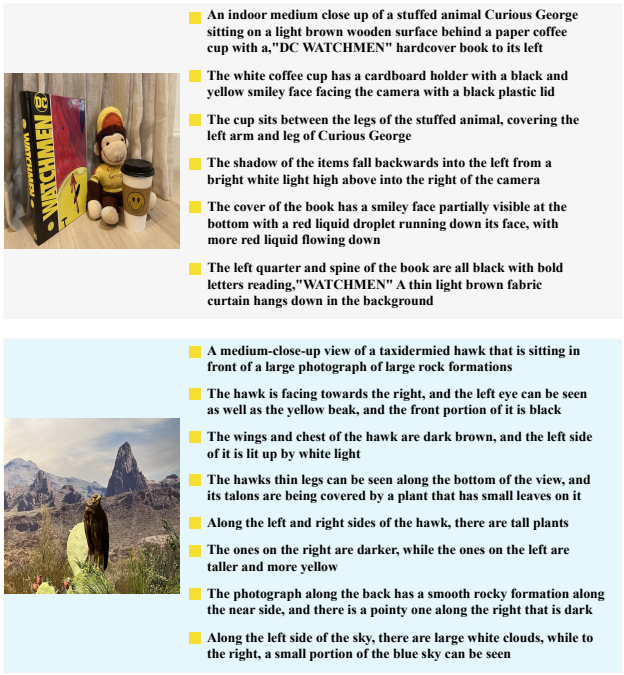

Figure 10. Dataset samples from DOCCI-FG [37] for the fine-grained retrieval task. For each image, we split the long caption into individual sentences each serving as a positive image-text pair for the benchmark.

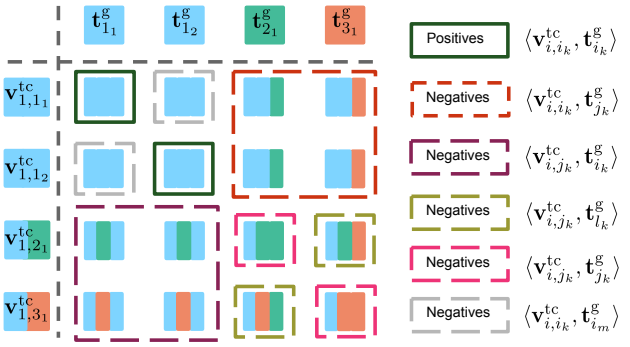

Figure 11. Illustration of all possible positive and negative pairs for FLAIR.

D.1. Pre-training on Different Data Sources

To demonstrate that our model is not limited to data curated by DreamLIP [60], we also pre-train FLAIR on the original CC3M [45] (CC3M-orig) and PixelProse [46]. For CC3M-orig and PixelProse, we use the same pre-training configurations as CC3M-recap and CC12M-recap, respectively. Detailed configurations are available in Sec. E. CC3M-orig contains one conceptual caption per image, while PixelProse re-captioned 15M images from CC12M [4], RedCaps [11], and CommonPool [16] using Gemini-Pro [42]. Unlike DreamLIP, which uses three MLLMs for re-captioning, PixelProse employs a single MLLM, resulting in shorter captions.

We evaluate FLAIR on the standard retrieval task and compare its performance to CLIP [41] trained on the same datasets. The results are summarized in Tab. 8.

As shown in Tab. 8, even when pre-trained on CC3M-orig, where FLAIR cannot leverage additional augmented captions, it still achieves a 2% improvement over CLIP in terms of R@1 on the MSCOCO dataset [30]. This demonstrates that FLAIR is capable of effectively enhancing the retrieval performance even on datasets with only global captions. Furthermore, when pre-trained on PixelProse, FLAIR achieves an 8% improvement in both text-to-image (T2I) and image-to-text (I2T) retrieval tasks on MSCOCO. These results indicate that FLAIR is versatile and can be applied to datasets where images are captioned by a different MLLM, while maintaining significant performance gains.

