Adaptive Patch Selector for Faster Language-Image Pre-training

Ayush K. Rai
ayush.rai3@mail.dcu.ie

Bo Wan
bwan@esat.kuleuven.be

Federico Cocchi
federico.cocchi@unimore.it

Francesco Tonini
francesco.tonini@unitn.it

Giancarlo Paoletti
giancarlo.paoletti.ext@leonardo.com

Luca Zanella
luca.zanella-3@unitn.it

1. Introduction

Contrastive pre-training of Vision-Language Models (VLMs) e.g. CLIP [7] have demonstrated excellent performance over a variety of tasks such as zero-shot transferability [7], text-to-image generation [8] etc. However, the major bottleneck in training such models is the computational burden and energy consumption involved. For instance, CLIP trained on 400 million data for 32 epochs requires thousands of GPU-days [3] creating a need for more efficient training strategies for VLMs. In this direction, taking inspiration from the sparse computation of Masked Autoencoder [2], the authors of FLIP [4] proposed a method for randomly dropping large portions of image patches while pretraining CLIP models. This enables FLIP to significantly outperform open-source implementation of CLIP (OpenCLIP [3]), while offering faster training speed and noticeable performance over variety of downstream tasks.

However, the main disadvantage of FLIP is that its patch selection mechanism randomly discards image patches without accounting for the information present in them. This motivates us to investigate the following two research question:

1. **Does FLIP's patch selection mechanism need to focus on choosing the most relevant image patches and disregarding the less important ones?**

2. **What is the right way to estimate the relevance (importance) of image patches in terms of quality of information contained in them?**

In this work, we investigate the above research questions by addressing limitations of random patch selection method in FLIP and further present a collection of *proof-of-concept* patch selection strategies. In particular, we propose multiple strategies such as checkerboard pattern, average least similarity, adaptive patch selection based on attention map of DINOv2 [6], and self-distillation from CLIP. We provide zero-shot classification results on ImageNet-1K [1] under variety of training settings (Sec. 2). We also show the performance of CLIP and FLIP when trained from scratch on CC3M [10] dataset, due to unavailability of the larger LAION-400M [9] dataset for our experiments. Our findings have yielded the following important observations:

1. Naive patch selection strategies such as checkerboard pattern and average least similarity achieve similar results to *vanilla* FLIP trained with random masking strategy.

2. Patches selected based on attention maps extracted from DINOv2 [6] outperform naive approaches and FLIP, indicating the effectiveness of our propose adaptive patch selection strategy.

3. The patches selected through self-distillation from CLIP using KL divergence based loss (end-to-end pipeline) performs marginally better than random patch selection strategy, which further validates the proficiency of our approach.

2. Method

In our proposed framework, shown in Fig. 1, we first introduce heuristic methods such as *checkerboard* and *average least similarity*. Then, we propose adaptive patch selectors that exploit the attention map of DINOv2 [6]. However, due to the computational requirement of DINOv2, we also propose a *self-distillation* approach that leverages the holistic similarity available in CLIP.

2.1. Checkerboard pattern

This method alternates between selecting odd and even patches in a checkerboard-like pattern, ensuring uniform pixel distribution for enhanced model robustness against local variations.

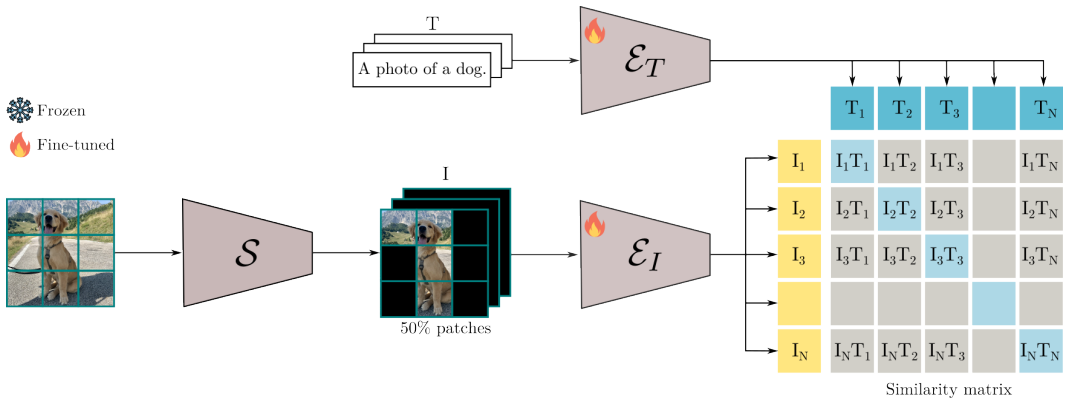

Figure 1. Illustration of our proposed framework. The Selector model $\mathcal{S}$ identifies the most informative patches from the input images. These selected patches are subsequently fed into CLIP's image encoder $\mathcal{E}_\mathcal{I}$ for pre-training for image-text semantic alignment within the latent space.