Data	Method	MSCOCO T2I R@1	MSCOCO T2I R@5	MSCOCO I2T R@1	MSCOCO I2T R@5	Flickr30k T2I R@1	Flickr30k T2I R@5	Flickr30k I2T R@1	Flickr30k I2T R@5
CC3M-orig [45]	CLIP [41]	4.75	14.53	5.90	17.56	9.19	23.61	12.13	29.68
	FLAIR	**6.45**	**18.14**	**8.00**	**22.48**	**12.70**	**30.20**	**17.55**	**38.66**
PixelProse [46]	CLIP [41]	28.86	54.05	48.50	74.24	54.06	77.81	79.09	94.87
	FLAIR	**36.08**	**61.18**	**56.56**	**79.06**	**64.87**	**85.03**	**86.69**	**97.14**

Table 8. Standard zero-shot image-text retrieval on the validation splits of Flickr30k [40] and MSCOCO [30]. CLIP and FLAIR are pre-trained on CC3M-orig and PixelProse under the same training configurations, using ViT-B/16 as the vision encoder.

	MSCOCO		DOCCI		Urban-1K		VOC20	ImageNet
Merging	T2I@1	I2T@1	T2I@1	I2T@1	T2I@1	I2T@1	mIoU	Top-1
No	35.8	47.1	14.8	33.2	46.4	42.4	52.2	29.9
Always	34.2	46.8	12.4	30.1	**70.9**	**64.7**	54.9	27.7
Random	**37.7**	**51.6**	**15.1**	**35.7**	69.5	63.5	**59.7**	**33.8**

Table 9. Ablation study on merging strategies for sampling captions. No: only sample 1 sentence as the sampled caption. Always: always merge 3 sampled sentences into one caption. Random: each caption is merged randomly from 1-3 sentences. We train FLAIR on CC3M-recap with 8 captions per image.

	MSCOCO		DOCCI		Urban-1K		VOC20	ImageNet
S	T2I@1	I2T@1	T2I@1	I2T@1	T2I@1	I2T@1	mIoU	Top-1
2	37.1	50.7	14.2	35.2	68.5	62.8	59.0	32.0
3	**37.7**	51.6	**15.1**	**35.7**	**69.5**	**63.5**	**59.7**	**33.8**
4	37.5	**52.0**	14.6	35.2	**69.5**	63.2	57.4	33.0

Table 10. Ablation on the maximum number of sentences (S) to be merged to create a new sub-caption. We trained FLAIR with $S \in [2, 4]$ on the CC3M-recap dataset under the same training configuration. The best results are **bold**.

D.2. Sampling Strategy

When sampling diverse captions, we randomly merge $s \in \{1, \ldots, S\}$ sentences in the original MLLM-generated long captions to form a single caption. To evaluate this strategy, we compare FLAIR with three settings: randomly merging 1–3 sentences, always merging 3 sentences, and no merging. The results, presented in Tab. 9, show that always merging 3 sentences improves Urban-1K T2I R@1 and I2T R@1 by 1.4% and 1.2%, respectively. However, it decreases T2I R@1 and I2T R@1 on MSCOCO by 3.5% and 5.2%, indicating a bias towards long retrieval tasks at the expense of short retrieval performance.

Conversely, random merging outperforms the no-merging setting across all metrics, effectively balancing short and long retrieval tasks. Additionally, it enhances model performance by introducing diverse data augmentations through caption variations.

	MSCOCO		DOCCI		Urban-1K		VOC20	ImageNet
K	T2I@1	I2T@1	T2I@1	I2T@1	T2I@1	I2T@1	mIoU	Top-1
CLIP [6]	27.0	38.9	10.3	25.0	41.3	37.7	3.16	23.8
SigLIP [58]	28.3	40.1	10.4	24.9	42.8	40.5	3.1	25.4
2	36.4	49.1	13.9	35.5	68.7	62.9	56.3	31.2
4	36.7	49.8	14.2	35.4	69.1	62.7	57.4	32.8
6	37.4	51.2	14.9	35.4	69.8	61.4	59.5	33.6
8	37.7	51.6	**15.1**	**35.7**	69.5	63.5	59.7	**33.8**
10	**37.8**	**51.7**	15.0	35.1	**71.6**	**64.2**	**60.9**	33.6

Table 11. Ablation results on the number of sub-captions K for FLAIR. OpenCLIP [6], SigLIP [58] and FLAIR are pre-trained on CC3M-recap under the same configuration. All models use ViT-B/16 as vision encoder. The best results are **bold**.

D.3. Number of Merged Sentences

In the diverse caption sampling strategy, each new caption is created by merging up to S sentences. In Tab. 10, we train FLAIR with $S = 2$, $S = 3$, and $S = 4$. Compared to $S = 2$, $S = 3$ yields consistent improvements across all downstream tasks. However, increasing to $S = 4$ does not lead to further gains, likely because merging four sentences often exceeds the 77-token limit of the text encoder. Based on these findings, we set $S = 3$ for our main experiments.

D.4. Number of Sampled Sub-captions

In Tab. 11, we pre-train FLAIR with a different number of sampled captions K ranging from 2 to 10 on the CC3M-recap dataset. We also compared to CLIP and SigLIP pre-trained on the same dataset. First, even when $K = 2$, FLAIR surpasses SigLIP by 8.1% (T2I R@1) and 9.0% (I2T R@1) on MSCOCO retrieval. Increasing to $K = 8$ further brings 1.3% and 2.5% increase in T2I R@1 and I2T R@1 on MSCOCO. Generally, we notice that the performance converges when $K \in (6, 10)$. However, increasing K introduces extra computation overhead, since the text encoder process K captions in every iteration. Therefore, we choose $K = 8$ as our main setting, as it achieves a good balance between performance and computation.

E. Implementation Details

In this section, we describe the detailed implementation of pre-training and downstream tasks evaluation.

Pre-training. Our implementation is based on the OpenCLIP [6] code base with the ViT-B/16 architecture for the image encoder. Both image and text encoder consist of 12 transformer layers, and the embedding size is fixed at 512. Specifically for FLAIR, we replace the final pooling layer of the image encoder with our text-conditioned attention pooling, while the rest of the layers remain unchanged. Our loss function initializes t at 0.07 and b at -10, consistent with the settings used in SigLIP. We follow DreamLIP's pre-training configuration as displayed in Tab. 12. However,

Config	CC3M-recap	CC12M-recap	YFCC15M-recap	Merged-30M
Batch size	$1,024$	$6,134$	$6,134$	$6,134$
Optimizer	AdamW [32]			
Learning rate	5×10^{-4}			
Weight decay	0.5	0.5	0.5	0.2
Adam β	$\beta_1, \beta_2 = (0.9, 0.98)$			
Adam ϵ	1×10^{-8}	1×10^{-8}	1×10^{-8}	1×10^{-6}
Total epochs	32			
Warm up	$2,000$(steps)			
LR scheduler	cosine decay			

Table 12. Pre-training hyper-parameters for FLAIR and all re-trained baseline methods. LR scheduler: Learning Rate scheduler.

Method	Data Size	VOC20	Cityscapes	Context59	ADE20K	COCO-Stuff	Average
CLIP [41]	400M	41.8	**5.5**	**9.2**	**3.2**	4.4	12.8
OpenCLIP [6]	2B	**47.2**	5.1	9.0	2.9	**5.0**	**13.9**
MetaCLIP [53]	2.5B	35.4	5.0	8.1	2.2	4.3	11.0
FLAIR-*CLIP*	3M	**60.9**	8.9	15.6	8.0	9.7	20.6
FLAIR-*TC*	3M	53.9	**20.6**	**23.8**	**13.1**	**13.1**	**24.9**
FLAIR-*CLIP*	12M	**69.7**	14.5	17.4	10.0	12.2	24.8
FLAIR-*TC*	12M	55.1	**20.1**	**22.9**	**13.3**	**15.4**	**25.4**
FLAIR-*CLIP*	15M	**71.5**	13.3	**18.4**	9.0	12.5	**24.9**
FLAIR-*TC*	15M	49.2	**16.5**	17.4	**9.1**	**13.6**	21.2
FLAIR-*CLIP*	30M	**73.0**	**13.6**	**18.6**	10.4	13.3	**25.8**
FLAIR-*TC*	30M	48.3	**13.6**	17.4	**10.8**	**14.4**	20.9

Table 13. Mean Intersection over Union (mIoU) for zero-shot semantic segmentation on VOC20 [13], Cityscapes [8], Context59 [34], ADE20K [61], and COCO-Stuff [2]. All models employ ViT-B/16 as the vision encoder. The best results are **bold**.

we use 6K batch size for CC12M-recap, YFCC15M-recap and Merged30M due to GPU RAM limit. Experiments on CC3M-recap used on 8 NVIDIA A100 40GB GPUs and 32 GPUs on the other datasets. All baseline models, CLIP and SigLIP, follow the same pre-training configurations.