2.2. Average least similarity

In Fig. 2a, we project the input image into D-dimensional patches and calculate the cosine similarity between all patch pairs. We then retain patches with the lowest average similarity, emphasizing those with richer and less redundant information, fostering effective selection for the transformer model.

2.3. Learning from DINOv2's attention map

As shown in Fig. 2b, our method's patch selector model, denoted as $\mathcal{S}$, consists of three components: a pre-trained DINOv2 image encoder $\mathcal{E}$ serving as an attention-based patch relevance estimator, an attention aggregator, and a selection strategy. In particular, when provided with an input image $\mathcal{I}$, $\mathcal{S}$ first extracts attention maps from the `[CLS]` token produced by the final layer of $\mathcal{E}$. Subsequently, $\mathcal{S}$ combines the various attention maps generated by different attention heads, each focusing on distinct representative aspects of objects. Ultimately, $\mathcal{S}$ utilizes a multinomial probability distribution derived from the attention scores to randomly choose 50% of the visual tokens.

The areas of focus of attention heads should correspond to the way humans perceive objects in images and, by extension, nouns in text captions. We investigate three different approaches for combining attention maps generated by these heads: entropy-based aggregation, mean-based aggregation, and sum-based aggregation. In entropy-based aggregation, we compute a probability distribution of attention scores for each head. Low entropy signifies agreement among heads, often indicating the background, while higher entropy suggests uncertainty among heads, highlighting potentially informative regions with object-specific features. In the mean-based aggregation, we calculate the average of the values of the attention heads, while in the sum-based aggregation, we sum the values of the attention heads.

2.4. Self-distillation from CLIP

A straightforward approach to patch selection is to identify and prioritize the most pertinent patches within the input texts. Leveraging the holistic similarity offered by a pre-trained CLIP model between images and texts, we can extend this similarity analysis to a more granular, patch-level perspective, using it to guide our patch selection process.

In detail, for the patch tokens in each image, we employ a 2-layer transformer to compute their importance scores. Subsequently, we select 50% patch tokens (denote the token number as $K = 24$) based on the normalized importance scores $S_{pred} \in \mathbf{R}^K$. These selected patches are then input into a Vision Transformer to generate the image representation, following a similar methodology as FLIP. The importance scores are supervised by comparing the similarity of the CLIP text embeddings with the patch embeddings, which is denoted as $S_{clip} \in \mathbb{R}^K$. We use a standard KL divergence as a loss term, i.e., $\mathcal{L}_{distill} = KL_{dist}(S_{pred}, S_{clip})$. This loss term serves as a means to refine our selector's ability to select patches that are most congruent with the textual context, promoting a more robust and contextually relevant image representation.

3. Experiments

All experiments were conducted on the CC3M [10] dataset, consisting of 3 million pairs of image-text collected from the web followed by a series of filters. The batch size used for the experiments is 16k, to match the implementation used by FLIP [4], with the sole exception of DINO, where the batch size is 4k. This was possible by fully utilizing 4 nodes and 16 GPUs of the Leonardo HPC. We consider the implementation of CLIP and FLIP, where image patches are dropped randomly, as baselines. The latter is evaluated after 20 and 200 epochs, in order to assess the trend during training. Results exploiting the heuristics for