Large-scale Pre-trained CLIP Models. In the main paper, we report the values for OpenCLIP (2B) and SigLIP (10B). Both models employ ViT-B/16 as the vision encoder. Those values were obtained by evaluating the pre-trained weights of OpenCLIP. "OpenCLIP (2B)" refers to the ViT-B/16 model trained on the LAION-2B dataset with the pre-trained name of "laion2b_s34b_b88k". "SigLIP (10B)" refers to the ViT-B/16-SigLIP model trained on the WebLI dataset with the pre-trained name of "webli". The Llip [26] and MetaCLIP [53] results for zero-shot image classification are directly obtained from their papers.

Zero-shot Semantic Segmentation. As discussed in Sec. 4 of the main paper, zero-shot semantic segmentation is based on the similarity between local image tokens and global text queries $\{\langle \mathbf{v}_i^{\text{loc}}, \mathbf{t}_j^{\text{g}} \rangle \mid j \in \{1, 2, \ldots, M\}\}$, where M represents the number of class names in the dataset. Compared to CLIP, a key advantage of FLAIR is its flexibility during inference: it can either directly compute $\langle \mathbf{v}_i^{\text{loc}}, \mathbf{t}_j^{\text{g}} \rangle$ without applying $f_{\text{AttnPool}}(.)$ (FLAIR-*CLIP*), or first use $f_{\text{AttnPool}}(\mathbf{v}_i^{\text{loc}}, \mathbf{t}_j^{\text{g}})$ to generate fine-grained embeddings $\mathbf{v}_{i,j}^{\text{tc}}$, and then compute $\langle \mathbf{v}_{i,j}^{\text{tc}}, \mathbf{t}_j^{\text{g}} \rangle$ (FLAIR-*TC*). Segmentation results for both approaches are reported in Tab. 13. For implementation details, including window size, stride, and other parameters, we used the design choices described in [50].

Interestingly, using the CLIP method increases mIoU on VOC20 by approximately 10%, while the TC method improves performance on other datasets. Both methods outperform OpenCLIP and SigLIP models trained on billions of images. This indicates that the segmentation capability of FLAIR is not solely reliant on the attention pooling layer, because the local image tokens $\mathbf{v}_{\text{loc}}$ encode strong localization information independently.

Zero-shot Image Classification. We follow the prompt ensemble strategy described in LaCLIP [14] and ALIP [54], employing the same prompt templates. For each class name, we compute the average text embedding across all templates, which is then used to calculate the similarity between test images and class embeddings. For zero-shot ImageNet classification, we use the seven prompt templates recommended by [41], consistent with LaCLIP [14].

Top-K Selection in Zero-shot Image-Text Retrieval FLAIR can accelerate inference by employing a top-K feature selection approach similar to ALBEF [28]. First, we use coarse-grained (global) embeddings to compute coarse similarity scores: $(\langle \mathbf{v}_i^{\text{g}}, \mathbf{t}_j^{\text{g}} \rangle)$. For each image i, we select the top-K captions based on these scores. The image is then conditioned only on these captions to generate conditioned embeddings $\mathbf{v}_{i,j}^{\text{tc}}$. Finally, we compute fine-grained similarity scores between the conditioned embeddings and each text embedding $(\langle \mathbf{v}_{i,j}^{\text{tc}}, \mathbf{t}_j^{\text{g}} \rangle)$, as discussed in Sec. 4.1.

We ablate the running time and performance of FLAIR based on different selections of K, as displayed in Tab. 14. The original FLAIR ($K = N$) inference speed of 36.2 ms per image can be sped up to 17.3 ms on MSCOCO without loss of performance when $K = 128$. In this way, the inference overhead over SigLIP is low (17.3ms vs. 13.7ms).

Method	K	MSCOCO			DOCCI			Urban-1K		
		I2T	T2I	Time(ms)	I2T	T2I	Time(ms)	I2T	T2I	Time(ms)
SigLIP	0	28.3	40.1	**13.7**	**10.4**	24.9	**16.0**	42.8	40.5	**13.5**
FLAIR	0	34.9	46.9	15.7	12.6	28.8	18.8	65.9	61.3	14.0
FLAIR	16	35.3	51.5	17.1	12.3	35.3	22.2	**69.5**	**63.6**	14.7
FLAIR	128	**37.7**	**51.6**	17.3	14.8	**35.8**	23.2	**69.5**	63.4	16.2
FLAIR	N	**37.7**	**51.6**	36.2	**15.1**	35.7	50.6	**69.5**	63.5	17.1

Table 14. Per-image running time comparison. K: selecting top-k pairs for precise retrieval. N: #samples in the dataset.