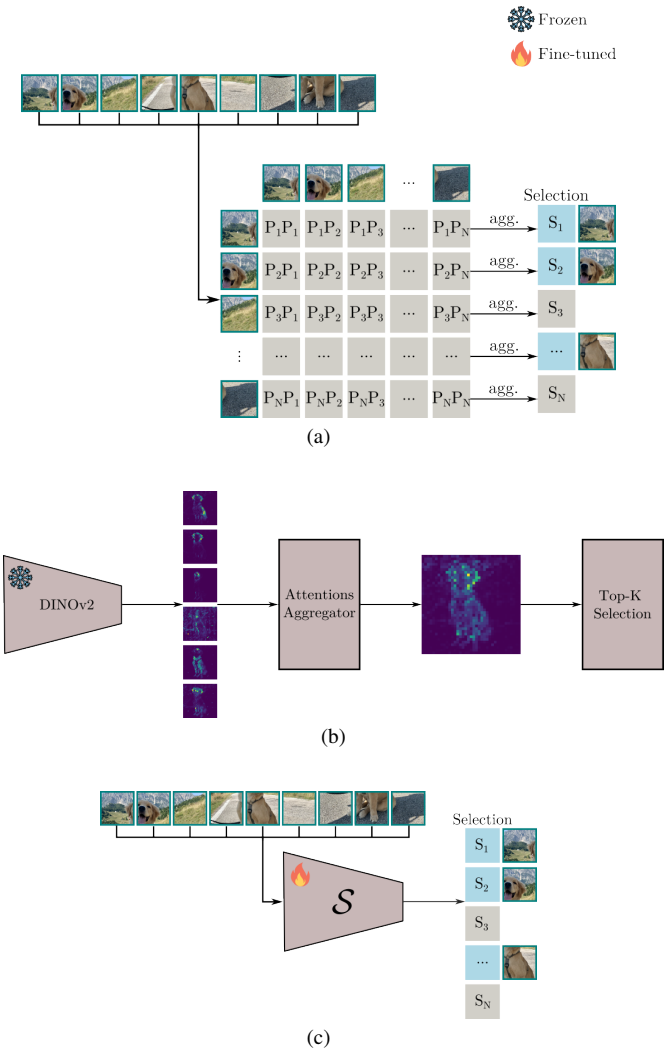

Figure 2. Overview of the various patch selection strategies: (a) Average Least Similarity (b) Attention Map-Based (DINOv2) (c) Learned Selector via Self-Distillation Loss. Please refer to Sec. 2 for additional details.

the patches filter obtain comparable results with the baseline. Instead, results with DINO's feature-based approaches have a performance improvement and reach 8% in zero-shot on Imagenet-1k.

Due to the small amount of data used for training, we also evaluated a model with a smaller number of parameters (ViT-S/32). The results of the models of different sizes are comparable with each other.

4. Discussion

The results clearly demonstrate that CC3M does not enable either CLIP or FLIP (Random) to attain competitive performance comparable to their training on LAION-400M dataset. This behavior stems from the restricted size of the CC3M dataset and the project's time constraints. Additionally, there are no pretrained checkpoints available for CLIP and FLIP models finetuned on CC3M data. Moreover, because of the substantial differences in training iterations per epoch, we have chosen not to employ a checkpoint pretrained on the LAION-400M dataset, which is currently unavailable on the cluster.

In our evaluation, we observe that heuristic patch selection techniques, such as the checkerboard pattern and average least similarity, achieve results on par with a conventional FLIP model trained using a random masking strategy. However, when patches are chosen based on attention maps extracted from DINOv2 [6], we obtain a remarkable improvement, surpassing both naive methods and FLIP. This outcome highlights the efficacy of our proposed adaptive patch selection strategy. Furthermore, patches selected via self-distillation from CLIP, utilizing a KL divergence-based loss in an end-to-end pipeline, also outperform a random patch selection strategy. This finding underscores our hypothesis that FLIP may lead to the loss of important details when patches are randomly sampled.

In future work, we plan to further experiment with training strategy of the self-distillation pipeline such as training CLIP without any patch drop for initial few epochs and then incorporate self-distillation loss to guide the patch selection process. Also an interesting experiment would to incorporate slot attention [5] in the patch selection mechanism. We hope our work can inspire further research in this direction.

Table 1. Zero-shot accuracy on ImageNet-1k using CLIP pretrained on CC3M. Our attention maps are based on DinoV2. The Random patch selection strategy we used is FLIP. The "Baseline" denotes vanilla CLIP, where all patches were utilized.

Visual Backbone	*Patch selection strategy*	*# Epochs*	*Top 1 (%)*	*Top 5 (%)*
ViT-B 32	- (Baseline)	15	5.4	14.5
ViT-B 32	Random	20	4.8 (↓ 0.6)	13.1 (↓ 1.4)
ViT-B 32	Random	200	5.7 (↑ 0.3)	14.8 (↑ 0.3)
ViT-S 32	Random	200	4.7 (↓ 0.7)	12.7 (↓ 1.8)
ViT-B 32	Checkerboard	200	5.4 (≈ 0.0)	13.7 (↓ 0.8)
ViT-B 32	Avg. least similarity	200	5.6 (↑ 0.2)	14.2 (↓ 0.3)
ViT-B 32	Self distillation loss	20	5.2 (↓ 0.2)	14.3 (↓ 0.2)
ViT-B 32	Self distillation loss	50	5.6 (↑ 0.2)	14.3 (↓ 0.2)
ViT-B 32	Attention Map (Sum)	11	8.3 (↑ 2.9)	20.2 (↑ 5.7)
ViT-B 32	Attention Map (Mean)	14	8.3 (↑ 2.9)	20.3 (↑ 5.8)
ViT-B 32	Attention Map (Entropy)	15	**8.5** (↑ 3.1)	**20.5** (↑ 6.0)

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[2] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

[3] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Han-

naneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, 2021.

[4] Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. Scaling language-image pre-training via masking. In *CVPR*, 2023.

[5] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *NeurIPS*, 2020.

[6] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

[7] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. 2021.

[8] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[9] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[10] 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.